Computer-Aided Design 45 (2013) 463–472

Contents lists available at SciVerse ScienceDirect

Computer-Aided Design

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

GPU-based computation of discrete periodic centroidal Voronoi tessellation in hyperbolic space

Liang Shuai [a], Xiaohu Guo [a,*], Miao Jin [b]

[a] University of Texas at Dallas, United States
[b] University of Louisiana at Lafayette, United States

ARTICLE INFO

Keywords:
Centroidal Voronoi tessellation
Universal covering space
Hyperbolic space
GPU algorithm

ABSTRACT

Periodic centroidal Voronoi tessellation (CVT) in hyperbolic space provides a nice theoretical framework for computing the constrained CVT on high-genus (genus $>$ 1) surfaces. This paper addresses two computational issues related to such a hyperbolic CVT framework: (1) efficient reduction of unnecessary site copies in neighbor domains on the universal covering space, based on two special rules; (2) GPU-based parallel algorithms to compute a discrete version of the hyperbolic CVT. Our experiments show that with the dramatically reduced number of unnecessary site copies in neighbor domains and the GPU-based parallel algorithms, we significantly speed up the computation of CVT for high-genus surfaces. The proposed discrete hyperbolic CVT guarantees to converge and produces high-quality results.

1. Introduction

The centroidal Voronoi tessellation (CVT) [1] is a special type of Voronoi diagram, where every site coincides exactly with the centroid of its Voronoi cell. The celebrating Gersho's conjecture [2] in 2D, proved by Tóth [3], states that the shape of the Voronoi cells converges to uniform regular hexagons when the CVT is optimized globally. This property inspired many researchers to compute the constrained CVT [4] on surfaces, for applications where a uniform sampling or remeshing of the surface is desired.

Different methods of computing constrained CVT on surfaces can be roughly categorized into two classes: "extrinsic" and "intrinsic" approaches. Extrinsic approaches [4–6] compute an Euclidean Voronoi diagram in the ambient 3D space and its intersection of a surface, with sites constrained on the surface. If two regions of the surface are close in the 3D space but far away along the surface, the computed constrained CVT on surface tends to be incorrect [7,8].

Intrinsic approaches [9,7,8], which overcome the above limitations of their extrinsic counterparts, compute the CVT in a 2D parameter domain of a surface, with a density function applied to compensate the introduced area distortion of surface parametrization. To allow sites move freely across artificially cut open surface boundaries on its parameter domain for non-topological disk surfaces, Rong et al. [8] proposed to compute the CVT in a 2D periodic parameter domain, called the *Universal Covering Spaces* of surfaces, which are 2D spaces with constant curvatures — spherical, Euclidean, and hyperbolic spaces.

Computing a CVT in a 2D periodic parameter domain is equivalent to computing a periodic CVT in its corresponding space. Computing a periodic CVT in spherical and Euclidean spaces has been well studied in previous literatures [4,10,11]. Thus the main challenge resides in the efficient computation of the periodic CVT in hyperbolic space. In this paper we propose several strategies to speed up the computation of periodic CVT in hyperbolic space, including two special rules to efficiently reduce the number of site copies, and a GPU-based parallel computation framework of the discrete hyperbolic CVT.

1.1. Preliminaries

We first introduce briefly the concept of universal covering space, and then present two definitions related to periodic CVT in hyperbolic space.

A *covering map* is a surjective continuous map π from a topological space $\bar{U}$ to another topological space U such that any point $\mathbf{p}$ in U has a neighborhood $N(\mathbf{p})$ satisfying $\pi^{-1}(N(\mathbf{p}))$ is a collection of disjoint sets, and each set can be homeomorphically mapped onto U by π. $(\bar{U}, \pi)$ is called the *covering space* of U, and sometimes people simply denote it as $\bar{U}$. $(\bar{U}, \pi)$ is the *universal covering space (UCS)* of U if $\bar{U}$ is simply connected. A *deck transformation* $\phi : \bar{U} \to \bar{U}$ keeps the covering map π unchanged: $\pi = \pi \circ \phi$. All deck transformations form a so called *Fuchsian group* G. A *fundamental domain* F of the UCS is a subset of $\bar{U}$ such that $\bar{U} = \cup_{\phi \in G} \phi(F)$. The UCS of a high-genus (genus $>$ 1) surface can be conformally embedded into a 2D hyperbolic space [12]. Fig. 1

* Corresponding author.
E-mail address: xguo@utdallas.edu (X. Guo).

