Transfinite barycentric coordinates for arbitrary planar domains

Qingjun Chang · Kai Hormann

Abstract

Generalized barycentric coordinates provide a simple way of interpolating data given at the vertices of a polygon or polyhedron, with widespread applications in computer graphics, geometry processing, and other fields. Transfinite barycentric coordinates, also known as barycentric kernels, extend this idea to curved domains and can be used to interpolate continuous data given on the boundary of such domains. We present a novel framework for defining non-negative barycentric kernels over arbitrary bounded planar domains. This framework is inspired by the construction of a transfinite version of maximum likelihood coordinates and can be used to define a variety of barycentric kernels, including a simple pseudo-harmonic kernel and a non-negative variant of the mean value kernel. Moreover, we propose a novel barycentric kernel which yields transfinite interpolants that are similar to harmonic interpolants. We tested our new kernel for domains and boundary data described by closed uniform quadratic B-splines and in particular for image deformation. The results indicate that our method has several advantages over alternative approaches.

Citation Info

Journal
Computer Aided Geometric Design

Volume
119, June 2025

Article
102433, 13 pages

Note
Proceedings of GMP

DOI
10.1016/j.cagd.2025.102433

1 Introduction

Generalized barycentric coordinates are widely used in applications like surface design [28], mesh parameterization [10] and deformation [23, 22, 38], colour interpolation [31], shading [21], image warping [40] and compositing [9], finite element methods [37, 16], and many more [20]. All these applications rely on the fact that data given at the vertices of a polygon or polyhedron can be interpolated efficiently by means of generalized barycentric coordinates.

In certain situations, it can be desirable to work with curved domains instead of polygons and to interpolate data defined at the boundary of such a domain (see Figure 1), which has led to the development of transfinite barycentric coordinates [2, 40].

Given a bounded planar domain Ω and a regular parameterization $p\colon[a,b]\to\partial\Omega$ of the boundary of Ω with $p(a)=p(b)$, a function $\lambda\colon\Omega\times[a,b]\to\mathbb{R}$ is called a *barycentric kernel* with respect to Ω, if

$$\int_a^b \lambda(x,t)\,\mathrm{d}t = 1, \qquad \int_a^b \lambda(x,t)p(t)\,\mathrm{d}t = x \tag{1}$$

for any $x\in\Omega$. As these conditions are the continuous analogue of the *partition of unity* and the *linear reproduction* properties of generalized barycentric coordinates [20], the values $\lambda(x,t)$, $t\in[a,b]$ are also referred to as *transfinite barycentric coordinates* of $x\in\Omega$ with respect to Ω.

(a) (b) (c) (d)

Figure 1: Interpolants of height data (ranging from 0 to 1) given at the boundary of a non-convex planar domain, based on transfinite mean value (a), positive Gordon–Wixom (b), harmonic (c), and our novel transfinite barycentric coordinates (d). The left images show a 3D view of each interpolant, the right images an orthographic projection from above. Height values in the intervals $[k,k+1]/10$ for $k=-1,0,\dots,10$ are colour-coded as indicated by the colour bar, with negative values in red, values above 1 in green, and values between 0 and 1 in colours from blue to yellow.

In addition to the conditions in (1), λ should satisfy the continuous equivalent of the *Lagrange property*,

$$\lambda(p(s),t)=\delta(s-t)$$

for any $s,t\in[a,b]$, where δ is the *Dirac delta function*, so that the *transfinite barycentric interpolant*

$$g:\Omega\to\mathbb{R},\qquad g(x)=\int_a^b \lambda(x,t)f(p(t))\,\mathrm{d}t \tag{2}$$

of continuous data $f:\partial\Omega\to\mathbb{R}$ given on the boundary of Ω is indeed an interpolant, that is, $g(y)=f(y)$ for any $y\in\partial\Omega$. The conditions in (1) ensure that this interpolant reproduces linear functions $h:\Omega\to\mathbb{R}$ in the sense that $g(x)=h(x)$ for any $x\in\Omega$, if $f(y)=h(y)$ for all $y\in\partial\Omega$.

Moreover, many applications require λ to be *non-negative*, that is,

$$\lambda(x,t)\geq 0$$

for any $x\in\Omega$, $t\in[a,b]$, so that the interpolated values $g(x)$ are inside the convex hull of the boundary data $f(\partial\Omega)=\{f(y):y\in\partial\Omega\}$. Finally, λ should be *smooth* (at least C^1) in x, because this smoothness carries over to the interpolant g. Note that the smoothness of λ in t is irrelevant, since $g(x)$ in (2) is defined by integrating with respect to this parameter.

Similar to generalized barycentric coordinates, barycentric kernels are often defined by first focussing on the interior of Ω and specifying a function $\mu\colon \operatorname{int}\Omega\times[a,b]\to\mathbb{R}$ with

$$\int_a^b \mu(x,t)(p(t)-x)\,\mathrm{d}t=0 \tag{3}$$

for any $x\in\operatorname{int}\Omega$ and then letting

$$\lambda(x,t)=\mu(x,t)\Big/\int_a^b \mu(x,s)\,\mathrm{d}s. \tag{4}$$

Together with (3), this normalization implies (1), and if μ is smooth and non-negative, then so is λ. In addition to the property in (3), the function μ should be such that $\lambda(x,t)$ in (4) is well-defined for all $x\in\operatorname{int}\Omega$, $t\in[a,b]$ and such that the continuous extension of λ over $\Omega\times[a,b]$ is interpolatory.

1.1 Related work

Warren et al. [40] were the first to propose the extension of generalized barycentric coordinates to convex domains and to reason why Wachspress coordinates [39, 31] converge to the *Wachspress kernel*, given by (4) with

$$\mu(x,t)=\frac{p'(t)\times p''(t)}{\big((p(t)-x)\times p'(t)\big)^2},$$

where $u\times v=u_1v_2-u_2v_1$ denotes the 2D cross product of the vectors $u=(u_1,u_2)^\mathsf{T}$ and $v=(v_1,v_2)^\mathsf{T}$.

A similar extension of mean value coordinates [11, 19] leads to the *mean value kernel* [23, 7], given by (4) with

$$\mu(x,t)=\frac{(p(t)-x)\times p'(t)}{\|p(t)-x\|^3}. \tag{5}$$

Both kernels are non-negative and special cases of a whole family of barycentric kernels [35] that extend the family of 3-point coordinates [13] from convex polygons to convex domains. Another special case is the *Laplace kernel* [25], the continuous version of discrete harmonic coordinates [33, 8], given by (4) with

$$\mu(x,t)=\frac{2\|p'(t)\|^2(p(t)-x)\times p'(t)-\|p(t)-x\|^2p'(t)\times p''(t)}{\big((p(t)-x)\times p'(t)\big)^2},$$

but this kernel can be negative. For all members of this family, the discrete interpolants converge quadratically to the transfinite interpolants [24, 25], and a rigorous proof of the interpolation property of non-negative barycentric kernels over convex domains is given by Floater and Kosinka [14].

