Computers & Graphics 53 (2015) 28–36

Contents lists available at ScienceDirect

Computers & Graphics

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

40 years of Computer Graphics in Darmstadt

Beam meshes

Ronald Richter, Marc Alexa *

Marchstr. 23, 10587 Berlin, Germany

ARTICLE INFO

Article history:
Received 18 August 2015
Accepted 20 August 2015
Available online 9 October 2015

Keywords:
Geometry processing
Centroidal Voronoi diagram
Digital manufacturing

ABSTRACT

We present an approach for representing free-form geometry with a set of beams with rectangular cross-section. This requires the edges of the mesh to be free of torsion. We generate such meshes in a two step procedure: first we generate a coarse, low valence mesh approximation using a new variant of anisotropic centroidal Voronoi tessellation. Then we modify the mesh and create beams by incorporating constraints using iterative optimization. For fabrication we provide solutions for designing the joints, generating a cutting place for CNC machines, and suggesting a building sequence. The approach is demonstrated at several virtual and real results.

1. Introduction

The recent trend in what is commonly referred to as 3d printing has also reignited the interest in other forms of automatic or semi-automatic manufacturing of arbitrary free-form shapes. A particularly interesting approach is based on cutting out planar pieces from a sheet of material and assembling the pieces to represent the volume or boundary of the shape [1–5].

In this work, we aim at physically realizing an arbitrary polygon mesh in timber, plastic, or steel by one planar beam for each edge in the mesh (see Fig. 1). Compared to an arbitrary polygon mesh, the main geometric limitation for this to be possible is that each edge needs to be torsion-free: edges lie in planes that intersect in a common line in the vertices (Fig. 1, bottom). Moreover, we consider additional constraints that make manufacturing simpler, for example extruding the beam from a constant cross section, i. e. not only its thickness is constant but also its height. We call meshes with torsion-free edges *beam meshes*. The goal of this work is to approximate a given 3d geometry with a beam mesh and to manufacture it.

Requiring beams to be straight and have constant height means that the polygon mesh has an offset mesh with parallel edges. Meshes with *planar faces* that have parallel offsets have been analyzed in detail in the seminal work of Pottmann et al. [6]. However, such piecewise planar meshes, for which a parallel offset mesh exists, form a restricted (linear) space and it is notoriously difficult to approximate arbitrary free-form shapes with planar meshes.

In contrast to the work by Pottmann et al. [6], in our approach we consider polygon meshes with arbitrary, not necessarily planar faces. We observe that generically *dual triangle meshes* are torsion free (Section 2). Consequently, we first approximate the given input geometry with a coarse dual triangle mesh (Section 4). Then we optimize the edges of this mesh to satisfy several constraints derived from practical considerations (Section 5). This two-step approach creates the desired beam mesh.

We also explain how to physically realize the beam mesh (Section 6). The non-zero thickness of the beams requires consideration of the geometry at the joints. The beams are then cut out of planar sheets of material using CNC machining. We show how to lay out the beams on a planar surface and how to mark the beams such that the subsequent construction is easy.

We provide several examples of physically realizable beam meshes, some of which we have actually constructed out of wood.

2. Preliminaries

We represent the coarse polygon mesh, which will be the basis of the beam mesh, by its $\mathfrak{v}$ vertices

$$V = (\mathbf{v}_0, \mathbf{v}_1, \ldots), \quad \{\mathbf{v}_i \in \mathbb{R}^3\} \tag{1}$$

and denote by Δ the matrix that generates $\mathfrak{e}$ directed edge vectors

$$\mathbf{e}_{ij} = \mathbf{v}_j - \mathbf{v}_i \tag{2}$$

from the vertices. Faces are shortest cycles of edges. We assume that the faces form a closed manifold. Under this assumption Δ represents the combinatorics of the mesh. For convenience we denote by $\mathcal{L}_i$ the cyclically ordered set of vertices connected to vertex i by an edge. Vertices have an associated (non-vanishing) normal vector $\mathbf{n}_i \in \mathbb{R}^3$, $\|\mathbf{n}_i\| > 0$.

