Computers & Graphics xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Computers & Graphics

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

ELSEVIER

Special Issue on CAD/Graphics 2017

Untrimming: Precise conversion of trimmed-surfaces to tensor-product surfaces

Q1 Fady Massarwi*, Boris van Sosin, Gershon Elber

Q2 *Department of Computer Science, Technion, Israel*

Q3

ARTICLE INFO

Article history:
Received 15 June 2017
Revised 10 August 2017
Accepted 13 August 2017
Available online xxx

Keywords:
Composition
Line-sweep quadrangulation
Optimal quadrangulation
Precise integration
Precise bounding box

ABSTRACT

Trimmed B-spline surfaces are common in the geometric computer aided design (CAD) community due to their capability to represent complex shapes that can not be modeled with ease using tensor product B-spline and NURBs surfaces. However, in many cases, handling trimmed-surfaces is far more complex than tensor-product (non-trimmed) surfaces. Many algorithms that operate on tensor-product surfaces, such as algorithms toward rendering, analysis and manufacturing, need to be specially adapted to consider the trimming domains. Frequently, these special adaptations result in lack of accuracy and elevated complexity. In this paper, we present an algorithm for converting general trimmed surfaces into a set of tensor-product (typically B-spline) surfaces. We focus on two algorithms to divide the parametric space of the trimmed surface into four-sided quadrilaterals with freeform curved boundaries, which is the first step of the algorithm. Then, the quadrilaterals are parameterized as planar parametric patches, only to be lifted to the Euclidean space using a surface-surface composition, resulting in tensor product surfaces that precisely tile the input trimmed surface in Euclidean space. The algorithm is robust and precise. We show that we can handle complex, industrial level, objects, with numerous high orders and rational surfaces and trimming curves. Finally, the algorithm provides user control on some properties of the generated tensor-product surfaces.

1. Introduction

Tensor product (Bézier, B-spline and NURBs) surfaces are widely used in geometric computer aided design (CAD) due to their simple structure, mathematical form, and powerful geometrical properties that make them intuitive to use. However, they are limited to the rectangular topology, making it difficult to create general 3D objects. That is, the rectangular topology does not allow to represent with ease general boundaries, including holes. Due to these limitations of the tensor product surface representation, trimmed-surfaces were introduced [1]:

Definition 1.1. A *trimmed B-spline surface*, S_t, is a tensor-product B-spline surface, S, whose domain is bounded by a set of trimming B-spline closed curves, $\mathcal{C}_t$. Typically, one, outer boundary trimming curve exists, and other internal trimming curves define holes in the parametric domain. The orientation of the trimming curves is defined such that the trimmed-surface lies on the same side (e.g right) of the trimming curves, as we move along the trimming curve.

A common method for creating CAD models is by applying Boolean set operations between simpler models [2–4], where the intersection curves between the surfaces of the models define the trimming curves. In the ensuing discussion and unless otherwise stated, we will refer to a trimmed B-spline surface/curve while it can also be a Bézier or a NURBs surface/curve.

Compared to tensor-product surfaces, trimmed-surfaces ease the process of representing results of Boolean set operations, and allow simpler modeling of complex shapes. However, there are difficulties in using trimmed-surfaces compared to tensor product surfaces. Due to the complex parametric boundaries and the non-rectangular topology, powerful geometrical properties of the B-spline representation, such as the convex hull property [1], are less faithful to trimmed-surfaces than to tensor-product surfaces. Algorithms designed for tensor product B-spline surfaces, such as algorithms toward rendering, manufacturing and analysis, do not directly extend to trimmed-surfaces and require special treatments, if at all feasible.

A recent development in physical analysis, Iso Geometric Analysis (IGA) [5], performs the analysis directly in spline spaces over the spline surfaces of the models, which practically means models with trimmed-surfaces. IGA requires precise integration over the surfaces, among others. However, integration over trimmed

* Corresponding author.
E-mail addresses: fadym@cs.technion.ac.il, fadyam8@gmail.com (F. Massarwi).

http://dx.doi.org/10.1016/j.cag.2017.08.009
0097-8493/

B-spline basis functions is a challenging non-trivial task, in the general case. Approximating trimmed-surfaces by piecewise-linear elements, in order to simplify the integration process, will result in loss of accuracy and might affect the quality and convergence of the analysis. Methods to precisely integrate over the trimming domains are required, in order to have a complete and accurate IGA over trimmed-surfaces. One way to achieve this goal, is by first converting the trimmed-surfaces to tensor-products. In this work, we present the untrimming process not only as a geometry conversion process but also as an intermediate representation to precisely integrate over trimmed domains and hence is a precise fit to the IGA approach, for trimmed surfaces.