Transfinite barycentric interpolants over convex domains can also be expressed as weighted Gordon–Wixom interpolants. The idea of Gordon and Wixom [17] is to consider for any $x \in \Omega$ and $\theta \in [0, 2\pi]$ the line L passing through x at angle θ and consisting of the points $L(r) = x + r(\cos\theta, \sin\theta)^\mathsf{T}$, $r \in \mathbb{R}$. As Ω is convex, L intersects $\partial\Omega$ in exactly two points, $y_1 = L(d_1)$ and $y_{-1} = L(-d_{-1})$, where $d_1 = \|y_1 - x\|$ and $d_{-1} = \|y_{-1} - x\|$. Linearly interpolating the boundary data at these points with respect to x gives

$$g_{1,-1}(x,\theta) = \left(\frac{f(y_1)}{d_1} + \frac{f(y_{-1})}{d_{-1}}\right) \Big/ \left(\frac{1}{d_1} + \frac{1}{d_{-1}}\right),$$

and averaging over θ then defines the *Gordon–Wixom interpolant*

$$g(x) = \frac{1}{2\pi} \int_0^{2\pi} g_{1,-1}(x,\theta)\,\mathrm{d}\theta. \tag{6}$$

Belyaev [2] generalizes this interpolation scheme by introducing a weight function $\omega\colon \Omega \times [0, 2\pi] \to \mathbb{R}$,

$$g(x) = \int_0^{2\pi} g_{1,-1}(x,\theta)\omega(x,\theta)\,\mathrm{d}\theta \Big/ \int_0^{2\pi} \omega(x,\theta)\,\mathrm{d}\theta,$$

and shows that these *weighted Gordon–Wixom interpolants* are linearly precise if ω is *centrally-symmetric*, that is, $\omega(x,\theta) = \omega(x,-\theta)$. Moreover, if $\omega(x,\theta)$ is a function of $d = d_1 d_{-1}$, then this interpolation scheme is *pseudo-harmonic*, that is, it reproduces harmonic interpolants if Ω is a disk. This is clearly the case for the weight function $\omega(x,\theta) = 1$ which gives the original Gordon–Wixom interpolant in (6), but not for the weight functions that reproduce transfinite Wachspress and mean value interpolants.

A third alternative is to express barycentric kernels as *similarity kernels* using complex numbers. In this setting, the transfinite versions of the 3-point coordinates have very simple expressions, and it is easy to show that the Wachspress and the Laplace kernel are identical if Ω is a circle [5]. Moreover, the complex formulation reveals that the transfinite version of moving least squares coordinates [30] is pseudo-harmonic but possibly negative [6].

Some of the aforementioned barycentric interpolation schemes extend or can be generalized to non-convex domains. For example, the mean value weight function in (5) is well-defined for arbitrary domains with piecewise C^1-continuous boundaries, but no longer guaranteed to be non-negative for all $x \in \Omega$. Hence, the transfinite mean value interpolant may take on values outside the convex hull of the boundary data (see Figure 1a), which in turn can cause non-intuitive results in applications like image deformation (see Figure 11). The same holds for the generalization of the weighted Gordon–Wixom interpolant to non-convex domains proposed by Belyaev [2], which replaces the linear interpolation $g_{1,-1}$ by barycentric rational interpolation [36, 3, 12] of the boundary data at all the intersection points of the line L and $\partial\Omega$.

Manson et al. [29] propose an alternative extension of Gordon–Wixom coordinates to arbitrary domains that overcomes this limitation. Denoting by $y_1, \ldots, y_m$ the intersection points of L and $\partial\Omega$ which correspond to positive line parameters, that is, $y_i = L(d_i)$ with $d_i = \|y_i - x\|$ for $i = 1, \ldots, m$ and by $y_{-n}, \ldots, y_{-1}$ those with negative parameters, $y_j = L(-d_{-j})$, $d_{-j} = \|y_{-j} - x\|$, $j = 1, \ldots, n$, they suggest to consider the linearly interpolated values

$$g_{i,j}(x,\theta) = \left(\frac{f(y_i)}{d_i} + \frac{f(y_{-j})}{d_{-j}}\right) \Big/ \left(\frac{1}{d_i} + \frac{1}{d_{-j}}\right)$$

for all pairs of boundary data at $y_1, \ldots, y_m$ and at $y_{-n}, \ldots, y_{-1}$ and to average and normalize their weighted sum over θ,

$$g(x) = \int_0^{2\pi} \sum_{i=1}^{m} \sum_{j=-n}^{-1} g_{i,j}(x,\theta)\omega_{i,j}(x,\theta)\,\mathrm{d}\theta \Big/ \int_0^{2\pi} \sum_{i=1}^{m} \sum_{j=-n}^{-1} \omega_{i,j}(x,\theta)\,\mathrm{d}\theta,$$

where the weight functions are defined as $\omega_{i,j}(x,\theta) = \frac{d_i + d_j}{d_i^2 d_j^2} h_i h_j$ with h_k denoting the unsigned distance between x and the line tangent to $\partial\Omega$ at y_k for $k \in \{-n, \ldots, -1, 1, \ldots, m\}$. Manson et al. [29] show that g interpolates the given boundary data (see Figure 1b) and can be expressed as in (2) for some non-negative barycentric kernel λ which is as smooth as the boundary of Ω. However, this interpolation scheme is not pseudo-harmonic [6].

To the best of our knowledge, apart from these transfinite positive Gordon–Wixom (PGW) coordinates, the only barycentric kernel that is non-negative for arbitrary domains is the *Poisson kernel*, giving rise to

harmonic interpolants of the boundary data (see Figure 1c), but this kernel does not have an explicit form, except in some special cases. For example, if Ω is the unit disk, then the Poisson kernel is given by (4) with

$$\mu(x,t) = \frac{1}{\|p(t)-x\|^2}. \tag{7}$$

In addition to the aforementioned approaches, there exist several constructions of transfinite barycentric coordinates that differ from the definition above. On the one hand, it is possible to extend the transfinite mean value interpolant to the Hermite setting [7, 15, 26], in which the interpolant also matches given derivative data along the boundary. On the other hand, transfinite Cauchy–Green coordinates [32, 27] give rise to conformal mappings, but they do not have the interpolation property.

1.2 Contribution

We propose a novel and flexible framework for constructing barycentric kernels for arbitrary domains. In essence, our construction is a transfinite and generalized version of maximum likelihood coordinates [4], where the generalization is achieved by introducing an appropriate weight function. The resulting barycentric kernels are as smooth as this weight function and non-negative (Section 2). We use this framework to define a simple pseudo-harmonic kernel, a novel kernel that is similar to the Poisson kernel for a wide range of domains, and a non-negative variant of the mean value kernel (Section 3).

The results (Section 4) show that our barycentric kernels give rise to transfinite barycentric interpolants that are faster to evaluate and exhibit better smoothness than positive Gordon–Wixom interpolants. Moreover, the weight function can be chosen such that the interpolants are similar to harmonic interpolants, which have no closed form in general and must be approximated numerically, usually by functions that are piecewise linear over the triangles of a domain triangulation. While this approximation requires solving a large global linear system, our interpolants can be evaluated at any domain point by solving a local non-linear convex optimization problem in just two variables.

2 A general framework for non-negative barycentric kernels

2.1 Discrete maximum likelihood coordinates

The construction of maximum likelihood coordinates [4] can be summarized as follows. Given a planar polygon Ω with vertices $p_1,\dots,p_n$ and some point $x \in \operatorname{int}\Omega$, we first project the vertices onto the unit circle around x, yielding the projected points

$$\hat{p}_i = \frac{p_i - x}{r_i}, \qquad r_i = \|p_i - x\|, \qquad i = 1,\dots,n. \tag{8}$$

This step, which assures the Lagrange property of the coordinates, is followed by two averaging steps that are required for guaranteeing the linearity of the coordinates along the edges of Ω. These two steps replace each $\hat{p}_i$ with a normalized convex combination of $\hat{p}_{i-1}$, $\hat{p}_i$, and $\hat{p}_{i+1}$, but we skip the details, because they can be ignored for our purposes, as we shall see below in Section 2.2.

The next step is to determine positive barycentric coordinates $\hat{\lambda}_1,\dots,\hat{\lambda}_n$ of the origin $0 \in \mathbb{R}^2$ with respect to $\hat{p}_1,\dots,\hat{p}_n$ by maximizing $\sum_{i=1}^n \log \hat{\lambda}_i$ subject to the constraints $\sum_{i=1}^n \hat{\lambda}_i = 1$ and $\sum_{i=1}^n \hat{\lambda}_i \hat{p}_i = 0$. The *maximum likelihood coordinates* of $x \in \operatorname{int}\Omega$ are finally defined as

$$\lambda_i(x) = \frac{\hat{\lambda}_i}{r_i} \Bigg/ \sum_{j=1}^n \frac{\hat{\lambda}_j}{r_j}, \qquad i = 1,\dots,n. \tag{9}$$

The continuous extensions of these coordinates to Ω satisfy all key properties of generalized barycentric coordinates and are non-negative, even for non-convex polygons.

