Harmonic Coordinates for Character Articulation

Pushkar Joshi   Mark Meyer   Tony DeRose
Brian Green   Tom Sanocki
Pixar Animation Studios
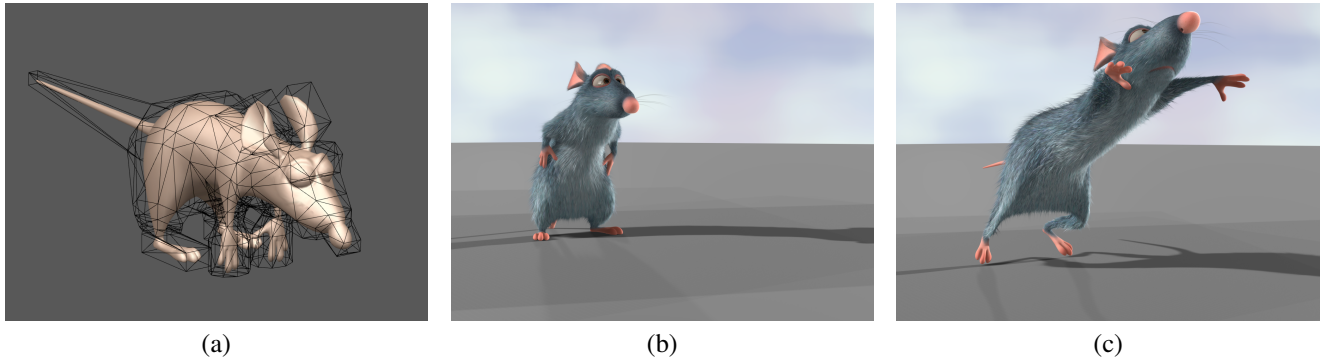
(a)   (b)   (c)

Figure 1: *A character posed using using harmonic coordinates. (a) The character and cage (shown in black) at bind-time; (b) and (c) are two poses from an animated clip. All images © Disney/Pixar.*

Abstract

In this paper we consider the problem of creating and controlling volume deformations used to articulate characters for use in high-end applications such as computer generated feature films. We introduce a method we call harmonic coordinates that significantly improves upon existing volume deformation techniques. Our deformations are controlled using a topologically flexible structure, called a cage, that consists of a closed three dimensional mesh. The cage can optionally be augmented with additional interior vertices, edges, and faces to more precisely control the interior behavior of the deformation. We show that harmonic coordinates are generalized barycentric coordinates that can be extended to any dimension. Moreover, they are the first system of generalized barycentric coordinates that are non-negative even in strongly concave situations, and their magnitude falls off with distance as measured within the cage.

CR Categories: I.3.5 [Computational Geometry and Object Modeling]: Geometric algorithms, languages, and systems.

Keywords: Barycentric coordinates, mean value coordinates, free form deformations, rigging.

1 Introduction

Character articulation, sometimes called rigging, is an important component of high-end animation systems of the kind used in fea-

ture film production. Modern high-end systems, most notably SoftImage XSI® and Maya®, offer a variety of articulation methods such as enveloping [Lewis et al. 2000], blend shapes [Joshi et al. 2006], and chains of arbitrary deformations. In the realm of deformations, free-form deformations as introduced by Sederberg and Parry [1986] are particularly popular for a number of reasons. First, they offer smooth and intuitive control over the motion of the character using only a few parameters, namely, the locations of the free-form lattice control points. Second, there are virtually no restrictions on the three-dimensional model of the character — the only requirement is that the character model is completely enclosed by the control lattice.

However, free-form deformation has some drawbacks. Articulating a multi-limbed character is best accomplished using a lattice that conforms to the geometry of the character. However, given the topological rigidity of a lattice, it is often necessary to combine several overlapping lattices, and each of the lattices possess interior points that can be difficult and annoying to articulate. The problem of multiple overlapping lattices was addressed by MacCracken and Joy [1996] where lattices were generalized to arbitrary volume meshes, but their method still requires the introduction and articulation of numerous interior control points.

Ju et al [2005] introduced a promising new approach that is even more topologically flexible, wherein the character to be deformed (henceforth called the *object*) is positioned relative to a coarse closed triangular surface mesh (henceforth called the *cage*). The object is then "bound" to the cage by computing a weight $g_i(p)$ of each cage vertex C_i evaluated at the position of every object point p. As the cage vertices are moved to new locations C_i', the deformed points p' are computed from

$$p' = \sum_i g_i(p)C_i'. \qquad (1)$$

An example is shown in Figure 2(b). The weight functions $g_i(p)$ used by Ju et al. are known as mean value coordinates [Floater 2003; Floater et al. 2005; Ju et al. 2005]. Mean value coordinates are a form of generalized barycentric coordinates that have a number of uses, but they are particularly interesting in the context of character articulation because:

Bind

Mean Value

Harmonic

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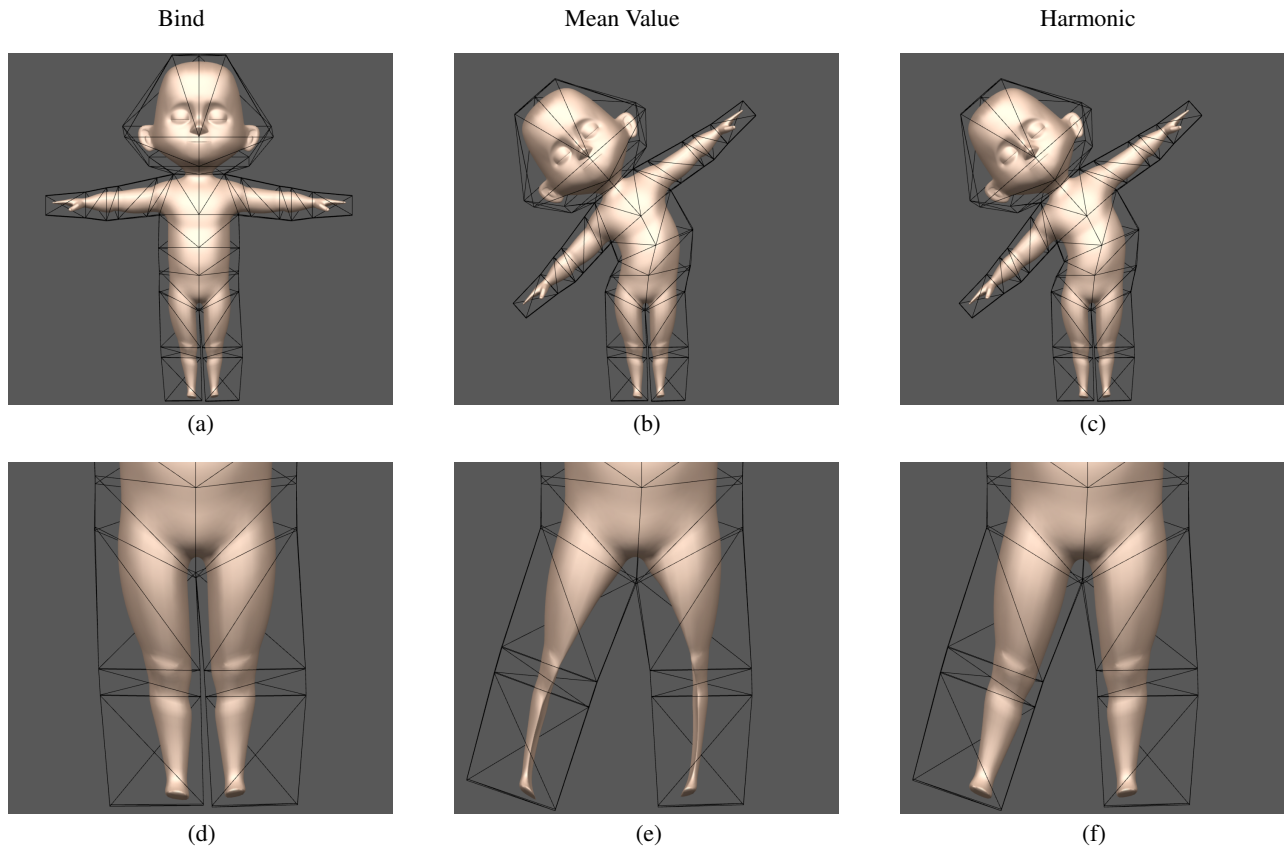
(d)   (e)   (f)

Figure 2: *A comparison between deformations based on mean value and harmonic coordinates. The first row shows a torso bend and the second row shows a leg bend for a typical character. The* left *column shows the cage (shown in black) and object (shown in beige). The* middle *column shows modified cages and the corresponding deformed objects using mean value coordinates. The* right *column shows modified cages and deformed objects using harmonic coordinates. Notice that the two methods perform similarly for the torso bend. However, in strongly concave situations such as the legs, harmonic coordinates produced significantly more pleasing results.*

- The cage that controls the deformation can be any closed triangular surface mesh, so there is a great deal of topological and geometric flexibility when designing the cage.

- The coordinates are smooth, so the deformation is smooth.

- The coordinates reproduce linear functions, so the object doesn't "pop" when it is bound. That is, the coordinates are such that setting C_i' to C_i in Equation 1 results in p' reducing to p.

However, mean value coordinates have a drawback for use in character articulation, as illustrated by the bipedal character shown in Figure 2. Notice how the modified cage vertices for the leg on the left in Figure 2(e) significantly influence the position of object points in the leg on the right. This occurs because mean value coordinates are based on Euclidean (straight-line) distances between cage vertices and object points. Since the distance between the modified cage vertices and the object points in the leg on the right are relatively small in the bind pose, the influence is relatively large. Notice too that the displacement of those object points is in a direction opposite to the displacement of the cage vertices. This occurs because the mean value coordinates are negative, as shown in Figure 4(b). This undesirable movement is particularly striking in interactive use, as demonstrated in accompanying video (see [Joshi et al. 2007]). Such behavior is unacceptable for the articulation of characters in feature film production.

The undesirable behavior illustrated above occurs because mean value coordinates lack two properties that are essential for high-end

character articulation; namely:

- Interior locality: Informally, the coordinates should fall off as a function of the distance between cage vertices and object points, where distance is measured *within the cage*.

- Non-negativity: As illustrated in Figures 2(e) and 4(b), if an object point whose coordinate relative to a cage vertex is negative, the object point and cage vertex will move in opposite directions. To prevent this unintuitive behavior, we seek coordinates that are guaranteed to be non-negative on the interior of the cage, even in strongly concave situations.

In this paper, we show that coordinates possessing the these two critical properties can be produced as solutions to Laplace's equation. Since solutions to Laplace's equation are generically referred to as harmonic functions, we therefore call these coordinates *harmonic coordinates*, and the deformations they generate *harmonic deformations*.[1]

Unlike mean value coordinates, harmonic coordinates do not, in general, possess a closed form expression. Instead, they must be approximated using a numerical solver. There are two potential drawbacks of a numerical solution: time and accuracy. For character articulation, as the animator moves the cage vertices, the object

---

[1]Since each component of a harmonic deformation is a harmonic function, many texts refer to such deformations as harmonic maps. We prefer the term harmonic deformation because of the context in which they're used in this paper.

must deform in real-time. For such fully interactive deformations the coordinates should be computed as a pre-process, prior to any user interaction. This precomputation is necessary even if a closed form exists, as noted by Ju et al. [2005]. Given that coordinates must be precomputed whether or not a closed form exists, the time required to run the harmonic coordinate solver is acceptable (see Sections 6 and 7). The key issue regarding accuracy is whether it is possible to reproduce linear functions using reasonable time and space. For harmonic coordinates this is in fact the case, again as shown in Section 6.

To put our analysis of previous work and contributions in context, we reiterate the problem we are trying to solve and why. Our goal is to develop a topologically flexible "cage-based" method of controlling volume deformations. We specifically do not seek a direct manipulation method, such as methods based on local differential coordinates (c.f. Sorkine [2006] and Sumner et al. [2005]), though we share some of the same mathematical underpinnings. While such direct manipulation methods have produced impressive results (particularly under extreme deformations), cage-based methods allow us to decouple the geometry being articulated (the cage) from the geometry of the character. This decoupling is desirable for several reasons. First, our character models generally do not consist of a single mesh. Rather, they are typically described using a rather large collection of modeling primitives, including separate meshes for skin, teeth, and clothing, as well as non-mesh primitives such as spheres for eyeballs. Cage-based methods are ideal for coherently deforming such large and diverse collections of primitives, whereas direct manipulation mesh deformation techniques address the deformation of individual meshes. Second, the decoupling allows us to reuse the articulation as the character geometry changes. We can also reuse articulation of the cage to articulate the full version of the character, as well as low detail versions for faster preview and rendering. Such reuse of articulation is significant since, at least at our studio, articulation typically takes an order of magnitude longer to author than static geometry.

1.1 Contributions

In this paper we offer the first cage-based deformation technique guaranteeing that the influence of each cage vertex is non-negative and falls off with distance as measured within the cage (see Section 2). Moreover, it is the first cage-based deformation technique that enables users to refine the deformation behavior over the interior of the cage by optionally adding vertices, edges, and faces (see Section 3). Taken together, these contributions comprise a powerful new deformation method for use in high-end character articulation.