0010-4485/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2012.10.029

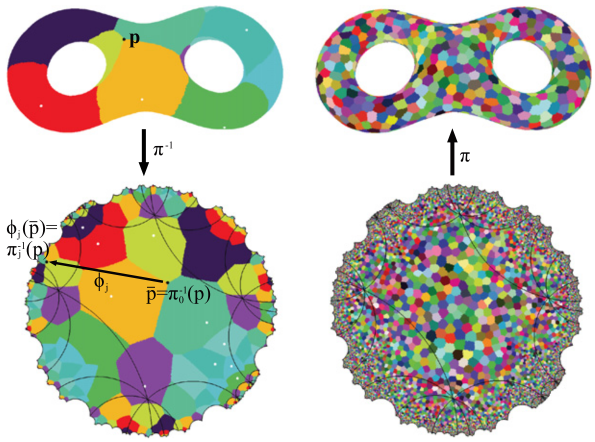

Fig. 1. Left: a point $\mathbf{p}$ on the surface is mapped to $\bar{\mathbf{p}}$ in the center domain by the inverse covering map π_0^{-1}, and $\bar{\mathbf{p}}$ can be mapped to $\pi_j^{-1}(\mathbf{p})$ in the neighbor domain j by the deck transformation ϕ_j; Right: we can get the CVT result on a genus-2 surface by computing a periodic CVT in the hyperbolic UCS.

shows a double-torus surface and its UCS embedded on hyperbolic plane. We refer readers to Munkres's book [13] for more details.

Definition 1. Let U be a surface and $(\bar{U}, \pi)$ be its UCS. Given a point set $S = \{\mathbf{s}_i \in U \mid i = 1, \ldots, n\}$ on U, the *Voronoi diagram in universal covering space* of U induced by S can be defined as the subdivision of $\bar{U}$ into Voronoi cells Ω_i^{ucs}:

$$\Omega_i^{ucs} = \{\mathbf{p} \in \bar{U} \mid d_{\bar{U}}(\mathbf{p}, \bar{\mathbf{s}}_i) < d_{\bar{U}}(\mathbf{p}, \bar{\mathbf{s}}_j), \exists\, \bar{\mathbf{s}}_i \in \pi^{-1}(\mathbf{s}_i), \forall\, \bar{\mathbf{s}}_j \in \pi^{-1}(\mathbf{s}_j), \forall j \neq i\}, \tag{1}$$

where $d_{\bar{U}}(\cdot, \cdot)$ is the distance in space $\bar{U}$.

Note that in the definition each point (or *site*) $\mathbf{s}_i$ is duplicated into infinite number of copies via π^{-1}. The fundamental domain is periodically repeated in UCS, so a Voronoi diagram in UCS is also a *periodic Voronoi diagram (PVD)*.

Due to its periodicity, the PVD defined in (1) only needs to be computed in one fundamental domain (referred as *center domain* $\bar{U}_0$ later). Define the *neighbor domains* of $\bar{U}_0$ be the fundamental domains which share at least one vertex with $\bar{U}_0$, denoted by $\bar{U}_j$, $j = 1, \ldots, h$, where $h = 16g^2 - 8g$ is the number of neighbor domains for a genus-g surface. Use $\pi_j^{-1}(\mathbf{s})$ to denote the preimages of site $\mathbf{s}$ in different fundamental domains ($j = 0$ for the center domain and $j = 1, \ldots, h$ for neighbor domains). Let ϕ_j be such a deck transformation that $\phi_j(\bar{\mathbf{s}}) = \pi_j^{-1}(\mathbf{s})$, where $\bar{\mathbf{s}} = \pi_0^{-1}(\mathbf{s})$ (Fig. 1). Then we can refer to $\phi_j(\bar{\mathbf{s}}_i), j = 1, \ldots, h$ as the *site copies* of site $\bar{\mathbf{s}}_i = \pi_0^{-1}(\mathbf{s}_i)$ later for convenience.

This paper focuses on PVD in 2D hyperbolic space. There are different models of 2D hyperbolic space, such as Poincaré disk, Klein disk, Poincaré half-plane, and Minkowski model. All these models are equivalent. In this paper we use the Poincaré disk model to visualize the hyperbolic Voronoi diagram (see Fig. 1) and the Minkowski model to define the centroid of a hyperbolic region [8].

