Computers & Graphics 57 (2016) 92–101

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Technical Section

3D Model deformations with arbitrary control points ☆

M. Àngels Cerveró *, Àlvar Vinacua, Pere Brunet

Departament de Ciències de la Computació, Universitat Politècnica de Catalunya, Campus Nord, 08034 Barcelona, Spain

ARTICLE INFO

Article history:
Received 16 January 2016
Received in revised form
17 March 2016
Accepted 25 March 2016
Available online 7 April 2016

Keywords:
Deformations
Cage-based
Interactive mesh deformation

ABSTRACT

Cage-based space deformations are often used to edit and animate images and geometric models. The deformations of the cage are easily transferred to the model by recomputing fixed convex combinations of the vertices of the cage, the control points. In current cage-based schemes the configuration of edges and facets between these control points affects the resulting deformations. In this paper we present a family of similar schemes that includes some of the current techniques, but also new schemes that depend only on the positions of the control points. We prove that these methods afford a solution under fairly general conditions and result in an easy and flexible way to deform objects using freely placed control points, with the necessary conditions of positivity and continuity.

1. Introduction

Techniques to deform three dimensional models are important in computer graphics. They can be used as modelling tools, to animate models, or within simulations. Additionally, some applications may require the deformation to satisfy other restrictions, like clamped portions of the model, or volume preservation.

A large number of methods currently in use and in the literature follow the Cage paradigm, whereby the model is surrounded by a coarse polyhedral cage, and the vertices of that cage—the control vertices—are used as handles to control the deformations. To describe how the space inside (and around) the cage deforms as the vertices of the cage move, some form of generalised barycentric coordinates with respect to the control vertices is used. These schemes give each point a set of coordinates that depend on the relative position of the point itself and the control points. Given such a coordinate system, when the cage is deformed, it is just a matter of computing the new positions of the points with the given coordinates to retrieve the deformed model. If these coordinates are smooth, the induced deformations will also be smooth. The advantage, of course, is that the user (or the simulation or optimisation code) must only concern with a small number of handles (the control points) as opposed to a very large number of points (the vertices of the model). This paradigm is simple, elegant and efficiently deforms the models. However, some cage-free deformation techniques have been introduced recently. They provide more flexibility in the choice of deformation handles—which may not be connected—and provide powerful tools to make the deformation process more versatile and intuitive.

In our research, we are especially interested in the deformation of soft tissues in medical or biological models. These models represent organs and tissues which are soft and lack an internal rigid structure. They are elastic but incompressible. In these cases an obvious guiding structure to help in devising a cage seldom exists, making schemes that do not rely on connectivity more natural to use.

In this paper we propose some new methods to compute a set of generalised barycentric coordinates which are *cage-free* and depend only on the positions of the deformation handles. The main contributions we present here are:

- The definition of a formal framework, the Celestial Coordinates, in which many of the existing schemes can be described.
- Two new Celestial Coordinate schemes that depend only on the positions of the control points, and not on their connectivity.

Section 2 discusses the previous work in this area. Then, Section 3 defines the Celestial Coordinates family and Sections 4 and 5 derive two new systems that belong to this family. Finally, Sections 6–8 present results to evaluate these new schemes and our plans for future work along these lines.

2. Previous work

There is a lot of bibliography proposing different types of Generalised Barycentric Coordinate (GBC) systems so the deformations of the control points have the desired properties of smoothness, locality and real-time responsiveness.

☆This article was recommended for publication by S. Hahmann.

* Corresponding author at: Departament de Ciències de la Computació, Universitat Politècnica de Catalunya, Campus Nord, 08034, Barcelona, Spain.