1.2 Previous work

Our work lies at the intersection of three large bodies of literature: volume deformations, generalized barycentric coordinates, and harmonic functions. The number of related papers is therefore very large. In this section we summarize the most closely related work.

In the area of volume deformations, as mentioned above, Sederberg and Parry [1986] introduced a method based on three dimensional lattices. MacCracken and Joy [1996] subsequently used recursive subdivision to achieve more topological flexibility. Similar in spirit to our application where a coarse cage is used to pose a character, Capell et al. [2002] use the method of MacCracken and Joy to transfer motion of a coarse subdivision lattice to the geometry of a character. Use of subdivision is crucial for their purposes since refinement is used to create the hierarchical basis they use to accelerate the simulation of elastic deformation. However, we wish to avoid subdivision-based methods because, like the original free-form deformation, they require the interior of the cage to be meshed, and they contain interior vertices that must be articulated. This makes authoring of cage articulation more difficult than cage-based schemes like ours that do not require interior points. A direct manipulation volume deformation method was introduced by Igarashi et al. [2005]. The deformations they produce are intuitive, but their method does not employ a cage, so the reuse advantages mentioned above are not realized.

In the area of barycentric coordinates, Ju et al. [2005] used generalized barycentric coordinates, specifically, mean value coordinates (see also Floater et al.[2005]) to define deformations controlled by a surface mesh rather than a three dimensional lattice. The problem of generalizing barycentric coordinates is rich and has received considerable attention in recent years ([Wachpress 1975], [R.Sibson 1981], [Loop and DeRose 1989], [Warren 1996], [Meyer et al. 2002], [Floater 2003]). Ju et al. [2005] provide a good overview. Our method improves on previous methods by using a different generalized barycentric coordinate formulation, one based on Laplace's equation.

In the area of harmonic functions, we note that Laplace's Equation, harmonic functions, and harmonic maps have often been mentioned in previous constructions of generalized barycentric coordinates in two dimensions. For instance, the "cotangent weights" of Pinkall and Polthier [1993] and Meyer et al. [2002] can be derived from piecewise linear discretizations of Laplace's equation. Similarly, Floater's construction of mean value coordinates [Floater 2003] was motivated by the mean value theorem for harmonic functions. Independent of our work, Floater and co-workers [2006] recently observed that solutions to Laplace's equation in two dimensions could be used as generalized barycentric coordinates. They did not pursue the observation because their objective was to find a closed-form formulation. However, for character articulation, non-negativity and interior locality are far more important than closed-form expressions. Harmonic coordinates are the only coordinates we are aware of that possess these crucial properties.

Harmonic functions have also been used to define surface deformations. For example, Zayer et al. [2005] describe a method for direct mesh manipulation based on properties of two-dimensional harmonic functions. Similar to our method, they construct a basis function per controllable vertex, where each function is the result of solving Laplace's equation subject to Kronecker delta type constraints. Shi et al. [2006] extended the method of Zayer et al. by providing a multigrid-based solver. In some sense, our method can be seen as combining the benefits of Zayer et al. and Shi et al. with the those of Ju et al.. This combination is far from trivial however since we expose the deep connection between generalized barycentric coordinates (in any dimension) with harmonic functions. We use this connection to create the first cage-based deformation method with non-negative basis functions and the ability to specify optional and flexible interior controls.

Finally, harmonic functions have been used in scattered data interpolation. For instance, the membrane and thin-plate splines introduced by Duchon [1977] are harmonic and bi-harmonic functions, respectively, that are used to construct a smooth interpolant (Wahba [1990]). These splines are often used as radial basis functions to define coordinates over the interior of a given space. For instance, Carr et al. [2001] used polyharmonic spline functions to reconstruct smooth 2-manifold surfaces from point clouds, and Choe et al. [2001] used similar splines as a basis function for interpolating marker motion on a human face model. These methods are all designed to interpolate at a *finite* set of points, whereas our method is designed to interpolate over the entire *continuous* boundary of the cage.
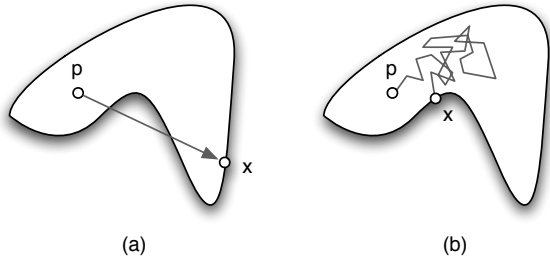
Figure 3: *Mean value vs harmonic interpolation. (a) The straight-line paths corresponding to mean value interpolation. (b) The Brownian paths corresponding to harmonic interpolation.*

2 Theory

In this section we formalize the discussion of the previous section and prove that harmonic coordinates possess all the properties necessary for use in high-end character articulation.

We begin by considering the construction of mean value coordinates as described in [Floater et al. 2005] and [Ju et al. 2005]. They derive coordinates starting with a "mean value interpolant" of a function f defined on a closed boundary. To compute an interpolant value for each interior point p, they consider each point x on the boundary, multiply $f(x)$ by the reciprocal distance from x to p, then average over all x (see Figure 3(a)). This definition makes it clear that mean value coordinates involve straight-line distances irrespective of the visibility of x from p. For character articulation, the cage often has large concavities, and a more useful interpolant would respect visibility of an cage vertex from an object point. To construct such an interpolant, we can average not over all straight-line paths, but rather over all *Brownian paths* leaving p, where the value assigned to each path is the value of f at the point the path first hits the cage boundary (see Figure 3(b)). At first, this interpolant seems intractable to compute. However, a famous result from stochastic processes (c.f. [Port and Stone 1978], [Bass 1995]) states that the interpolant thus produced (in any dimension) in fact satisfies Laplace's equation subject to the boundary conditions given by f [2]. Therefore, we can obtain improved generalized barycentric coordinates from a numerical solution of Laplace's equation in the cage interior.

More formally, let a cage C be a polyhedron in d dimensions – that is, a closed (not necessarily convex) volume with a piecewise linear boundary. In two dimensions, a cage is a region of the plane bounded by a closed polygon (such as the ones shown in Figure 4), and in three dimensions a cage is a closed region of space bounded by planar (though not necessarily triangular) faces. For each of the vertices C_i of the cage, we seek a function $h_i(p)$ defined on C subject to the following conditions (listed in the order that they are proved later):

1. Interpolation: $h_i(C_j) = \delta_{i,j}$.

2. Smoothness: The functions $h_i(p)$ are at least C^1 smooth in the interior of the cage.

3. Non-negativity: $h_i(p) \geq 0$, for all $p \in C$.

4. Interior locality: We quantify the notion of interior locality introduced in Section 1 as follows: interior locality holds, if, in addition to non-negativity, the coordinate functions have no interior extrema.