The object of interest in this work is a *beam quad*, describing a planar beam as an offset to the vertices $\mathbf{v}_i$ and $\mathbf{v}_j$ of an edge $\mathbf{e}_{ij}$

* Corresponding author. Tel.: +49 30 314 73100; fax: +49 30 314 23596
E-mail address: marc.alexa@tu-berlin.de (M. Alexa).

http://dx.doi.org/10.1016/j.cag.2015.08.007

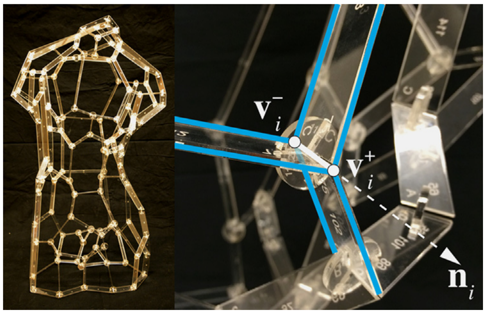

Fig. 1. A geometric shape represented by a mesh of beams, i.e. edges with a rectangular cross section. Beam quads are planar and intersect in one line segment $(\mathbf{v}_i^+, \mathbf{v}_i^-)$ point in the direction of the surface normal $\mathbf{n}_i$ in each vertex.

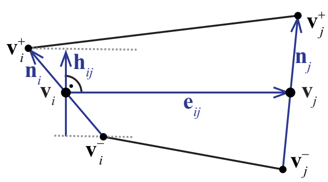

Fig. 2. Notation for a beam in the beam mesh representation.

along the positive and negative normal directions:

$$\begin{aligned} \mathbf{v}_i^- &= \mathbf{v}_i - \mathbf{n}_i \\ \mathbf{v}_i^+ &= \mathbf{v}_i + \mathbf{n}_i \\ \mathbf{v}_j^+ &= \mathbf{v}_j + \mathbf{n}_j \\ \mathbf{v}_j^- &= \mathbf{v}_j - \mathbf{n}_j \end{aligned} \tag{3}$$

For reference, we note that a vertex $\mathbf{v}_i$ is located in the centroid of the offset vertices, i. e.

$$\mathbf{v}_i = \frac{\mathbf{v}_i^- + \mathbf{v}_i^+}{2}. \tag{4}$$

We call the edge

$$\mathbf{v}_i^+ - \mathbf{v}_i^- = 2\mathbf{n}_i \tag{5}$$

the *normal edge* of a beam quad incident at vertex i; and we call the edges

$$\mathbf{v}_j^\pm - \mathbf{v}_i^\pm = \mathbf{e}_{ij} \pm (\mathbf{n}_j - \mathbf{n}_i) \tag{6}$$

offset edges of the beam quad for edge $\mathbf{e}_{ij}$.

We measure *height* of the beam quad orthogonal to the mid-edge $\mathbf{e}_{ij}$. The height vector in vertex i relative to the beam quad ij is

$$\mathbf{h}_{ij} = 2\left(\mathbf{n}_i - \frac{\mathbf{n}_i^\top \mathbf{e}_{ij}}{\mathbf{e}_{ij}^\top \mathbf{e}_{ij}} \mathbf{e}_{ij}\right) \tag{7}$$

and yields the height of the beam ij in vertex i as $\|\mathbf{h}_{ij}\|$. The height of the quad is not necessarily constant and for the height vector in vertex j of the same quad we use the notation $\mathbf{h}_{ji}$. The notation is summarized in Fig. 2.