2.2 Transfinite maximum likelihood coordinates

In order to derive a transfinite version of these coordinates, let us investigate what happens in the limit if we construct maximum likelihood coordinates and interpolants with respect to an increasing number of uniform samples from the parametric boundary p of a bounded planar domain Ω (see Figure 2).

Figure 2: Maximum likelihood interpolant for an increasing number of uniform samples along the boundary of the domain in Figure 1, for the same boundary data.

For any $n \geq 3$, let Ω_n be the polygon with vertices $p_i = p(t_i)$, $i = 1,\dots,n$, where $t_i = a + i(b-a)/n$. In the limit, as $n \to \infty$, the set of vertices $\{p_i : i = 1,\dots,n\}$ becomes dense and converges to the continuous set of boundary points $\{p(t) : t \in [a,b]\}$, and Ω_n converges to Ω. Likewise, the set of projected points in (8) converges to $\hat{p}([a,b])$, where $\hat{p} : [a,b] \to S^1$ with

$$\hat{p}(t) = \frac{p(t)-x}{r(t)}, \qquad r(t) = \|p(t)-x\|, \qquad t \in [a,b] \tag{10}$$

is the parameterized curve that we obtain by projecting the boundary curve p onto the unit circle around x. As the distance between $\hat{p}_i$ and $\hat{p}_{i+1}$ converges to zero, it is now clear why the two averaging steps have no effect in the limit and can thus be ignored in the transfinite setting. This can also be seen in Figure 2, where the result for $n = 500$ was computed with these two averaging steps, but the result for $n = \infty$ was obtained by discretizing the transfinite interpolant without any averaging steps (see Section 4.1).

The problem of determining positive coordinates $\hat{\lambda}_1,\dots,\hat{\lambda}_n$ then turns into the problem of finding a positive function $\hat{\lambda} : [a,b] \to \mathbb{R}$ that maximizes $\int_a^b \log \hat{\lambda}(t)\,\mathrm{d}t$ subject to the constraints $\int_a^b \hat{\lambda}(t)\,\mathrm{d}t = 1$ and $\int_a^b \hat{\lambda}(t)\hat{p}(t)\,\mathrm{d}t = 0$. At this point we propose to generalize the construction by introducing a positive weight function $\omega_x : [a,b] \to \mathbb{R}$, which may depend on $x \in \operatorname{int}\Omega$, and to find more generally the function $\hat{\lambda}$ that maximizes

$$\ell[\hat{\lambda}] = \int_a^b \log \hat{\lambda}(t)\omega_x(t)\,\mathrm{d}t \tag{11}$$

subject to the constraints

$$\int_a^b \hat{\lambda}(t)\omega_x(t)\,\mathrm{d}t = 1, \qquad \int_a^b \hat{\lambda}(t)\hat{p}(t)\omega_x(t)\,\mathrm{d}t = 0. \tag{12}$$

Clearly, the original construction is just a special case of this generalization, obtained by using the constant weight function $\omega_x(t) = 1$.

As in the discrete case [4], we can use Lagrangian multipliers to solve this constrained optimization problem, but we need to invoke some variational calculus. The *Lagrangian* for the problem is

$$L(t,\hat{\lambda}) = \log \hat{\lambda}(t)\omega_x(t) - \phi_0\big(\hat{\lambda}(t)\omega_x(t) - \tfrac{1}{b-a}\big) - \phi^{\mathsf{T}}\hat{\lambda}(t)\hat{p}(t)\omega_x(t),$$

where $\phi_0 \in \mathbb{R}$ and $\phi = (\phi_1,\phi_2)^{\mathsf{T}} \in \mathbb{R}^2$ are the Lagrangian multipliers. Since the functional ℓ in (11) is strictly concave over the space of positive functions, its maximum under the constraints in (12) is obtained at a stationary point $\hat{\lambda}$ of the functional

$$J[\hat{\lambda}] = \int_a^b L(t,\hat{\lambda})\,\mathrm{d}t = \ell[\hat{\lambda}] - \phi_0\left(\int_a^b \hat{\lambda}(t)\omega_x(t)\,\mathrm{d}t - 1\right) - \phi^{\mathsf{T}}\int_a^b \hat{\lambda}(t)\hat{p}(t)\omega_x(t)\,\mathrm{d}t.$$

According to the fundamental lemma of calculus of variations, finding a stationary point of $J[\hat{\lambda}]$ is equivalent to solving the associated *Euler–Lagrange equation*

$$\frac{\partial L}{\partial \hat{\lambda}}(t,\hat{\lambda}) = \frac{\omega_x(t)}{\hat{\lambda}(t)} - \phi_0\omega_x(t) - \phi^{\mathsf{T}}\hat{p}(t)\omega_x(t) = \left(\frac{1}{\hat{\lambda}(t)} - \phi_0 - \phi^{\mathsf{T}}\hat{p}(t)\right)\omega_x(t) = 0. \tag{13}$$

Since ω_x is a positive function, the optimal $\hat{\lambda}$ is therefore defined as

$$\hat{\lambda}(t) = \frac{1}{\phi_0 + \phi^{\mathsf{T}}\hat{p}(t)}, \qquad t \in [a,b]. \tag{14}$$

To determine the value of the Lagrangian multiplier ϕ_0, we multiply both sides of (13) by $\hat{\lambda}(t)$ and integrate over t,

$$\int_a^b \omega_x(t)\,\mathrm{d}t - \phi_0 \int_a^b \hat{\lambda}(t)\omega_x(t)\,\mathrm{d}t - \phi^\mathsf{T} \int_a^b \hat{\lambda}(t)\hat{p}(t)\omega_x(t)\,\mathrm{d}t = 0,$$

and after inserting the constraints (12), we get

$$\phi_0 = \int_a^b \omega_x(t)\,\mathrm{d}t. \tag{15}$$

To find the other Lagrangian multiplier ϕ, we use (14) in the second constraint in (12) and notice that

$$\int_a^b \hat{\lambda}(t)\hat{p}(t)\omega_x(t)\,\mathrm{d}t = \int_a^b \frac{\hat{p}(t)}{\phi_0 + \phi^\mathsf{T}\hat{p}(t)}\omega_x(t)\,\mathrm{d}t = -\nabla F(\phi)$$

for the function $F\colon \Phi \to \mathbb{R}$ with

$$F(\phi) = -\int_a^b \log\bigl(\phi_0 + \phi^\mathsf{T}\hat{p}(t)\bigr)\omega_x(t)\,\mathrm{d}t. \tag{16}$$

Hence, satisfying this second constraint is equivalent to finding the unique minimum of the strictly convex function F. Note that the domain Φ of F consists of all ϕ for which $\phi_0 + \phi^\mathsf{T}\hat{p}(t) > 0$ for all $t \in [a,b]$, which guarantees that the optimal $\hat{\lambda}$ in (14) is indeed a positive function. As $\hat{p}(t)$ covers the whole unit circle S^1 for $t \in [a,b]$, we conclude that Φ is the open disk of radius ϕ_0 around the origin, that is, $\Phi = \{\phi \in \mathbb{R}^2 : \|\phi\| < \phi_0\}$. In analogy to the discrete setting [4], we see that solving the initial constrained optimization problem over the space of positive functions simplifies to the minimization of a bivariate convex function over a disk, which can be done efficiently with Newton's method, using $0 \in \Phi$ as initial guess.