5. Linear reproduction: Given an arbitrary function $f(p)$, the

---

coordinate functions can be used to define an interpolant $H[f](p)$ according to:

$$H[f](p) = \sum_i h_i(p)f(C_i) \qquad (2)$$

Following Ju et al. [2005], we require $H[f](p)$ to be exact for linear functions. As shown by Ju et. al, taking $f(p) = p$ means that

$$p = \sum_i h_i(p)C_i \qquad (3)$$

which is the "non-popping" condition mentioned in Section 1.

6. Affine-invariance: $\sum_i h_i(p) = 1$ for all $p \in C$.

7. Strict generalization of barycentric coordinates: when C is a simplex, $h_i(p)$ is the barycentric coordinate of p with respect to C_i.

Mean value coordinates possess all but two of these properties, namely, non-negativity and interior locality. We claim that coordinate functions satisfying all seven properties can be obtained as solutions to Laplace's equation

$$\triangledown^2 h_i(p) = 0, \quad p \in Int(C) \qquad (4)$$

if the boundary conditions are appropriately chosen.

To gain some insight into how the boundary conditions are determined, we consider first the construction of harmonic coordinates in two dimensions. It will then be clear how the construction generalizes to d dimensions. For reasons that will soon become apparent, the appropriate boundary conditions for $h_i(p)$ in two dimensions are as follows. Let ∂p denote a point on the boundary ∂C of C, then

$$h_i(\partial p) = \phi_i(\partial p), \quad \text{for all } \partial p \in \partial C \qquad (5)$$

where $\phi_i(\partial p)$ is the (univariate) piecewise linear function such that $\phi_i(C_j) = \delta_{i,j}$. For example, if C is the cage shown in Figure 4(a), then $\phi_i(\partial p)$ is the piecewise linear function defined on the edges $e_1, ..., e_{19}$ such that $\phi_i(C_j) = \delta_{i,j}$, for $i, j = 1, ..., 19$.

We now show that functions satisfying Equation 4 subject to Equation 5 possess the properties enumerated above.

1. Interpolation: by construction $h_i(C_j) = \phi_i(C_j) = \delta_{i,j}$.

2. Smoothness: Away from the boundary harmonic coordinates are solutions to Laplace's equation, and hence they are C^∞ in the cage interior. On the boundary they are only as smooth as the boundary conditions, and hence are only guaranteed to be C^0 on the boundary.

3. Non-negativity: harmonic functions achieve their extrema at their boundaries. Since boundary values are restricted to $[0, 1]$, interior values are also restricted to $[0, 1]$, and are therefore non-negative. An example is shown in Figure 4(c).

4. Interior locality: follows from non-negativity and the fact that harmonic functions possess no interior extrema.

5. Linear reproduction: Let $f(p)$ be an arbitrary linear function. We need to show that $H[f](p) = f(p)$, where $H[f](p)$ is defined as in Equation 2. We begin by establishing that $H[f](p) = f(p)$ everywhere on the boundary of C. If ∂p is a point on the boundary of C, then by construction

$$H[f](\partial p) = \sum_i h_i(\partial p)f(C_i) = \sum_i \phi_i(\partial p)f(C_i) \qquad (6)$$

The functions $\phi_i(\partial p)$ are the univariate linear B-spline basis functions (commonly known as the "hat function" basis), which are capable of reproducing all linear functions on ∂C (in fact, they reproduce all piecewise linear functions on ∂C).
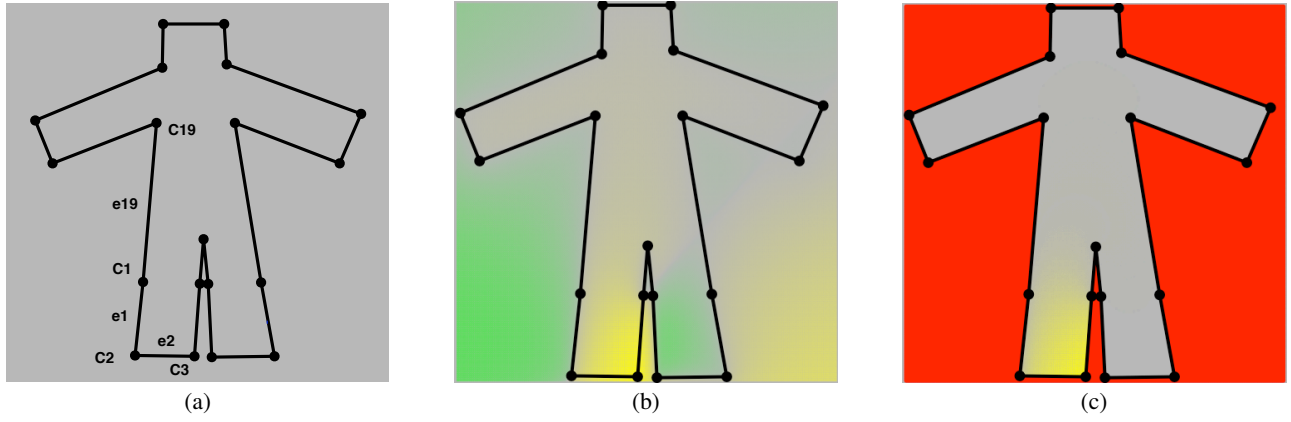
Figure 4: *A comparison of coordinate functions for a concave cage. (a) A 2D cage with vertices $C_1,...,C_{19}$; (b) the value of the mean value coordinate for C_3 (yellow indicates positive values, green indicates negative values); (c) the value of the harmonic coordinate for C_3 (red denotes the exterior of the cage where the function is undefined). To accentuate values near zero, intensities of yellow and green are proportional to the square root of the coordinate function value. The significant influence of the position of C_3 on object points in the leg on the right is indicated by the presence of green in the right leg of (b). The corresponding influence in (c) is essentially zero.*

Next we extend the result to the interior of C. Since $f(p)$ is linear, all second derivatives vanish. That is, $\bigtriangledown^2 f(p) = 0$ and $f(p)$ satisfies Laplace's equation on the interior of C. Since $H[f](p)$ is a linear combination of harmonic functions, it also satisfies Laplace's equation on the interior.

