Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 163 (2016) 149 – 161

Procedia Engineering

www.elsevier.com/locate/procedia

25th International Meshing Roundtable (IMR25)

Discrete CAD model for visualization and meshing

P. Laug[a,*], H. Borouchaki[a,b]

[a] *Gamma3 project-team, Inria Saclay - Île-de-France, France*
[b] *Gamma3 project-team, University of Technology of Troyes, France*

Abstract

During the design of an object using a CAD (computer aided design) platform, the user can visualize the ongoing model at every moment. Visualization is based on a discrete representation of the model that coexists with the exact analytical representation of the object. Most CAD systems have this discrete representation available, and each of them applies its own construction methodology. This paper presents a new method to build a discrete ("triangulated" with quadrilaterals and triangles) model for CAD surfaces. It presents two major particularities: most elements are aligned with iso-parametric curves and the accuracy of the surface approximation is controlled. In addition, we present a new technique of surface mesh generation that is based on this discrete model. Several examples are presented to confirm the efficacy of this approach.

Peer-review under responsibility of the organizing committee of IMR 25
Keywords: CAD surface; geometric modeling; surface meshing; discrete model; quadtree support.

1. Introduction

In most CAD (computer aided design) systems, any object is defined from its boundary that is constituted by a collection of surface patches. Each patch is defined by a continuous parametric function (typically NURBS) over a bidimensional domain that is called the parametric domain.

A first challenging problem is to visualize such a surface efficiently, in particular on a video display. Most 3D graphics interfaces require a discrete representation of the surface as a set of simple polygons. In this context, the main objective is processing speed, whatever the shape quality of the elements, while preserving the geometric accuracy of the model.

A second problem is to generate meshes for numerical simulations of physical phenomena, applying methods like FEM (finite element method). In this case, the size and shape of the elements must comply with strict specifications tailored to the geometry and the physics of the simulated phenomenon. During the mesh generation, many queries to the CAD system are performed (to evaluate locations of points and also derivatives of parametric functions), which can be time-consuming and difficult to parallelize (due to cache defaults that are involved in most CAD systems).

* Corresponding author. Tel.: +33 1 77 57 80 34.
E-mail address: patrick.laug@inria.fr

Peer-review under responsibility of the organizing committee of IMR 25
doi:10.1016/j.proeng.2016.11.039

In addition, the parametric function representing the surface may vary in a highly irregular manner, and also have degenerate derivatives, resulting in low quality elements.

In this paper, a new method for generating a discrete model (a surface "triangulation" containing quadrilaterals and triangles) for visualization is presented. The model uses a quadtree that follows a network of iso-parametric curves, which generally constitutes an adequate representation of the surface (in particular for polynomial and rational functions). The triangulated model is conformal on each patch. For these reasons, it can also be used as a geometric support for mesh generation. Using the geometric support, the mesher is completely disconnected from the CAD system and is therefore faster and parallelizable. In this case, the surface is redefined by piecewise simple polynomial functions, thus avoiding any degeneracy of derivatives.

In the following, Sec. 2 presents the construction of the discrete model, while Sec. 3 presents a method of using this support to generate meshes. Finally, our approach is illustrated by some examples.

2. Discrete model of a parametric surface

A parametric surface Σ is defined by a regular map σ (having a continuity of class C^1 at least) from a domain Ω of $\mathbb{R}^2$ to $\mathbb{R}^3$:

$$\sigma : \Omega \subset \mathbb{R}^2 \to \Sigma \subset \mathbb{R}^3, \qquad (u, v) \mapsto \sigma(u, v). \tag{1}$$

When the pair of parameters (u, v) sweeps the parametric domain Ω, the point $\sigma(u, v)$ moves on the surface patch Σ. It is not uncommon that a CAD parametric domain is a simple unit square $[0, 1]^2$ but actually it can be can be any shape, possibly having one or more connected components, with possibly inner holes.

The discrete geometric modeling here consists in building a "triangulation" (actually made up of quadrilaterals and triangles) that accurately reflects the geometry of Σ. As in the case of curves, and in general for a variety that is a subset of a global space ($\mathbb{R}^3$ here), the quality of the support is controlled via the two properties of *proximity* and *regularity*. Proximity controls the distance between the elements of the geometric support and the corresponding surface. As for regularity, it controls the angular gap between the elements (or tangent planes to the elements) and tangent planes to the surface.

Iso-u curves (obtained by giving a constant value to u) and iso-v curves (constant v) generally give an accurate description of the geometry of a parametric surface. This mode of representation is particularly suited to polynomial and rational surfaces that are commonly used in CAD systems. To build the geometric support of a patch, we propose a method using a network of iso-parametric curves as privileged directions for the edges. The method consists in building this support indirectly via the parametric domain, as detailed below step by step.

2.1. Discrete model of curves in 2 and 3 dimensions

The first step consists in building a geometric support that is common to the boundary of Ω and its image that constitutes the boundary of Σ. In general, the boundary of Ω is defined by one or more smooth curves (for instance, in the simple case where Ω is a square, its boundary can be defined by four straight edges). Let Γ_2 be such a smooth curve, and let $\Gamma_3 = \sigma(\Gamma_2)$ be its image on the border of Σ. The bidimensional curve Γ_2 is defined by a regular function ω (having a continuity of class C^1 at least) from an interval $[a, b]$ of $\mathbb{R}$ to $\mathbb{R}^2$:

$$\omega : [a, b] \subset \mathbb{R} \to \Gamma_2 \subset \mathbb{R}^2, \qquad t \mapsto \omega(t), \tag{2}$$

and the tridimensional curve Γ_3 is defined by the composition $(\sigma \circ \omega)$ of functions $\omega : \mathbb{R} \to \mathbb{R}^2$ and $\sigma : \mathbb{R}^2 \to \mathbb{R}^3$.

In this case, the support is a curve discretization controlled by both properties of *proximity* (here controlling the distance between the edges of the discretization and the curve) and *regularity* (controlling the angular gap between the edges and the tangents to the curve). Creating the support amounts to first defining the geometric support of the bidimensional curve Γ_2, and then to enrich this support (by recursive subdivisions) so that it is also a geometric support of the tridimensional curve Γ_3.

First, a coarse discretization of Γ_2 is considered, having only one edge if curve Γ_2 is open and two edges if it is closed. Then the discretization is recursively refined while the C^0-distance (proximity) or C^1-distance (regularity) between the curve and the edges of the discretization exceeds a given threshold. The C^0-distance between an edge