Once $\hat{\lambda}$ is found, the *weighted maximum likelihood kernel* at $x \in \operatorname{int}\Omega$ is defined as

$$\lambda(x,t) = \frac{\hat{\lambda}(t)\omega_x(t)}{r(t)} \Bigg/ \int_a^b \frac{\hat{\lambda}(s)\omega_x(s)}{r(s)}\,\mathrm{d}s, \qquad t \in [a,b], \tag{17}$$

the continuous equivalent of (9). This kernel is well-defined and positive, because $\hat{\lambda}$, ω_x, and r are positive functions for any $x \in \operatorname{int}\Omega$, and it clearly satisfies the partition of unity property in (1). Moreover, since $p(t) = \hat{p}(t)r(t) + x$ by (10), it follows from (12) that

$$\begin{aligned}\int_a^b \lambda(x,t)p(t)\,\mathrm{d}t &= \int_a^b \frac{\hat{\lambda}(t)\omega_x(t)}{r(t)}\bigl(\hat{p}(t)r(t) + x\bigr)\,\mathrm{d}t \Bigg/ \int_a^b \frac{\hat{\lambda}(s)\omega_x(s)}{r(s)}\,\mathrm{d}s \\ &= \left(\int_a^b \hat{\lambda}(t)\hat{p}(t)\omega_x(t)\,\mathrm{d}t + x\int_a^b \frac{\hat{\lambda}(t)\omega_x(t)}{r(t)}\,\mathrm{d}t\right) \Bigg/ \int_a^b \frac{\hat{\lambda}(s)\omega_x(s)}{r(s)}\,\mathrm{d}s = x,\end{aligned}$$

which confirms that this kernel is also linearly precise.

3 Choosing the weight function

3.1 The unweighted kernel

Figure 2 shows how the maximum likelihood interpolants for an increasing number of n uniform boundary samples converge to the function g in (2) using the maximum likelihood kernel λ in (17) with the constant weight function $\omega_x(t) = 1$. Comparing the values along the boundary with those from the interpolants in Figure 1, they seem to be different, indicating that λ might not be interpolatory, although it is reasonable to expect this property to carry over from the discrete setting. In fact, since the factor $1/r(t)$ in (17), which is a consequence of the initial projection step, diverges to infinity at $t = s$ but remains finite for any $t \neq s$ as x approaches a boundary point $p(s)$, the kernel λ should converge to $\delta(t-s)$ in the limit. Although it remains future work to prove this conjecture, our experiments (see Figure 3b) suggest that this is indeed the case. However, the convergence is very slow and even at a distance of 10^{-4} from the boundary, the interpolated values are still quite far from the boundary values (see Figure 3e).

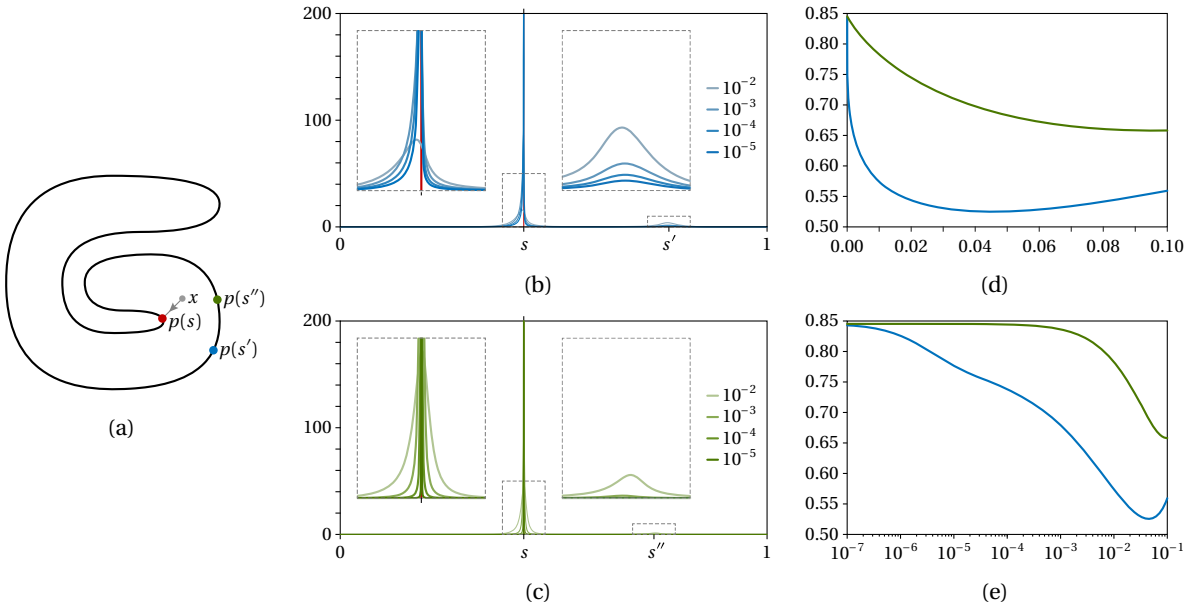


Figure 3: Behaviour of weighted maximum likelihood kernels with weight functions $\omega_x(t)=1$ (blue) and $\omega_x(t)=1/r(t)$ (green) and corresponding interpolants near the boundary of the domain from Figure 1, whose bounding box is the unit square. As x approaches $p(s)$ (a), the kernel $\lambda(x,t)$, plotted over t for x at a distance of 10^{-k} with $k=2,3,4,5$, converges to $\delta(t-s)$ (b,c) and the interpolant, shown for x at a distance from 0 to 0.1 in linear (d) and logarithmic (e) scale, converges to $f(s)$. Note how the local maximum of $\lambda(x,t)$ at s' (b) or s'' (c), which corresponds to the influence of the boundary data near $p(s')$ or $p(s'')$ on the interpolant, diminishes as x gets closer to $p(s)$.

3.2 A pseudo-harmonic kernel

To improve the boundary behaviour of the interpolant, we can use the weight function

$$\omega_x(t)=\frac{1}{r(t)}. \tag{18}$$

This effectively replaces the factor $1/r(t)$ in (17) with $1/r(t)^2$, which diverges faster to infinity at $t=s$ as x approaches $p(s)$, thus guaranteeing a faster convergence of λ to $\delta(t-s)$ (see Figure 3c). Moreover, the resulting kernel turns out to be pseudo-harmonic. To see this, let Ω be the unit disk with boundary parameterization $p(t)=(\cos t,\sin t)^\mathsf{T}$ for $t\in[0,2\pi]$. In this setting, the gradient of F in (16) is

$$\nabla F(\phi)=-\int_0^{2\pi}\frac{1}{\phi_0+\phi^\mathsf{T}\hat{p}(t)}\cdot\frac{p(t)-x}{r(t)^2}\,\mathrm{d}t.$$

Recalling that the identity function can be expressed, with the help of the Poisson kernel, as

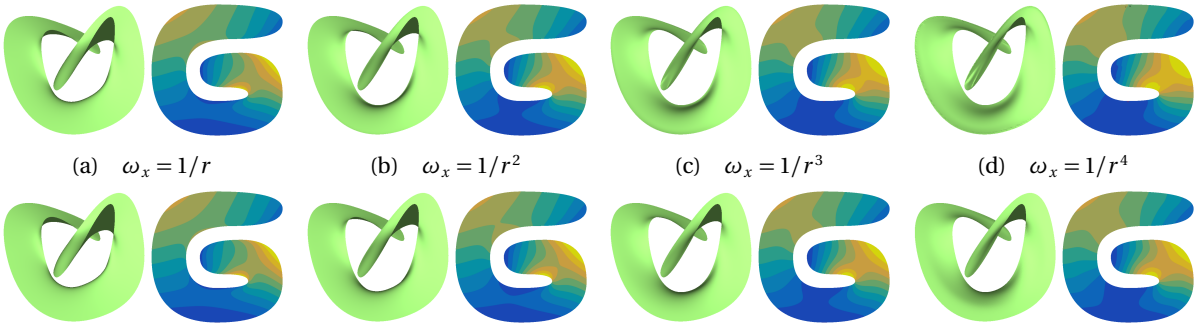
$$x=\int_0^{2\pi}\frac{p(t)}{r(t)^2}\,\mathrm{d}t\Big/\int_0^{2\pi}\frac{1}{r(s)^2}\,\mathrm{d}s,$$

we conclude that ∇F vanishes at $\phi=0$. As F is strictly convex, this is the unique minimum, and it follows from (14) that $\hat{\lambda}$ is the constant function with value $1/\phi_0$. Therefore, the barycentric kernel (17) corresponding to the weight function in (18) simplifies indeed to the Poisson kernel

$$\lambda(x,t)=\frac{1}{r(t)^2}\Big/\int_0^{2\pi}\frac{1}{r(s)^2}\,\mathrm{d}s.$$

3.3 Imitating the Poisson kernel

While identical to harmonic interpolants if Ω is a disk, the transfinite interpolants corresponding to the simple weight function in (18) are quite different from harmonic interpolants and more similar to PGW



(a) $\omega_x = 1/r$ (b) $\omega_x = 1/r^2$ (c) $\omega_x = 1/r^3$ (d) $\omega_x = 1/r^4$

Figure 4: Height interpolants for the bayrcentric kernel in (17) with different weight functions ω_x (top). Domain, boundary data, and colour-coding are as in Figure 1. Adding a constant to ω_x and using instead $\tilde{\omega}_x$ in (19) has a small effect on the interpolant (bottom), but greatly improves the numerical robustness.