$f(p)$ and $H[f](p)$ satisfy the same boundary conditions and are both solutions to Laplace's equation. Therefore, by uniqueness of solutions to Laplace's equation, they must be the same function.

6. Affine invariance: affine invariance follows immediately from the linear reproduction property by substituting $f(p) := 1$ in Equation 2.

7. Strict generalization of barycentric coordinates. If the cage C consists of a single triangle, the harmonic coordinates reduce to simple barycentric coordinates. Let $\beta_j(p)$ denote the barycentric coordinates of p with respect to the triangle. To establish that $h_j(p) = \beta_j(p)$, note that $\beta_j(p)$ is a linear function, so we can use the linear reproduction property above by taking $f(p) = \beta_j(p)$:

$$
\begin{aligned}
\beta_j(p) &= H[\beta_j](p) \\
&= \sum_i h_i(p)\beta_j(C_i) \\
&= \sum_i h_i(p)\delta_{i,j} \\
&= h_j(p)
\end{aligned}
$$

It is straightforward to generalize harmonic coordinates from two to d dimensions. First, consider harmonic coordinates in one dimension: the cage is a line segment bounded by two vertices C_0 and C_1, and Laplace's equation reduces to

$$
\frac{d^2 h_i(p)}{dp^2} = 0. \qquad (7)
$$

Thus, $h_i(p)$ is a linear function, and the proper (zero dimensional) boundary conditions come from the interpolation property: $h_i(C_j) = \delta_{i,j}$.

With this insight, we can repose the next higher (two) dimensional construction for the cage C. Start with the interpolation conditions $h_i(C_j) = \delta_{i,j}$. This determines the coordinates on the 0-dimensional

facets (the vertices) of C. Next, extend the coordinates to the 1-dimensional facets (the edges) of C using the one dimensional version of Laplace's equation. Finally, extend them to the two dimensional facets (the interior) of C using the two dimensional version of Laplace's equation.

The extension to three and higher dimensions follows immediately: the harmonic coordinates $h_i(p)$ for a d dimensional cage C with vertices C_i, are the unique functions such that:

- $h_i(C_j) = \delta_{i,j}$.

- On every facet of dimension $k \leq d$, the k dimensional Laplace equation is satisfied.

Using mathematical induction, we can prove that these d dimensional harmonic coordinates possess the above required properties. The induction is on the facet dimension, starting with the 1-facets as the base case. The proofs given above for two dimensions are in fact valid in any dimension k assuming that linear reproduction is achieved on the $k - 1$ facets. These proofs therefore serve as the inductive step.

Having defined coordinates in this way, by construction we have the following additional property that is shared by barycentric coordinates as well as Warren's construction [1996]:

- Dimension reduction: d dimensional harmonic coordinates, when restricted to a $k < d$ dimensional facet, reduce to k dimensional harmonic coordinates.

For example, a three dimensional cage bounded by triangular facets possesses harmonic coordinates that reduce to barycentric coordinates on the faces. Similarly, a dodecahedral cage will have 3D harmonic coordinates that reduce to 2D harmonic coordinates on its pentagonal faces.

In the next two sections we describe extensions to the basic construction of harmonic coordinates that dramatically enhance the controllability and range of practical application in character articulation.

3 Interior Control

Harmonic deformations as formulated thus far are completely defined by the cage boundary, which is ideal in many instances. However, it is sometimes helpful to give artists optional control over the
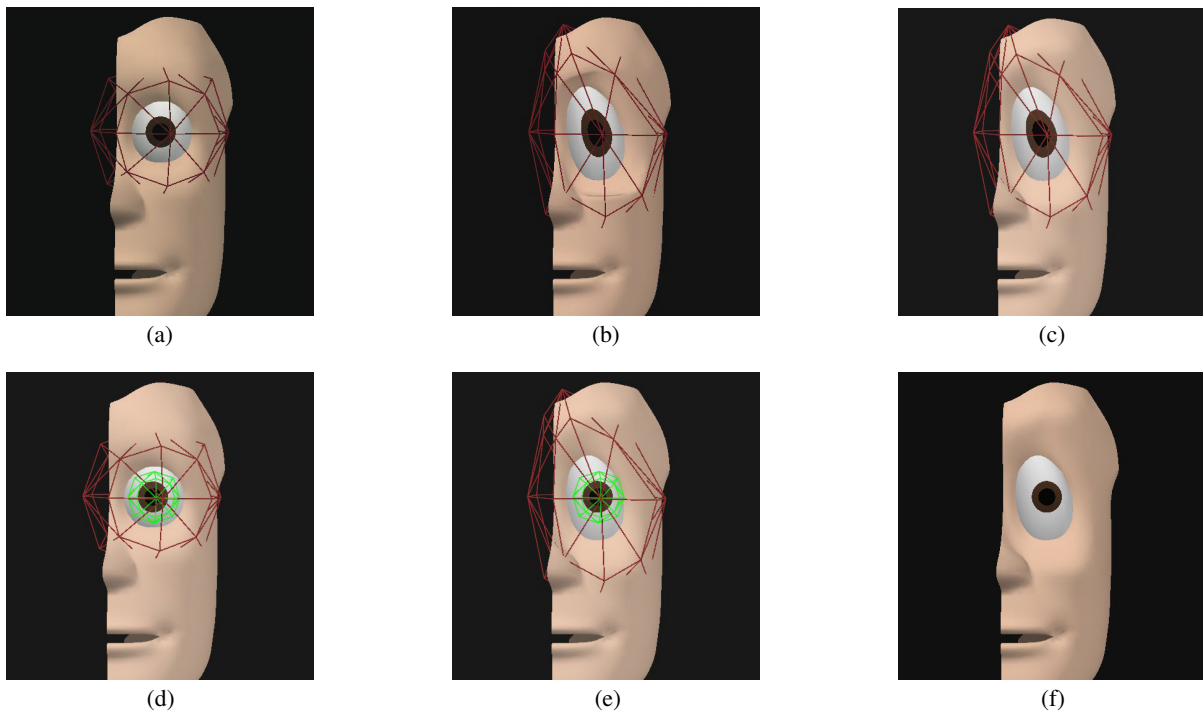
Figure 5: *Interior control in three dimensions. (a) A character face including an eye ball and a simple cage. The cage will be used to reposition all the object points within the cage. The object points serve as control points for two subdivision surfaces, one for the face, and one for the eye ball. This is the pose used for binding. (b) The deformed facial region. Since the deformation applies only to points of the face inside the cage, the control mesh for the facial surface is far from smoothly varying at the edge of the cage. (c) Point weighting is used to smooth the transition from the cage interior to its exterior. (d) The cage from (a) has been augmented with a interior triangulated sphere (shown in green) to control the shape of the iris. (e) The iris in the deformed facial region is kept round using the interior sphere. (f) The deformed limit subdivision surfaces.*

deformation in the cage interior. We can easily provide such control by adding cage elements (vertices, edges and faces) where they are needed.