Definition 2. Given a region Ω on the Minkowski model, and density $\rho(\mathbf{p})$ for any point $\mathbf{p} \in \Omega$, the *centroid* of Ω is defined as:

$$\mathbf{c} = \frac{1}{\eta} \int_{\Omega} \rho(\mathbf{p})\mathbf{p}\, d\mathbf{p}, \tag{2}$$

where

$$\eta = \left\| \int_{\Omega} \rho(\mathbf{p})\mathbf{p} d\mathbf{p} \right\|_M . \tag{3}$$

$\| \cdot \|_M$ is the Minkowski norm which can be defined through its inner product: $\| \cdot \|_M = \sqrt{\langle \cdot, \cdot \rangle_M}$.

Here the Minkowski inner product is defined as $\langle \mathbf{p}, \mathbf{q} \rangle_M = z_\mathbf{p} z_\mathbf{q} - x_\mathbf{p} x_\mathbf{q} - y_\mathbf{p} y_\mathbf{q}$ for two points $\mathbf{p} = (x_\mathbf{p}, y_\mathbf{p}, z_\mathbf{p})$ and $\mathbf{q} = (x_\mathbf{q}, y_\mathbf{q}, z_\mathbf{q})$ on the Minkowski model. Note that their hyperbolic distance can be computed by $d_M(\mathbf{p}, \mathbf{q}) = \cosh^{-1}(\langle \mathbf{p}, \mathbf{q} \rangle_M)$.

Given a set of sites $S = \{\mathbf{s}_i \mid i = 1, \ldots, n\}$ on the Minkowski model, and its Voronoi diagram as $\Omega = \cup \Omega_i$, where Ω_i is the Voronoi cell associated with site $\mathbf{s}_i$, the *hyperbolic CVT energy* is defined as:

$$E(S, \Omega) = \sum_i \int_{\Omega_i} \rho(\mathbf{p}) \cosh(d_M(\mathbf{p}, \mathbf{s}_i))\, d\mathbf{p}. \tag{4}$$

With the above defined Voronoi diagram and centroid in hyperbolic space, the hyperbolic CVT energy is proved to converge with Lloyd's algorithm [8].

1.2. Motivation and contribution

Rong et al. [8] proposed a nice periodic CVT framework in hyperbolic space. However, two computational issues hinder their algorithms from being practical for general high-genus surfaces.

The first issue is that they compute the part of PVD within a center domain from a full site copies of the center domain and its neighbor domains: $\{\pi_j^{-1}(\mathbf{s}_i) \mid \forall \mathbf{s}_i \in S, j = 0, \ldots, h\}$, and then the intersection of the resulting Voronoi diagram with the center domain. Although the shape of periodic Voronoi cells inside the center domain can be affected by the site copies located in its neighbor domains, there are $16g^2 - 8g$ neighbor domains for a genus-g surface and only a small portion of the site copies in them will affect the Voronoi cells near the boundary of the center domain. In Section 3, We propose two simple rules, which can be computed efficiently, to significantly reduce the number of unnecessary site copies in neighbor domains.

The second issue is that computing a hyperbolic CVT is extremely time-consuming. For example, computing 1000 sites on a genus-3 Sculpture surface takes around 55 s for each Lloyd's iteration on a desktop computer with a Core 2 Duo 2.93 GHz CPU. We can utilize the parallel computability of the programmable GPU to accelerate this process. We first define a discrete version of the hyperbolic CVT with each triangle represented by its centroid and the constrained CVT approximated with clusters of triangles. Our discrete CVT is similar to Valette et al. [14]. However, it is formulated in a 2D periodic hyperbolic domain, with its discrete CVT energy proved to converge. We then introduce a parallel mesh flooding algorithm to efficiently compute the defined discrete hyperbolic PVD in Section 4.2. We further show that the energy of the discrete hyperbolic CVT is guaranteed to converge under our GPU-based computational framework in Section 5.

The contribution of this paper can be summarized as:

1. Two computing efficient rules are introduced to reduce the unnecessary site copies in neighbor domains for computing hyperbolic PVD;
2. A GPU-based parallel mesh flooding algorithm is proposed to compute the discrete hyperbolic Voronoi diagram.

Both of the two aspects serve for speeding up the computation of hyperbolic CVT.

2. Related work

In this section, we give a brief review of existing research related to this work.