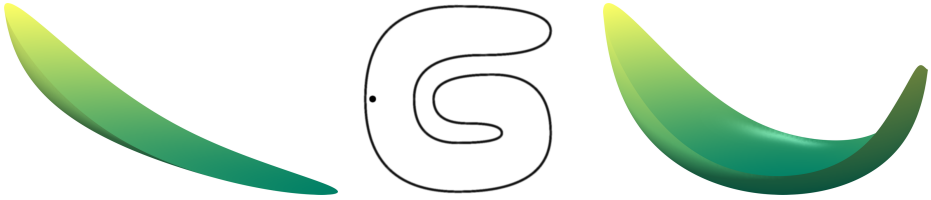


Figure 5: Plot of the function F in (16) over Φ for the indicated point $x \in \Omega$ (center) and the weight functions $\omega_x = 1/r^4$ (left) and $\tilde{\omega}_x$ (right).

interpolants for arbitrary domains (cf. Figure 4a, top with Figure 1b,c). However, our experiments indicate that the difference to harmonic interpolants decreases as we increase the power of r in the denominator of ω_x (see Figure 4b–d, top). Alas, for powers greater than three, numerical instabilities start to occur when evaluating the interpolants close to the boundary. The reason is that the minimum of the function F in (16) is located extremely close to the boundary of Φ for x near $\partial\Omega$ (see Figure 5, left), so that Newton's method has trouble converging to the correct solution.

An effective trick to remedy this problem is to add a suitable constant to the weight function. We found that replacing ω_x by

$$\tilde{\omega}_x = \omega_x + \int_a^b \omega_x(t)\,\mathrm{d}t \tag{19}$$

works well in practice (see Figure 4, bottom), as it has the effect of shifting the minimum of F towards the center of Φ (see Figure 5, right). In particular, based on our experiments (see Section 4), we propose to use the weight function

$$\omega_x(t) = \frac{1}{r(t)^4} + \int_a^b \frac{1}{r(s)^4}\,\mathrm{d}s, \tag{20}$$

which appears to give good and stable results that mimic the behaviour of harmonic interpolants for a wide range of domains.

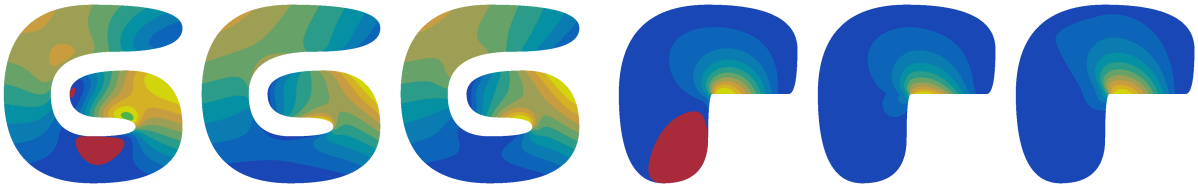
3.4 A modified mean value kernel

Another interesting kernel emerges if we try to simplify the minimization of the function F in (16). To this end, recall that the shape of the projected boundary curve $\hat{p}$ is a unit circle and that the projected boundary points can thus be expressed as $\hat{p}(t) = (\hat{p}_1(t), \hat{p}_2(t))^\mathsf{T} = (\cos\theta, \sin\theta)^\mathsf{T}$, where

$$\theta = \theta(t) = \arctan\frac{\hat{p}_2(t)}{\hat{p}_1(t)}. \tag{21}$$

If we now take as ω_x the speed at which $\hat{p}(t)$ travels along the unit circle S^1,

$$\omega_x(t) = \|\hat{p}'(t)\|,$$



(a) (b) (c) (d) (e) (f)

Figure 6: Height interpolants based on the mean value kernel (a,d), the modified mean value kernel in (22) (b,e), and the same kernel with ω_x replaced by ω_x^2 (c,f) for the domain and boundary data from Figure 1 (a–c) and a domain whose boundary contains a horizontal line segment (d–f). Note that the mean value and the modified mean value interpolants are identical inside $\ker\Omega$, but different outside (d,e), and that the C^0 artefact of the modified mean value interpolant, which is caused by the horizontal boundary line segment, vanishes if the weight function is squared (e,f).

then $\omega_x(t)\,\mathrm{d}t$ turns out to be equal to the (unsigend) arc length differential $|\mathrm{d}\theta|$. Consequently, ϕ_0 in (15) is equal to the length of the projected curve $\hat{p}$ and therefore equal to 2π for any $x\in\ker\Omega$ and greater than 2π, otherwise[1]. Moreover, if $x\in\ker\Omega$, then we can express F in (16), after changing the integration parameter from t to θ, as

$$F(\phi)=-\int_0^{2\pi}\log(\phi_0+\phi_1\cos\theta+\phi_2\sin\theta)\,\mathrm{d}\theta.$$

The gradient

$$\nabla F(\phi)=-\int_0^{2\pi}\frac{(\cos\theta,\sin\theta)^\mathsf{T}}{\phi_0+\phi_1\cos\theta+\phi_2\sin\theta}\,\mathrm{d}\theta$$

clearly vanishes at $\phi=0$ in this case, and using the same arguments as above we conclude that $\hat{\lambda}$ is a constant function, so that the barycentric kernel in (17) simplifies to

$$\lambda(x,t)=\frac{\omega_x(t)}{r(t)}\Big/\int_a^b\frac{\omega_x(s)}{r(s)}\,\mathrm{d}s. \tag{22}$$

Since θ in (21) can also be expressed in terms of $x=(x_1,x_2)^\mathsf{T}$ and the boundary point $p(t)=(p_1(t),p_2(t))^\mathsf{T}$ before projection as

$$\theta=\arctan\frac{p_2(t)-x_2}{p_1(t)-x_1},$$

a straightforward calculation (see Equation (10) in [7]) reveals that

$$\omega_x(t)=\left|\frac{\mathrm{d}\theta}{\mathrm{d}t}\right|=\frac{|(p(t)-x)\times p'(t)|}{r(t)^2}=|\mu(x,t)|\,r(t), \tag{23}$$

with $\mu(x,t)$ defined as in (5). Hence, the kernel in (22) is identical to the mean value kernel for any $x\in\ker\Omega$ (see Figure 6d,e), but, in contrast to the mean value kernel, non-negative for all other $x\in\Omega$.

However, the non-negativity comes at a price. As the absolute value function is only C^0 at the origin, it follows from (23) that ω_x, now seen as a function of x and t, is only C^0 at any pair (x,t) for which x lies on the line passing through $p(t)$ in the direction $p'(t)$, and the same is true for the kernel λ in (22). If the boundary of Ω is everywhere curved, so that $p''(t)\neq 0$ for all $t\in[a,b]$, then the coordinates of these C^0 artefacts vary in x as they vary in t. This implies that the C^0 artefacts themselves get smoothed out by integration with respect to t and the transfinite interpolant g in (2) is C^1 (see Figure 6b). But if the boundary of Ω contains a straight line segment with $p''(t)=0$ for $t\in[a',b']$, then the interpolant is only C^0 along the intersection of Ω with the line supporting this segment (see Figure 6e). To overcome this limitation, we can simply replace ω_x by ω_x^2 in (22), resulting in interpolants that are at least C^1 (see Figure 6c,f). However, according to our experiments, they are not as favourable as the ones obtained by using the weight function in (20).