As an example of the use of such interior control, consider the situation shown in Figure 5, where the region around a character's eye is deformed. The object points serve as control points for subdivision surface control meshes, in this case for two surfaces: one for the face, and one for the eye ball. All object points within the cage shown in Figure 5(a) are affected by the cage deformation. Since the object points just inside the cage would be deformed fully, and those just outside the cage would be left unchanged, the subdivision meshes would not be smooth in more extreme poses (see Figure 5(b)). This is a common issue in character articulation, so character rigging systems, including ours, provides a facility similar to cluster weighting as found in Maya. In this situation the weights are used to feather out the influence of the deformer near its boundary. Proper weighting together with harmonic deformation results in poses like the one shown in Figure 5(c). Because of the convex nature of the cage, the example up to now could have been created using mean value coordinates. However, eyes are an important part of story telling and our directors generally require the iris and pupil to stay round. Maintaining roundness of the iris and pupil while the surrounding geometry deforms has been extremely challenging in the past, requiring the addition of a second "correction" deformation whose purpose is to counter deform the iris and pupil. This process is delicate, time consuming, and approximate.

However, using harmonic deformations, the desired affect can be easily accomplished by augmenting the cage with an interior triangulated sphere (shown in green in Figure 5(d)) that surrounds the iris and pupil. By keeping the interior green component of the

cage round, the iris and pupil stay round as the geometry around them deforms, as shown in Figure 5(e) and (f). This kind of precise interior control is simply not available using other deformation methods. An animated version of this example is available on the accompanying video (see [Joshi et al. 2007]).

Harmonic coordinates are assigned to interior vertices in just the same way they are assigned to boundary vertices of the cage. Interior edges and faces are handled identically as well. In fact, our solver makes no distinction between boundary and interior components.

The interior controls in the previous example formed a manifold, but this need not be the case — it is sufficient for the interior of the cage to form what is known as a linear cell complex. Intuitively, a linear cell complex is a collection of "cells" of various dimensions, 0-cells (vertices), 1-cells (linear edges), 2-cells (planar faces), and so on, with the property that the intersection of any two cells is either empty or is another cell in the collection.

The two dimensional example shown in Figure 6 was chosen to better illustrate the generality of the method, though the same ideas apply equally well to three (and higher) dimensions. Figure 6(a) shows the bind-time configuration a biped character. In Figure 6(b), the cage vertices near the head, shown in yellow, are moved after binding, resulting in a deformation that influences the head, neck and shoulder regions most, but it also influences the under arms. This may or may not be considered desirable, depending on the needs of the articulator. If the articulator wishes to isolate the under arms from motion of the head vertices of the cage, he/she can add additional interior vertices and edges to the cage, as shown in Figure 6(c). This results in more localized deformations as shown
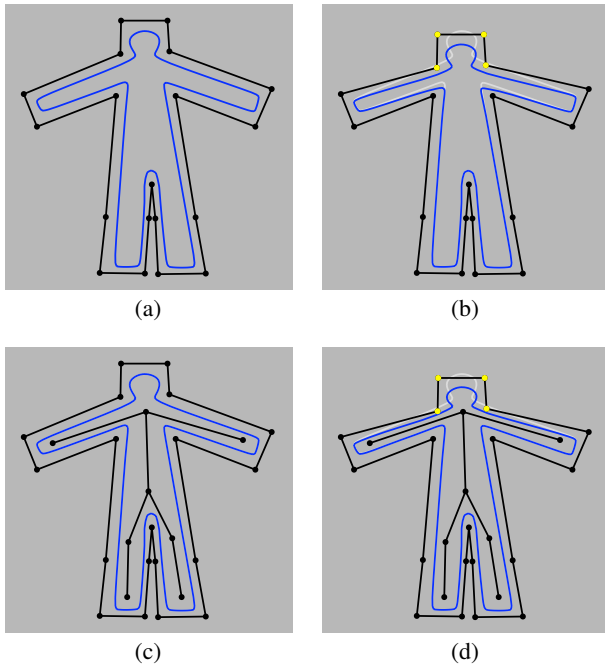
(a) (b)

(c) (d)

Figure 6: *Interior control using a linear cell complex. (a) An object and cage with no interior controls at bind-time. (b) the object deformed by the cage in (a). (c) An object and cage with interior controls at bind-time. (d) the object deformed by the cage in (c).*

Property	Fig 1	Fig 2	Fig 5	Fig 6
No. cage vertices	325	112	39	27
No. object points	9775	8019	269	136
Grid resolution — s	5	5	4	5
Solve time (sec.)	57.4	17.6	5.85	0.83
Pose time (sec.)	0.111	0.026	0.0001	0.0007
Solution grid size (MB)	9.2	3.7	0.32	0.048

Table 1: *Resource requirements for the examples in the paper. Solve times indicate the time to precompute harmonic coordinates for the cage interior, and the pose time indicates the time to compute new object positions during interactive posing. Timings in columns 1 and 2 were performed single threaded on a Dual Xeon™ 3.40GHz CPU with 3.8 GB of ram. Timings in columns 3 and 4 were performed single threaded on a PowerMac™ G5 Quad 2.5GHz with 1.5GB of ram.*

in Figure 6(d), where it should be noted that the under arms of the object are unaffected by the motion of the head vertices of the cage.

Although the examples of interior control given here are somewhat subtle, the control of such subtlety is an absolute requirement in high-end articulation. It is one of the most important factors that separates high-end articulation from less exacting applications.

Formally speaking, harmonic functions are C^∞ away from boundaries, but they are only guaranteed to be C^0 across boundary elements. Except for numerical issues, the blue object shown in Figure 6(d) would therefore be as smooth as the undeformed object. However, if a smooth object crosses an interior facet of the cage in the bind pose, its image under the deformation will only be C^0. Sometimes these discontinuities are desired. In situations where discontinuities are not acceptable, the articulator has a number of options. For instance, cluster weighting together with the smoothing introduced when the deformed control points are averaged to produce a B-spline or subdivision surface is often sufficient, as in Figure 5. Another alternative is to use a more dense distribution of interior facets that come close to but don't intersect the object in the bind pose, much like the situation shown in Figure 6.

4 Dynamic binding

Our discussion thus far, as well Ju et al. [2005], has implicitly assumed *static binding*; that is, object point locations are assumed to be known at the time the coordinate values are computed.

