Computer-Aided Design 55 (2014) 64–80

Contents lists available at ScienceDirect

Computer-Aided Design

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

Motorcycle graph enumeration from quadrilateral meshes for reverse engineering

Erkan Gunpinar[a,*], Masaki Moriguchi[b], Hiromasa Suzuki[c], Yutaka Ohtake[c]

[a] *Istanbul Technical University, Turkey*
[b] *Chuo University, Japan*
[c] *The University of Tokyo, Japan*

HIGHLIGHTS

- Motorcycle graphs of a given quadrilateral mesh are enumerated.
- Optimum motorcycle graph is found for reverse engineering.
- Highly curved parts of models are placed on motorcycle edges wherever possible.
- Mesh is cut into several sub-meshes.
- Enumeration is performed in each sub-mesh separately.

ARTICLE INFO

Article history:
Received 8 November 2013
Accepted 18 May 2014

Keywords:
Semi-regular quadrilateral mesh
Motorcycle graph
Mesh segmentation
Reverse engineering

ABSTRACT

Recently proposed quad-meshing techniques allow the generation of high-quality semi-regular quadrilateral meshes. This paper outlines the generation of quadrilateral segments using such meshes. Quadrilateral segments are advantageous in reverse engineering because they do not require surface trimming or surface parameterization. The motorcycle graph algorithm of Eppstein et al. produces the motorcycle graph of a given quadrilateral mesh consisting of quadrilateral segments. These graphs are preferable to base complexes, because the mesh can be represented with a smaller number of segments, as T-joints (where the intersection of two neighboring segments does not involve the whole edge or the vertex) are allowed in quadrilateral segmentation.

The proposed approach in this study enumerates all motorcycle graphs of a given quadrilateral mesh and optimum graph for reverse engineering is then selected. Due to the high computational cost of enumerating all these graphs, the mesh is cut into several sub-meshes whose motorcycle graphs are enumerated separately. The optimum graph is then selected based on a cost function that produces low values for graphs whose edges trace a large number of highly curved regions in the model. By applying several successive enumeration steps for each sub-mesh, a motorcycle graph for the given mesh is found. We also outline a method for the extraction of feature curves (sets of highly curved edges) and their integration into the proposed algorithm. Quadrilateral segments generated using the proposed techniques are validated by B-spline surfaces.

1. Introduction

Recently developed quad-meshing techniques can be used to generate high-quality quadrilateral meshes based on user-specified quality constraints. Such meshes are used in various applications such as mesh compression, texture mapping, surface modeling and finite element simulation. A quadrilateral mesh $Q\langle V, E, F\rangle$ consists of a set of quadrilateral faces F with edges E and vertices V. Quadrilateral segments are defined on Q as quadrilateral regions with four corners and four boundaries. In other words, they are isomorphic to an a, b-grid which is a mesh of unit squares in the rectangle $\{(x, y)|0 \leq x \leq a, 0 \leq y \leq b\}$. In this paper, quadrilateral segmentation is used to describe a partitioning of a quadrilateral mesh with quadrilateral segments. The left image of Fig. 1(a) shows a quadrilateral segmentation featuring several quadrilateral segments with boundaries in red and corners in blue.

* Correspondence to: Istanbul Technical University, Mechanical Engineering Faculty, Inonu Caddesi, No: 65, Gumussuyu, 34437, Istanbul, Turkey. Tel.: +90 212 2931300; fax: +90 212 2450795.

E-mail addresses: erkangunpinar@gmail.com, gunpinar@itu.edu.tr (E. Gunpinar).

http://dx.doi.org/10.1016/j.cad.2014.05.007
0010-4485/

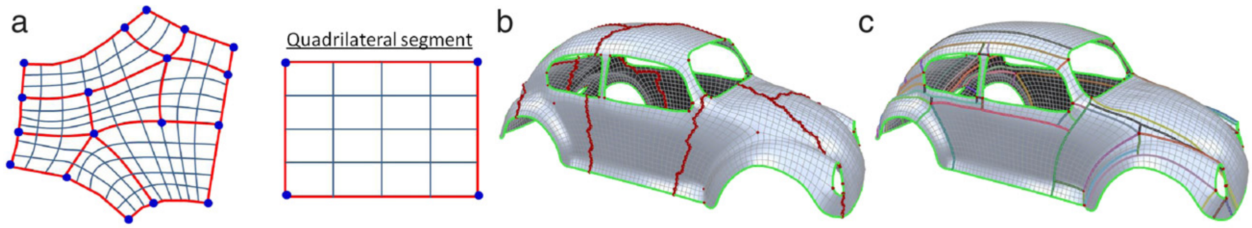

Fig. 1. (a) A quadrilateral segment (right) has four boundaries (in red) and four corners (in blue) consisting of regularly arranged quadrilateral faces. Quadrilateral segments in a quadrilateral mesh are illustrated in the left image in (a). The computational cost of enumerating all motorcycle graphs that can be generated from a quadrilateral mesh (with 48 extraordinary vertices) of the Beetle model is very high. Accordingly, the mesh is cut into several sub-meshes (b) to reduce the time required. Enumeration is performed in each sub-mesh separately with successive steps, and a quasi-optimum graph (c) is obtained within about one second. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The right image depicts a quadrilateral segment consisting of regularly arranged quadrilateral faces.