[1]The *kernel* $\ker\Omega$ of a *domain* Ω (not to be confused with the *kernel functions* frequently used in this article) consists of all points $x\in\Omega$ for which the segment $[x,y]$, for any $y\in\Omega$, lies entirely in Ω. In particular, $\ker\Omega=\Omega$ for any convex domain.

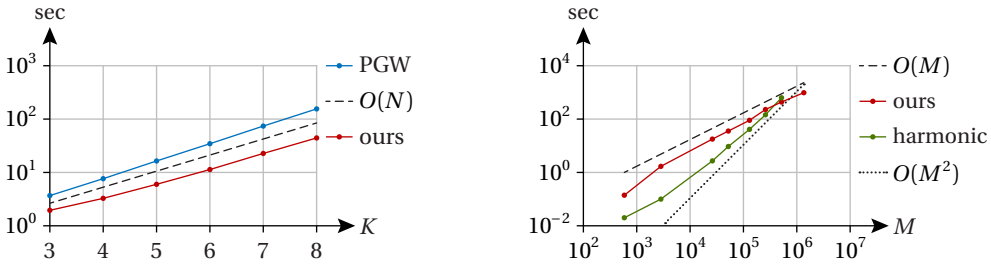


Figure 7: Running time in seconds for evaluating our interpolants and the PGW interpolant based on $N = 2^K 12$ boundary samples at 26000 domain points (left) and for computing the piecewise linear approximation of our interpolants and the harmonic interpolant over a domain triangulation with M vertices (right).

4 Results

4.1 Implementation

We have implemented the proposed framework for defining non-negative barycentric kernels in C++ on a Windows 10 laptop with a 2.4 GHz Intel i5-1135G7 processor and 16 GB RAM. In order to be able to compare our results to those given by the implementation of PGW interpolants [29], we decided to use quadratic B-splines for describing both the domain boundary and the boundary data. More precisely, given the n boundary control points $P_1, \dots, P_n \in \mathbb{R}^2$ and data control values $F_1, \dots, F_n \in \mathbb{R}$, we consider the B-spline curve and the B-spline function [34]

$$P(t) = \sum_{i=0}^{n+1} P_i N_i^2(t) \qquad \text{and} \qquad F(t) = \sum_{i=0}^{n+1} F_i N_i^2(t),$$

where $P_0 = P_n$, $P_{n+1} = P_1$, $F_0 = F_n$, and $F_{n+1} = F_1$, both over the uniform knot vector $(-2, -1, \dots, n+2)$. This guarantees the curve $p(t) = P(t)$ for $t \in [0, n]$ to enclose a C^1 domain Ω and the boundary data $f(p(t)) = F(t)$ for $t \in [0, n]$ to be periodic and C^1.

For the numerical evaluation of the interpolant g at any $x \in \Omega$, we replace the continuous parameter $t \in [0, n]$ by the N uniformly spaced parameters $\{t_k = kh : k = 0, \dots, N\}$ with $h = n/N$, and approximate any function that depends on t (e.g., ω_x or $\hat{\lambda}$) by the piecewise linear function that interpolates the function at $t_0, \dots, t_N$. Any integral of the form $\int_a^b \Psi(t)\,\mathrm{d}t$ (e.g., $\Psi = \log(\phi_0 + \phi^\mathsf{T} \hat{p})\omega_x$ in (16)) then simplifies to $h \sum_{k=1}^{N} \Psi(t_k)$, which is known as the trapezoid rule and converges at a rate of $O(h^2)$. Our experiments suggest that $N = 100n$ is sufficient for applications like image deformation (see Section 4.3). However, higher order numerical integration methods (e.g., Gaussian quadrature) could be used, if needed. For finding the minimum of the convex function F in (16), we follow Chang et al. [4] and use Newton's method with Armijo line search [1]. The convergence threshold ($\|\nabla F(\phi)\|_2 < 10^{-6}$) is usually reached after 3 to 8 iterations, but if x is close to $\partial\Omega$ (less than 10^{-3} times the diameter of Ω), then it may take up to 13 iterations.

4.2 Comparison

To compare the performance of our framework for defining and evaluating barycentric interpolants with PGW interpolants [29], we recall that the implementation of the latter approximates the boundary curve by the polygon P_K obtained after K times subdividing the initial boundary control polygon $P = [P_1, \dots, P_n]$ and then computes an approximation of the interpolant with respect to P_K. Since P_K has $2^K n$ vertices, it seems fair to compare the result with the one obtained by our implementation using $N = 2^K n$ boundary samples. Figure 7 (left) shows the result for the domain in Figure 1, which is defined by $n = 12$ control points. We observe that our method runs in $O(N)$ time and is faster than the PGW implementation, whose runtime complexity appears to be above $O(N)$, most likely because the search for pairs of edges with a non-vanishing visibility cone takes more than linear time.

For the comparison to harmonic interpolants, we use the classical finite element method that determines the piecewise linear approximation of the interpolant over a Delaunay triangulation $\mathcal{T}$ of the domain in Figure 1 with M interior vertices by solving a global linear system of size M. Since the system matrix is sparse, it is common to solve the system using a direct sparse LDL^T Cholesky factorization [18]. Our method

Figure 8: Example of a transfinite basis function for a non-convex domain defined by a quadratic B-spline with 20 control points, based on different barycentric kernels.

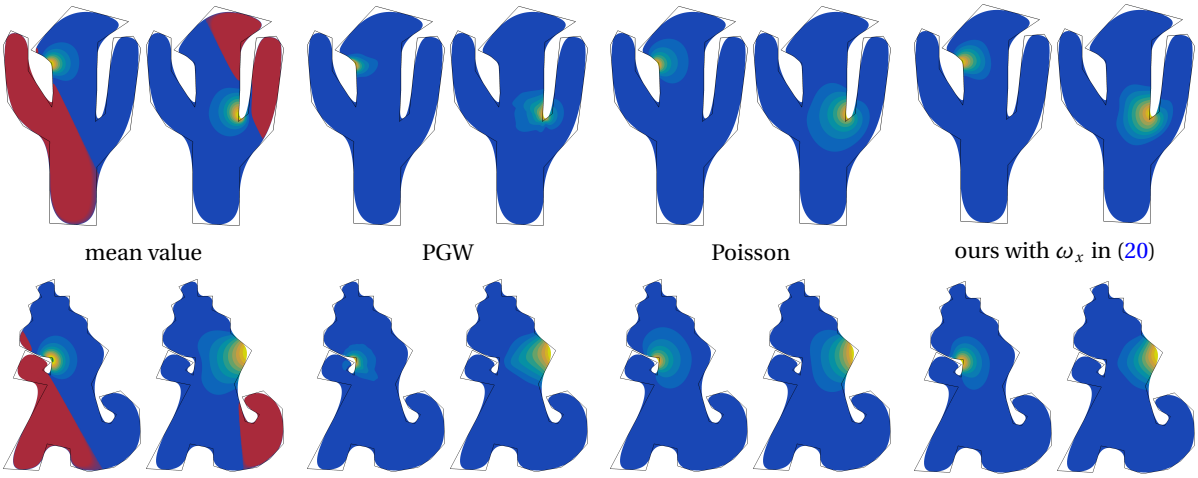


Figure 9: Examples of transfinite basis functions for two non-convex domains defined by quadratic B-splines.

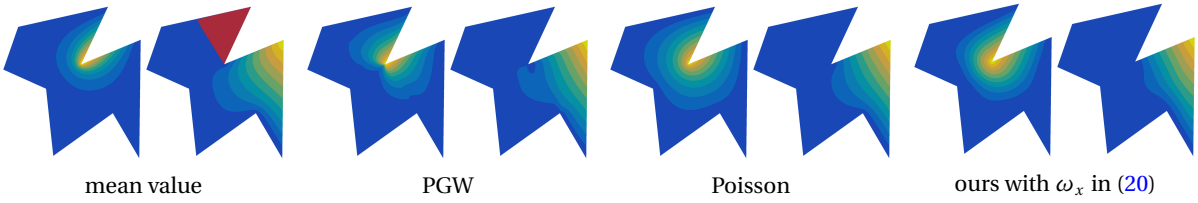


Figure 10: Examples of generalized barycentric coordinate functions associated with the vertices of a non-convex polygon, derived from different barycentric kernels.