However, consider a situation such as the one shown in Figure 7 where an arm is first twisted within the cage, then is bound to the cage and deformed. Both the arm twist and the cage deformation must occur at real-time rates for use by animators. Since the arm twist changes the position of object points within the cage, the object point locations are not known in advance. To address this issue, we store the entire solution grid for each coordinate at precompu-

tation time. At pose time, we then use multi-linear interpolation to compute the coordinates at each of the object points prior to applying Equation 1. We refer to this process as *dynamic binding*. Dynamic binding was also used in Figures 1(b) and 1(c) where the eye lids were opened prior to binding and deformation.

Naively storing the solution grids to support dynamic binding requires prohibitive amounts of memory. For instance, the solution grids for the character shown in Figure 2 took approximately 100MB to store naively. However, since each harmonic coordinate decays relatively quickly away from its corresponding cage vertex, significant savings are possible by sparsely storing the coordinate grids. The sparse grid data structure we use consists of an array, with one entry per grid cell. The contents of entry (i, j) consists of a list of index-value pairs, where the index names a cage vertex, and the value is the harmonic coordinate for that cage vertex at cell (i, j). The structure is sparse because only the index-value pairs whose value is above a threshold are stored. Using this scheme, the sparse solution grid for the character shown in Figure 2 required only about 3MB. We should also note that dynamic binding is not exclusive to harmonic coordinates — a similar strategy would work for free-form deformations or mean value coordinates.

5 Implementation

We currently use a straightforward multi-grid finite difference solver to compute harmonic coordinates. It is identical to the solver described in DeRose and Meyer [2006], except for the relaxation stencils used at the boundaries. Whereas simple averaging was used in DeRose and Meyer, we currently obtain higher accuracy at the boundaries by computing intersections of grid edges with the cage boundaries; we then use the reciprocal of the distance from the grid center to the intersection as the stencil weight.

The finest level of the grid has size 2^s on a side, where s is currently a user selected parameter. Running times, memory usage, and values of s for various examples used in the paper are presented in Section 6.

Our 3D implementation is currently limited to cages with triangular facets, though it would be easy to extend it to handle planar quad faces. Non-planar quad meshes are more problematic and are a topic of future research.

6 Statistics

Memory and time requirements for examples given in the paper are listed in Table 6. In each of the examples we selected the lowest grid resolution s that produced an acceptable discretization error. The error was measured as the ratio of the maximum residual at the
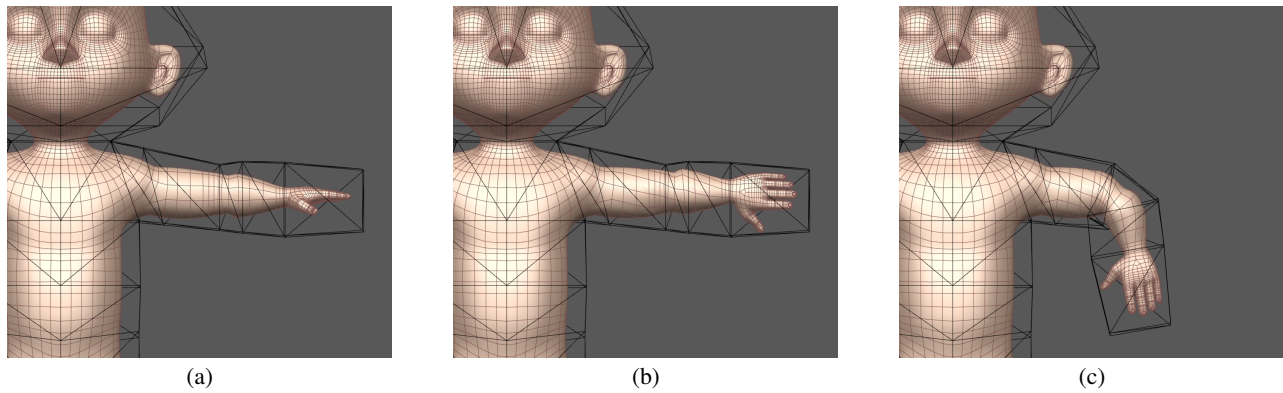
Figure 7: *Dynamic binding. (a) An object and cage in their rest poses. (b) the object after applying the arm twist. This is the pose in which the object is bound to the cage. (c) The deformed object after applying the harmonic deformation. Dynamic binding allows the arm twist as well as the harmonic deformation to be performed in real-time.*

object points to the diagonal of the bounding box of the object. In all cases the error was less than 0.005.

The total memory utilized by the harmonic solver during the pre-processing mentioned in Section 5 never exceeded 90MB for any of our examples. For each level of the multi-grid, our solver replaces the value at each cell with the average of its neighbors until the average change in value between successive iterations is less than 10^{-6}. This same number (10^{-6}) was used as a sparsity threshold when storing the solution grids for dynamic binding.

7 Summary and Future Work

We have shown that harmonic coordinates offer an effective character articulation method that improves on previous methods in several ways. First, harmonic coordinates are non-negative, and their influence falls off with distance as measured within the cage, leading to intuitive behavior even in strongly concave situations. Second, harmonic deformations offer greater topological freedom in crafting control cages. Specifically, arbitrary linear cell complexes can be used where necessary to more precisely control the interior nature of the deformation.

The main limitation of our method for use in character articulation is the lack of a closed formed expression. However, that limitation is far out weighed by the increased quality and controllability. Another limitation, at least for some applications, is that mean value coordinates are defined everywhere, whereas harmonic coordinates are defined only on the interior of the cage. For this reason, mean value coordinates can be used for extrapolation as well as interpolation. For some applications reliance on a closed cage might also be a limitation, but for the reasons cited earlier, cages are an advantage for character articulation. Finally, like other volume deformation methods such as subdivision and mean value coordinates, our method when applied to extreme deformations will not preserve fine mesh detail as well as methods based on differential coordinates, such as those described in Sorkine [2006] or Shi et al. [2006].

Our emphasis so far in the research has been on assessing quality, generality, and ease of authoring of the method. Our main area of future work is to improve the speed and memory requirements of the solver. The improvements we intend to explore include the following:

- Exploiting parallelism. Each harmonic coordinate function can be computed independently and in parallel on separate machines in our server farm. Parallelism alone should reduce the solve time for Figure 1 to the 15-20 second range.

- Use of factorization solvers. We currently use a simple "home grown" solver. The use of factorization solvers such as the Super LU method used by Zayer et al. [2005] should be faster for cages with a large number of vertices since factorization can be done once, requiring only the back substitution step to be done for each harmonic coordinate. Initial results for the cage of Figure 1 indicates a speed up of nearly a factor of three.