Motorcycle graphs were first presented in continuous settings and can be used to compute straight skeletons of polygons [1,2]. Later, Eppstein et al. [3] extended this concept to discrete settings (i.e., quadrilateral meshes) in order to generate canonical quadrilateral segmentation (i.e., meshes with the same connectivity that always generate the same quadrilateral segmentation). This paper highlights the concept of motorcycle graphs as a means of generating quadrilateral segments and describes a novel algorithm that enumerates all motorcycle graphs of a given quadrilateral mesh so that one of them can be used for a specific application based on the criteria provided. In this approach, the method of Bommes et al. [4] is used to generate a semi-regular quadrilateral mesh consisting of quadrilateral elements whose vertices mostly have a valence of four (*ordinary vertices*), while the others have other valences (*extraordinary vertices*).

This paper also outlines a methodology for the determination of an optimal motorcycle graph for a given quadrilateral mesh to support reverse engineering. As the placement of highly-curved regions on quadrilateral segment boundaries is a common practice in surface modeling applications, the cost formula introduced for optimum motorcycle graph selection produces small values for motorcycle graphs whose edges trace a large number of highly curved regions of the model. Additionally, feature curves (i.e., sets of highly curved edges) that are very difficult to capture with motorcycle graph edges are also extracted and integrated into the enumeration algorithm. Because of high computational cost of the enumeration algorithm, a more efficient enumeration method (in terms of computational cost) is also proposed. Mesh is cut into several sub-meshes and the optimal motorcycle graph in each sub-mesh is found. By applying successive enumeration steps, a motorcycle graph of a given mesh with proximity to the global optimum is obtained. Enumerating all motorcycle graphs of a Beetle model shown in Fig. 1(b), (c) takes several days according to our tests. Using the proposed approach in this paper, the enumeration takes about one second (see the solution in Fig. 1(c)).

This paper is organized as follows: Section 2 outlines related works, Section 3 details the motorcycle graph enumeration algorithm, Section 4 describes a quadrilateral segmentation technique for reverse engineering based on the enumeration algorithm in Section 3, Section 5 discusses the study's experiments, and Section 6 concludes the work.

2. Related works

The work reported here is related to quadrilateral meshing and mesh segmentation. Selected previous works are covered in this section.

2.1. Quadrilateral meshing

As the use of quadrilateral meshes is very common in a variety of computer graphics applications, quad-meshing has been studied in detail for a number of years. Recently proposed state-of-the-art techniques can be used to produce high-quality quadrilateral meshes whose elements are oriented according to principal curvature directions with good element quality such as square faces. Two review papers [5,6] have detailed quad-meshing methods comprehensively. Early approaches [7,8] define the quadrangulation problem based on the anisotropy of the given object. By tracing lines on the model using principal curvature directions, a quadrilateral mesh is generated from a triangular mesh. Rather than consisting of pure quadrilateral elements, the mesh generated also has triangular faces. The mixed-integer quadrangulation method [4] was motivated by the difficulty of defining principal curvature directions in spherical and flat parts as described in [7]. Accordingly, only certain principal curvature directions are utilized and are interpolated all over the mesh. Quadrangulation is performed using these directions. Quadrilateral meshes generated using this method consist of only quadrilateral elements, most of which are non-degenerated. We also utilize quadrilateral mesh generated using this method. Other recently proposed methods [9–19] can also be applied to produce pure triangle-free quadrilateral meshes using various techniques.

2.2. Quadrilateral mesh simplification

Recent works [20–26] have focused on structure optimization of quadrilateral meshes with the aim of placing extraordinary vertices in appropriate positions to simplify the base complex of the mesh. Myles et al. [22] discussed the alignment problem for extraordinary vertices and simply updated the positions of extraordinary vertices in the parametric domain, thereby achieving a simpler quadrilateral segment layout. In [23], the presence of helical structures in the model is highlighted as a problem of extraordinary vertex alignment. These structures start from an extraordinary vertex, make several turns around the model and stop at extraordinary vertices, thereby producing a not simple quadrilateral segment layout. They are removed by changing the local connectivity using Grid-Preserving operators. In a very recent method proposed in [26], a dual layout is first constructed from curvature-guided crossing loops on the surface, and a quadrilateral mesh is then generated using this layout. The question of how the global structure of quadrilateral meshes can be simplified more remains an ongoing issue, and the proposal of new techniques is expected in the future.

2.3. Mesh segmentation

Mesh segmentation has a long history in the graphics community, and numerous related literature surveys have been conducted [27,28]. A wide variety of algorithms have been proposed for this problem, based on watershed analysis [29,30] hierarchical clustering [30], min-cut/max-flow [31], surface approximation [32], primitive-based clustering [33,34], fuzzy clustering and