(here with $N = 1200$) is instead local in the sense that it evaluates the interpolant at each domain point individually by solving a 2-dimensional non-linear optimization problem. Figure 7 (right) confirms that our method therefore computes the values of the interpolant at all vertices of $\mathcal{T}$ in $O(M)$ time, while the runtime complexity for determining the approximate harmonic interpolant is closer to $O(M^2)$, at least for large M. The break-even occurs at $M \approx 400000$. Note that the linear solver did not manage to handle cases with more than one million vertices, since it ran out of memory on our platform. For both comparisons reported in Figure 7, the choice of weight function ω_x had a negligible effect on the running times in our method.

Regarding the shape quality of the barycentric interpolants, we propose to use the weight function in (20) in our framework. Figures 1, 8, and 9 confirm that the corresponding interpolants are more similar to harmonic interpolants than mean value and PGW interpolants and in general have a smoother shape than the latter. The same observation holds for classical generalized barycentric coordinates (see Figure 10), which we can generate with the implemented B-spline model by simply duplicating control points.

4.3 Image deformation

For interactive applications, one advantage of the B-spline model is that we can precompute the values $G_1(x) = \hat{G}_1(x) + \hat{G}_{n+1}(x)$, $G_i(x) = \hat{G}_i(x)$, $i = 2, \ldots, n-1$, and $G_n(x) = \hat{G}_0(x) + \hat{G}_n(x)$, where

$$\hat{G}_i(x) = \int_0^n \lambda(x,t) N_i^2(t)\,\mathrm{d}t, \qquad i = 0, \ldots, n+1,$$

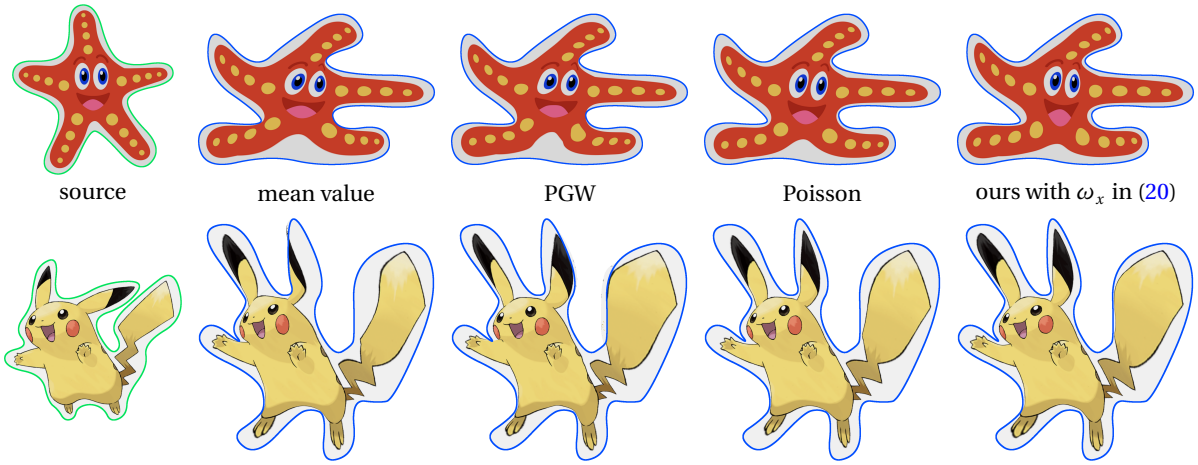


Figure 11: Deformation of a source image (left) bounded by a (green) quadratic B-spline curve with different barycentric kernels. While the deformations based on the mean value and PGW kernel tend to degenerate (see the "armpits" of the starfish) and may even fold over and "spill out" of the (blue) target curve (see the right ear and the tail of Pikachu), the deformations based on the Poisson and our novel barycentric kernel behave better and are quite similar to each other.

for a certain set of domain points $x \in \Omega$ (e.g., the vertices of a triangulation of Ω) and then evaluate the interpolant efficiently as

$$g(x) = \sum_{i=1}^{n} F_i G_i(x).$$

Figures 8 and 9 show some of these *transfinite basis functions* G_i for non-convex domains.

Replacing the data values $F_i \in \mathbb{R}$ with the control points $P_i' \in \mathbb{R}^2$ of a B-spline curve P' that encloses a *target domain* Ω', the transfinite interpolant g turns into a transfinite barycentric mapping from Ω to Ω'. This mapping can be used to deform a source image bounded by $\partial\Omega$, and the deformation can be controlled intuitively by changing the control points P_i'. Figure 11 shows some examples of this procedure for deformations based on different barycentric kernels.

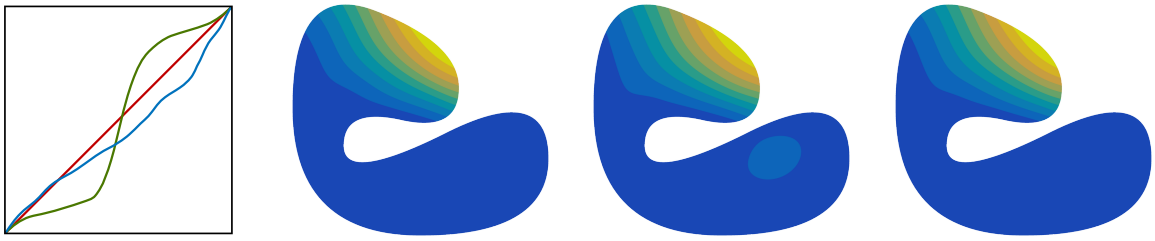
5 Conclusion

Although generalized barycentric coordinates have been studied intensively over the last two decades, remarkably few constructions of barycentric kernels for non-convex domains exist. With the framework in Section 2, it is now possible to generate and study a wide range of weighted maximum likelihood kernels. These kernels are barycentric by construction, non-negative for any positive weight function ω_x, and as smooth as ω_x, seen as a function in x. Although we did not manage to find a formal proof, it is reasonable to expect that these kernels are interpolatory, as long as the weight function ω_x does not annihilate the factor $1/r(t)$, that is, $\omega_x(t)$ must not converge to zero for $t = s$ as x converges to $p(s)$.

Interestingly, the factor $1/r(t)$ also plays a crucial rule for guaranteeing the interpolation property of the kernels in [35]. In fact, our kernels fit into their construction, if we interpret $\hat{\lambda}(t)\omega_x(t)\,\mathrm{d}t$ as the differential $\mathrm{d}\bar{P}_x$ of an auxiliary curve $\bar{P}_x$. The key difference between both approaches is that Schaefer et al. [35] get the linear reproduction property for free by using *polar duals* as auxiliary curves, while we have to solve for $\hat{\lambda}$, but the advantage of our method is that it handles arbitrary and not only convex domains.

Another difference of our framework to previous constructions is that all integrals depend on the parametric differential $\mathrm{d}t$ and not on the arc length differential $\mathrm{d}\theta$ of the unit circle around x. Hence, we can avoid calculating intersections between lines through x and $\partial\Omega$ and evaluate the interpolants with a simple numerical approximation scheme. However, as a consequence of this approach, most of our kernels and interpolants are parameterization-dependent, except for the modified mean value kernel with ω_x in (23). This can be seen as a limitation, but also as an additional degree of freedom in interactive applications. While we observed that the natural parameterization of quadratic B-splines usually gives very good results (see Figure 12), it remains future work to further explore this aspect.