- Use of adaptive grids. Since the coordinate fields vary slowly on the interior of the cage, it should be possible to use relatively large grid cells over most of the volume of the cage, transitioning to smaller cells only near the boundaries. This should allow us to more accurately adapt to detail at the boundary at reasonable cost.

- Localizing re-solves. As the cage is being designed, most edits are confined to a local region of the cage. Rather than completely re-solving for all coordinates as we do now, we should instead solve for only those coordinates that are influenced by the modified region. That is, we should not re-solve for coordinates whose (sparsified) value is zero in the region of change. Happily, the benefits of this optimization increase as the cage becomes more complicated.

Since cages are specifically designed to be fairly coarse, we expect cage complexities to range from tens of vertices (especially for local use cases such as the one shown in Figure 5), to no more than a thousand vertices for a full body cage of a complicated character. Using the improvements listed above, we expect to reduce full solve times into the tens of seconds range even for the most complicated cages we are likely to encounter.

Harmonic coordinates are a form of generalized barycentric coordinates and can be defined for any dimension. Generalized barycentric coordinates are fundamental building blocks in a number of other areas such as the construction of N-sided surface patches [Loop and DeRose 1989] and finite element analysis [Wachpress 1975]. As a second area of future research it would be interesting to investigate the use of d dimensional harmonic coordinates in application areas other than character articulation.

References

BASS, R. 1995. *Probabilistic Techniques in Analysis*. Springer-Verlag.

CAPELL, S., GREEN, S., CURLESS, B., DUCHAMP, T., AND POPOVIC, Z. 2002. A multiresolution framework for dynamic

deformations. In *ACM SIGGRAPH Symposium on Computer Animation*, ACM SIGGRAPH, 41–48.

CARR, J. C., BEATSON, R. K., CHERRIE, J. B., MITCHELL, T. J., FRIGHT, W. R., MCCALLUM, B. C., AND EVANS, T. R. 2001. Reconstruction and representation of 3D objects with radial basis functions. In *SIGGRAPH 2001, Computer Graphics Proceedings*, ACM Press / ACM SIGGRAPH, E. Fiume, Ed., 67–76.

CHOE, B., LEE, H., AND KO, H.-S. 2001. Performance-driven muscle-based facial animation. *The Journal of Visualization and Computer Animation 12*, 2, 67–79.

DEROSE, T., AND MEYER, M. 2006. Harmonic coordinates. Pixar Technical Memo 06-02, Pixar Animation Studios, January. http://graphics.pixar.com/HarmonicCoordinates/.

DUCHON, J. 1977. Splines minimizing rotation invariant seminorms in sobolev spaces. In *Lecture Notes in Mathematics*, Springer-Verlag, vol. 571.

FLOATER, M. S., KOS, G., AND REIMERS, M. 2005. Mean value coordinates in 3d. *Computer Aided Geometric Design 22*, 623–631.

FLOATER, M. S., HORMANN, K., AND KOS, G. 2006. A general construction of barycentric coordinates over convex polygons. *Advances in Comp. Math. 24*, 311–331.

FLOATER, M. 2003. Mean value coordinates. *Computer Aided Geometric Design 20*, 1, 19–27.

IGARASHI, T., MOSCOVICH, T., AND HUGHES, J. F. 2005. As-rigid-as-possible shape manipulation. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, ACM Press, New York, NY, USA, 1134–1141.

JOSHI, P., TIEN, W. C., DESBRUN, M., AND PIGHIN, F. 2006. Learning controls for blend shape based realistic facial animation. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses*, ACM Press, New York, NY, USA, 17.

JOSHI, P., MEYER, M., DEROSE, T., GREEN, B., AND SANOCKI, T. 2007. Harmonic coordinates for character articulation. Pixar Technical Memo 06-02b, Pixar Animation Studios. http://graphics.pixar.com/HarmonicCoordinatesB/.

JU, T., SCHAEFER, S., AND WARREN, J. 2005. Mean value coordinates for closed triangular meshes. *ACM Trans. Graph. 24*, 3, 561–566.

LEWIS, J. P., CORDNER, M., AND FONG, N. 2000. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 165–172.

LOOP, C. T., AND DEROSE, T. D. 1989. A multisided generalization of bézier surfaces. *ACM Trans. Graph. 8*, 3, 204–234.

MACCRACKEN, R., AND JOY, K. I. 1996. Free-form deformations with lattices of arbitrary topology. In *Proceedings of SIGGRAPH '96*, Annual Conference Series, 181–199.

MEYER, M., LEE, H., BARR, A., AND DESBRUN, M. 2002. Generalized barycentric coordinates for irregular polygons. *Journal of Graphics Tools 7*, 1, 13–22.

PINKHALL, U., AND POLTHIER, K. 1993. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2*, 15–36.

PORT, S. C., AND STONE, C. J. 1978. *Brownian Motion and Classical Potential Theory*. Academic Press.

R.SIBSON. 1981. A brief description of natural neighbor interpolation. In *Interpreting Multivariate Data*, V. Barnett, Ed. John Wiley, 21–36.

SEDERBERG, T. W., AND PARRY, S. R. 1986. Free-form deformation of solid geometric models. In *SIGGRAPH '86: Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 151–160.

SHI, L., YU, Y., BELL, N., AND FENG, W.-W. 2006. A fast multigrid algorithm for mesh deformation. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, ACM Press, New York, NY, USA, 1108–1117.

SORKINE, O. 2006. State of the art report: Differential representations for mesh processing. *Computer Graphics Forum 25*, 4.

SUMNER, R. W., ZWICKER, M., GOTSMAN, C., AND POPOVIC, J. 2005. Mesh-based inverse kinematics. *ACM Trans. Graph. 24*, 3, 488–495.

WACHPRESS, E. 1975. *A Rational Finite Element Basis*. Academic Press.

WAHBA, G. 1990. Spline models for observational data. In *CBMS-NSF Regional Conference Series in Applied Mathematics*, SIAM, Philadelphia, PA, USA, vol. 59.

WARREN, J. 1996. Barycentric coordinates for convex polytopes. *Advances in Computational Mathematics 6*, 97–108.

ZAYER, R., RÖSSL, C., KARNI, Z., AND SEIDEL, H.-P. 2005. Harmonic guidance for surface deformation. In *The European Association for Computer Graphics 26th Annual Conference : EUROGRAPHICS 2005*, Blackwell, Dublin, Ireland, M. Alexa and J. Marks, Eds., vol. 24 of *Computer Graphics Forum*, Eurographics, 601–609.