E-mail addresses: macervero@lsi.upc.edu (M.À. Cerveró), alvar@cs.upc.edu (À. Vinacua), pere@lsi.upc.edu (P. Brunet).

http://dx.doi.org/10.1016/j.cag.2016.03.010
0097-8493/

The best known examples are the classical *Barycentric Coordinates*, defined by Möbius in 1827, which restrict the cage to be a simplex. More recently, *Mean-Value Coordinates* (MVC) [1] have been extended to 3D [2,3]. They generalise barycentric coordinates to the kernel (the set of points that see all vertices) of star-shaped polyhedral cages. These coordinates are guaranteed to be positive in this kernel and C^∞ inside and outside of it. However, they are only C^0 on its boundary.

Harmonic Coordinates (HC) [4] do not have a closed formulation because they depend on the specific problem to solve. HC are C^∞ inside the cage and C^0 on its boundary. They have a more local effect than the MVC.

Positive Mean Value Coordinates (PMVC) [5] were introduced to ensure positive coordinates all over the cage, not only in its kernel. To fulfil this requirement they must relax the constrains of smoothness and continuity through the supporting planes of the boundary of this cage.

The *Green Coordinates* (GC) [6] are a new approach to perform shape-preserving deformations that require the normals of the faces of the cage to compute the coordinates. They are C^∞ inside and outside the cage but discontinuous at its boundary.

Although all these schemes provide efficient deformations, MVC may distort local details and PMVC and GC have discontinuities at the boundaries.

Li et al. [7] present a deformation technique using GC. The supporting cage is replaced by an umbrella shaped cell. This umbrella is automatically constructed over a point of the model specified by the user and updated during the deformation step. This method simulates shape deformation schemes in terms of the flexibility of the control handles. It also performs local shape-preserving deformations in real-time. Although the construction of the umbrella is completely transparent to the user, this technique is still completely dependent on the topology of the pseudo-cage.

Garcia et al. [8] present a multi-cage system to restrict the deformations to local region. Furthermore, their technique also increases the continuity of the coordinates across the boundaries between cages by computing a blending function applied in a parametrised neighbourhood of these faces.

Finally, Jacobson et al. [9] propose the *Bounded Biharmonic Weights* (BBW) that allow multiple deformation controls. The user can operate with cages, skeletons and isolated control points to accomplish the desired deformation. Their method reaches its goals through a space discretisation and the minimisation of a Laplacian energy.

3. Overview of Celestial Coordinates

Let us consider a set of $\mathbb{R}^3$ vertices $\mathcal{V}=\{\mathbf{v}_1,\dots,\mathbf{v}_n\}$. Unless explicitly defined otherwise, in what follows we will consider that the Deformation Domain $\mathcal{D}$ of the set $\mathcal{V}$ is its Convex Hull. Generalised Barycentric Coordinates (GBC) assign n coordinates to any 3D point $\mathbf{p}\in\mathcal{D}$,

$$\mathbf{p}=\sum_{i=1}^{n}\mathbf{v}_i\alpha_i(\mathbf{p}), \tag{1}$$

where $\alpha_i(\mathbf{p})$ is the Generalised Barycentric Coordinate of $\mathbf{p}$ with respect to the ith control vertex $\mathbf{v}_i$. These coordinates are only computed once for any relevant point of the deformable geometric model. Deformations are then driven by $\mathcal{V}$ in a very simple way through Eq. (1), using the initial coordinates $\alpha_i(\mathbf{p})$ and the new positions of the control vertices $\mathbf{v}'_i$.

By converting Eq. (1) into $\sum_{i=1}^{n}(\mathbf{v}_i\alpha_i(\mathbf{p})-\mathbf{p})=0$, we can write

$$\sum_{i=1}^{n}\beta_i\vec{\mathbf{q}}_i=\vec{\mathbf{0}} \tag{2}$$

where $\vec{\mathbf{q}}_i=\frac{(\mathbf{v}_i-\mathbf{p})}{d_i}$ and $d_i=\|(\mathbf{v}_i-\mathbf{p})\|$. With this expression, the final coordinates α_i are obtained as

$$\alpha_i(\mathbf{p})=\frac{\omega_i(\mathbf{p})}{\sum_{j=1}^{n}\omega_j(\mathbf{p})}, \tag{3}$$

where $\omega_i(\mathbf{p})=\frac{\beta_i}{d_i}$, a well-known alternative formulation of GBC.