The flexibility of our framework can be used to design kernels, for example, by using the weight function in (20), so that the interpolants are similar to harmonic interpolants, but in contrast to the latter, they can be



(a) (b) (c)

Figure 12: Transfinite basis function for our kernel with ω_x in (20) using the natural (red) parameterization of the boundary spline curve (a), after applying a monotonic (green) reparameterization (b), which creates a significant local maximum in the bottom right part of the domain, and after applying the arc-length (blue) reparameterization (c), which has little effect on the result in this example.

evaluated easily up to numerical precision at any $x \in \Omega$. Moreover, it is possible to evaluate the gradients of these interpolants by following the arguments in [4] and replacing all sums by integrals. It should also be straightforward to extend our construction to higher dimension $d > 2$. The only difference to the planar setting is that F in (16) is then a function in d variables, but it should still be easy to find the minimum of F with Newton's method.

Acknowledgements

This work was supported by the Swiss National Science Foundation (SNSF) under project number 188577. We thank the anonymous reviewers for their valuable comments and suggestions.

References

[1] L. Armijo. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16(1):1–3, Nov. 1966.

[2] A. Belyaev. On transfinite barycentric coordinates. In *Proceedings of the Fourth Symposium on Geometry Processing*, SGP '06, pages 89–99. Eurographics Association, Aire-la-Ville, June 2006.

[3] J.-P. Berrut. Rational functions for guaranteed and experimentally well-conditioned global interpolation. *Computers & Mathematics with Applications*, 15(1):1–16, 1988.

[4] Q. Chang, C. Deng, and K. Hormann. Maximum likelihood coordinates. *Computer Graphics Forum*, 42(5):Article e14908, 13 pages, Aug. 2023. [PDF]

[5] R. Chen and C. Gotsman. Complex transfinite barycentric mappings with similarity kernels. *Computer Graphics Forum*, 35(5):41–53, Aug. 2016.

[6] R. Chen and C. Gotsman. On pseudo-harmonic barycentric coordinates. *Computer Aided Geometric Design*, 44:15–35, May 2016.

[7] C. Dyken and M. S. Floater. Transfinite mean value interpolation. *Computer Aided Geometric Design*, 26(1):117–134, Jan. 2009.

[8] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, pages 173–182. Association for Computing Machinery, New York, Sept. 1995.

[9] Z. Farbman, G. Hoffer, Y. Lipman, D. Cohen-Or, and D. Lischinski. Coordinates for instant image cloning. *ACM Transactions on Graphics*, 28(3):Article 67, 9 pages, Aug. 2009.

[10] M. S. Floater. Parametrization and smooth approximation of surface triangulations. *Computer Aided Geometric Design*, 14(3):231–250, Apr. 1997.

[11] M. S. Floater. Mean value coordinates. *Computer Aided Geometric Design*, 20(1):19–27, Mar. 2003.

[12] M. S. Floater and K. Hormann. Barycentric rational interpolation with no poles and high rates of approximation. *Numerische Mathematik*, 107(2):315–331, Aug. 2007. [PDF]

[13] M. S. Floater, K. Hormann, and G. Kós. A general construction of barycentric coordinates over convex polygons. *Advances in Computational Mathematics*, 24(1–4):311–331, Jan. 2006. [PDF]

[14] M. S. Floater and J. Kosinka. Barycentric interpolation and mappings on smooth convex domains. In *Proceedings of the 14th Symposium on Solid and Physical Modeling*, SPM '10, pages 111–116. ACM, New York, Sept. 2010.

[15] M. S. Floater and C. Schulz. Pointwise radial minimization: Hermite interpolation on arbitrary domains. *Computer Graphics Forum*, 27(5):1505–1512, July 2008.

[16] A. Gillette, A. Rand, and C. Bajaj. Constructions of scalar and vector finite elements on polygonal and polyhedral meshes. *Computational Methods in Applied Mathematics*, 16(4):667–683, Oct. 2016.

[17] W. J. Gordon and J. A. Wixom. Pseudo-harmonic interpolation on convex domains. *SIAM Journal on Numerical Analysis*, 11(5):909–933, 1974.

[18] G. Guennebaud, B. Jacob, et al. Eigen 3.4.0. `http://eigen.tuxfamily.org`, Aug. 2021. [Online; accessed 17-May-2024].

[19] K. Hormann and M. S. Floater. Mean value coordinates for arbitrary planar polygons. *ACM Transactions on Graphics*, 25(4):1424–1441, Oct. 2006. [PDF]

[20] K. Hormann and N. Sukumar, editors. *Generalized Barycentric Coordinates in Computer Graphics and Computational Mechanics*. Taylor & Francis, CRC Press, Boca Raton, 2017. ISBN 978-1-4987-6359-2.

[21] K. Hormann and M. Tarini. A quadrilateral rendering primitive. In *Proceedings of Graphics Hardware*, GH '04, pages 7–14. Eurographics Association, Aire-la-Ville, Aug. 2004.

[22] P. Joshi, M. Meyer, T. DeRose, B. Green, and T. Sanocki. Harmonic coordinates for character articulation. *ACM Transactions on Graphics*, 26(3):Article 71, 9 pages, July 2007.

[23] T. Ju, S. Schaefer, and J. Warren. Mean value coordinates for closed triangular meshes. *ACM Transactions on Graphics*, 24(3):561–566, July 2005.

[24] J. Kosinka and M. Bartoň. Convergence of Wachspress coordinates: from polygons to curved domains. *Advances in Computational Mathematics*, 41(3):489–505, June 2015.

[25] J. Kosinka and M. Bartoň. Convergence of barycentric coordinates to barycentric kernels. *Computer Aided Geometric Design*, 43:200–210, Mar. 2016.

[26] X.-Y. Li, T. Ju, and S.-M. Hu. Cubic mean value coordinates. *ACM Transactions on Graphics*, 32(4):Article 126, 10 pages, July 2013.

[27] Z. Lin and R. Chen. Polynomial Cauchy coordinates for curved cages. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–8, Article 67, Tokyo, Dec. 2024.

[28] C. T. Loop and T. D. DeRose. A multisided generalization of Bézier surfaces. *ACM Transactions on Graphics*, 8(3):204–234, July 1989.

[29] J. Manson, K. Li, and S. Schaefer. Positive Gordon–Wixom coordinates. *Computer-Aided Design*, 43(11):1422–1426, Nov. 2011. The PGW source code is available at `http://josiahmanson.com/research/gordon_wixom_coords`.

[30] J. Manson and S. Schaefer. Moving least squares coordinates. *Computer Graphics Forum*, 29(5):1517–1524, July 2010.

[31] M. Meyer, H. Lee, A. Barr, and M. Desbrun. Generalized barycentric coordinates on irregular polygons. *Journal of Graphics Tools*, 7(1):13–22, 2002.

[32] E. Michel and J.-M. Thiery. Polynomial 2D Green coordinates for polygonal cages. In *SIGGRAPH 2023 Conference Proceedings*, pages 1–9, Article 23, Los Angeles, July 2023.

[33] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993.

[34] H. Prautzsch, W. Boehm, and M. Paluszny. *Bézier and B-spline Techniques*. Mathematics and Visualization. Springer, Berlin, 2002. ISBN 978-3-662-04919-8.

[35] S. Schaefer, T. Ju, and J. Warren. A unified, integral construction for coordinates over closed curves. *Computer Aided Geometric Design*, 24(8–9):481–493, Nov.–Dec. 2007.

[36] C. Schneider and W. Werner. Some new aspects of rational interpolation. *Mathematics of Computation*, 47(175):285–299, July 1986.

[37] N. Sukumar and E. A. Malsch. Recent advances in the construction of polygonal finite element interpolants. *Archives of Computational Methods in Engineering*, 13(1):129–163, Mar. 2006.

[38] J.-M. Thiery, P. Memari, and T. Boubekeur. Mean value coordinates for quad cages in 3D. *ACM Transactions on Graphics*, 37(6):Article 229, 14 pages, Dec. 2018.

[39] E. L. Wachspress. *A Rational Finite Element Basis*, volume 114 of *Mathematics in Science and Engineering*. Academic Press, New York, 1975. ISBN 978-0-12-728950-2.

[40] J. Warren, S. Schaefer, A. N. Hirani, and M. Desbrun. Barycentric coordinates for convex sets. *Advances in Computational Mathematics*, 27(3):319–338, Oct. 2007.