3. Overview

Using beam quads, the requirement that an edge is torsion-free is equivalent to the associated beam quad being planar. This is the case if the two normals $\mathbf{n}_i, \mathbf{n}_j$ and the edge vector $\mathbf{e}_{ij}$ are contained in the same plane, that is when

$$\det(\mathbf{n}_i, \mathbf{n}_j, \mathbf{e}_{ij}) = 0, \tag{8}$$

which includes the case in which the normals $\mathbf{n}_i, \mathbf{n}_j$ are parallel and the edge direction is arbitrary. However, we do ask that each of the normals $\mathbf{n}_i, \mathbf{n}_j$ and the edge vector $\mathbf{e}_{ij}$ are linearly independent, which means the height vectors $\mathbf{h}_{ij}, \mathbf{h}_{ji}$, and thus the heights, are non-zero. Under this assumption, Eq. (8) implies that all beam quads incident in a vertex $\mathbf{v}_i$ meet in the common line defined by the normal $\mathbf{n}_i$.

Given the vertex geometry of a general polygon mesh, the beam quads will in general not be planar if the normal vectors for the vertices are computed in the usual manner by summing up the cross products of adjacent incident edge vectors:

$$\mathbf{n}_i^* = \sum_{j \in \mathcal{L}_i} \mathbf{e}_{ij} \times \mathbf{e}_{i(j+1)}, \tag{9}$$

then the beam quads will in general not be planar.

Hence, we need to optimize the normals $\{\mathbf{n}_i\}$ to satisfy the torsion free constraint. This gives us $2\mathfrak{v}$ degrees of freedom (because the length of the normals is irrelevant for the torsion), while the planarity induces $\mathfrak{e}$ constraints. Consequently, it is important to have a large number of vertices relative to the number of edges. This just means the vertex degree, i. e. the number of edges incident on a vertex, should be as small as possible. The smallest useful vertex degree is 3, so we aim at using a polygon mesh with constant vertex degree of 3 or, in other words, a dual triangle mesh. Note that, indeed, the ratio of triangle and edges in a triangle mesh is 2/3, so that this is also the ratio of vertices and edges in a dual triangle mesh. As we have 2 degrees of freedom per constraint, we conclude that dual triangle meshes always have normals that are torsion free.

Based on this observation, our strategy is as follows:

1. Generate a coarse dual triangle mesh that represents the input geometry. We do this by first tessellating the input geometry into featureless faces and then extracting a polygon mesh with straight edges. This step generates the combinatorial structure Δ and a first approximation of the geometry $\{\mathbf{v}_i\}$.
2. Compute beam quads or, equivalently, directions $\{\mathbf{n}_i\}$ that satisfy the constraints. Apart from planarity of the quads we also consider constraints on the height of the elements or the smallest edge lengths. To make these constraints feasible we also allow the initial vertex positions to be updated.

Based on the resulting beam mesh, we compute the details to actually manufacture the shape from a given material with non-zero thickness. This requires additional geometric considerations, in particular for adding joints at the vertex positions. These steps are illustrated in Fig. 3.

4. Base mesh generation

For a given boundary representation of a three-dimensional shape $\mathcal{S}$, for example a triangulated surface, we search for a coarse dual triangle mesh. The goal is to balance the desire for a small number of beams with the required preservation of the prominent features of the shape.

We employ Voronoi diagrams for generating a dual triangle mesh [7,8]. Given locations $\mathbf{x}_i \in \mathbb{R}^3$, the Voronoi diagram is defined by cells Ω_i formed by all points in space that are closest to $\mathbf{x}_i$:

$$\Omega_i = \{\mathbf{p} \in \mathbb{R}^3 : d(\mathbf{x}_i, \mathbf{p}) < d(\mathbf{x}_j, \mathbf{p})\}. \tag{10}$$

Intersecting the Voronoi cells Ω_i against the shape $\mathcal{S}$ results in a tessellation of the surface into faces $\{\Omega_i \cap \mathcal{S}\}$. Generically, three faces meet in one point on the surface, i. e. vertices have degree three, as desired.