In what follows, we will use the term *Celestial Coordinates* to identify the family of positive Generalised Barycentric Coordinate (GBC) schemes. They are defined by Eq. (2) by imposing that $\forall i\ \beta_i\geq 0$

$$\sum_{i=1}^{n}\beta_i\vec{\mathbf{q}}_i=\vec{\mathbf{0}}\quad \forall i\ \beta_i\geq 0. \tag{4}$$

The name of *Celestial Coordinates* (CC) comes from the use of unit vectors $\vec{\mathbf{q}}_i$, which are the projection of the control vertices $\mathbf{v}_i$ over a unit Sphere during their computation process in Eq. (4). We call this Sphere the *Celestial Sphere*.

Positive Mean-Value Coordinates [5] are a good example of a member of the Celestial Coordinates family since they are always positive in their domain. However, most of the schemes discussed in the previous Section are not CC. For example, Mean-Value Coordinates [2] are only positive if $\mathbf{p}$ is located inside the Kernel of the user-defined cage. Thus, they behave as Celestial Coordinates if, and only if, the Cage is convex. A similar situation takes place with the Spherical Barycentric Coordinates [10]. Unlike these schemes, the goal of the next two sections is to propose CC schemes which are only based on the position of the control vertices $\mathbf{v}_i$ and which do not depend on user-defined Cages nor on automatically computed connectivity among these control vertices. We therefore define a CC-subfamily called *Point-Based Celestial Coordinates*. They include all schemes in the CC family that do not need any kind of connectivity between the control vertices to compute the set of coordinates.

Any algorithm providing a set of positive β_i values fulfilling Eq. (4) for every 3D point $\mathbf{p}\in\mathcal{D}$ is a CC-scheme candidate (it should also fulfill standard GBC properties as reproduction of the identity, reproduction of the unity and smoothness, as discussed in Section 6). Anyway, Eq. (4) has two possible interpretations:

- First, it can be seen as a set of three scalar products between $\mathbb{R}^n$ vectors. Let us define the vectors $\chi,y,z\in\mathbb{R}^n$ as the x-coordinates, y-coordinates and z-coordinates of the projected vectors $\vec{\mathbf{q}}_i$, respectively, and the vector $\boldsymbol{\beta}=\{\beta_1,\dots\beta_n\}$. Then, Eq. (4) requires that $\boldsymbol{\beta}\in\mathbb{R}^n_+$ and also that $\boldsymbol{\beta}\in V^\perp$ where V is the linear space spanned by χ,y,z. In other words, $\boldsymbol{\beta}$ must belong to the region $V^\perp\cap\mathbb{R}^n_+$ in $\mathbb{R}^n$.
- Eq. (4) can also be interpreted in $\mathbb{R}^3$, by defining a convex linear combination of the projected vectors $\vec{\mathbf{q}}_i$ which must result in the null vector.

Observe that $V^\perp\cap\mathbb{R}^n_+$ is always non-empty for points $\mathbf{p}\in\mathcal{D}$, as any point inside a Convex Hull can be expressed as a convex combination of the vertices that define this Convex Hull, and $\mathbf{p}$ is always in the Convex Hull of the projections of the control vertices on the unit Sphere.

The following sections present two new schemes that belong to the *Point-Based Celestial Coordinates* family. They are the *T-Celestial Coordinates* and the *S-Celestial Coordinates. T-Celestial Coordinates* derive from the first $\mathbb{R}^n$ interpretation, whereas *S-Celestial Coordinates* come from the $\mathbb{R}^3$ one.

4. T-Celestial Coordinates

T-Celestial Coordinates derive from the $\mathbb{R}^n$ interpretation in Section 3. They are based on a transformation function T which maps any vector in the span of the vectors $\chi,y,z\in\mathbb{R}^n$ to the positive region $\mathbb{R}^n_+$. The computation of the vector $\boldsymbol{\beta}$ for any point