In this paper, we introduce an algorithm with two variations for converting general trimmed-surfaces into a set of tensor product B-spline surfaces, a process we denote as *untrimming*. The algorithm is robust and precise[1], and is able to handle complex industrial models composed of thousands of trimmed-surfaces. The algorithm first divides the trimmed parametric domain into quadrilaterals with freeform boundary curves while precisely preserving the trimming curves. Then, the quadrilaterals are parameterized into planar patches. Finally, these (tensor-product planar) patches are lifted to the Euclidean space via a symbolic surface-surface composition [6–8]. The end result is a precise tiling of the original trimmed surface, by tensor product surfaces, albeit of higher degrees. The main contribution of this work includes two variations of a quadrangulation algorithm of freeform (trimmed) domain. The first variation of the algorithm builds the quadrangulation using a fast line-sweep based algorithm, and the second variation builds the quadrangulation based on the minimization of a given weight function, which enables some control over different desired properties of the generated output. The untrimming algorithm can handle rational and arbitrary order trimming curves and trimmed-surfaces.

We like to emphasize that the presented conversion is precise for each individual trimmed surface. If cracks (black holes) exist (i.e. due to imprecise Boolean Set operations) between different trimmed surfaces, these cracks will be precisely reconstructed, as this work focus on the precise reconstruction of individual trimmed surfaces as tensor products.

The rest of this document is organized as follows. Section 2 discusses related work and in Section 3, we describe our untrimming algorithm, with its two variations. In Section 4, we present some results of untrimming of several trimmed surfaces' domains and CAD models, and compare the different proposed methods. Finally, in Section 5, we conclude and discuss future planned research.

2. Related work

Several studies have proposed algorithms for generating quad meshes, such as [9–12]. However, these algorithms have been developed for triangular surface meshes, and are not easily adapted to trimmed B-spline surfaces with high-order B-spline trimming curves. A method for converting trimmed NURBs surfaces to Catmull–Clark subdivision surfaces is described in [13]. The method in [13] is limited to bi-cubic NURBs.

Other studies focused on rendering of trimmed-surfaces. Schollmeyer et al. [14] proposed a fast and direct method for rendering trimmed-surfaces that is aimed to avoid the inaccuracies introduced if the trimming curves are not precise. Martin et al. [15] proposed a ray tracing algorithm for trimmed-NURBs and provided an algorithm for ray-NURBs intersection that is based on hierarchical pruning and numerical refinements. Both methods, [14] and [15], exploit algorithms for a point inclusion test in the trimmed parametric domain. However, these methods allow a pixel error approximation in the trimming curve point inclusion test, and thus appropriate for rendering only. Further, it is unclear how can these methods be extended to precisely handle trimmed surfaces, for general, non-rendering, tasks.

Approximating trimmed-surfaces by a set of primitives have been studied, for example, in [16–18], where the challenge is to minimize the number of approximating triangles with respect to a user defined error tolerance. A common problem when tessellating trimmed surfaces, is the generation of cracks and gaps along common trimming boundaries between neighboring trimmed-surfaces (also known as "black holes"). Several studies have addressed the cracks problem [19,20] and suggested methods for fixing the tessellation errors and stitching the cracks. The cracks' problem could have potentially been avoided if the trimming of the trimmed surface has been precise and the surface is precisely converted to a set of tensor product surfaces. Unfortunately, the computation of the surface-surface intersection curves, as part of Boolean set operations, are rarely within machine precision.

The conversion of trimmed-surfaces into tensor-product surfaces have been studied in [6,21–24]. Hoschek and Schneider [22] uses curvature oriented segmentation in order to obtain bicubic Bézier patches, but [22] can not handle rational trimmed surfaces, and approximate them by a bi-cubic or bi-quintic polynomial surfaces. Further, the mapping process of the resulted quadrilaterals from the parametric domain to the Euclidean space is not precise and utilizes interpolation methods. Li and Chen [24] partitions the parametric space into quadrilaterals using feature points of the trimming curves, but it is designed to be precise only for bicubic polynomial B-spline surfaces, in an effort to reduce the degrees of the outcome. The trimming domain is partitioned in *turn* points, locations on the trimming curve $C_t(t) = (u(t), v(t))$ that satisfies $|u'(t)| = |v'(t)|$, and results in over-partitioning of the domain. Further, a closed piecewise C^1 discontinuous trimming curve may have no location for which $|u'(t)| = |v'(t)|$, cases that are not discussed in [24] while they are handled in this work. Hamann et al. [21] employs a scan-line based algorithm for partitioning the parametric space to a rectangular domains. However, their method involves triangulation and Voronoi-diagram computation over the trimmed-parametric domain, which makes it less robust and complex to implement. Also, [21] assumes that the ruling between any two monotone regular, (non vanishing derivative), curves always produces a regular (consistent Jacobian) surface, which we show to not necessarily be the case (and also show a remedy). Hui et al. [23] also employs a Voronoi-diagram approach for partitioning the trimmed domain into simpler cells, and improves the method proposed in [21] by using feature point matching approach rather than a scan-line approach in order to reconstruct four-sided surfaces. However, [23] does not provide methods for mapping the resulted tensor-product surfaces from the parametric space to the Euclidean space. Finally, while [22–24] recognize the importance of only regular patches in the output, they do not discuss how to achieve this goal.

In [6] several applications of functional composition of B-spline curves and surfaces have been introduced. Following [6], we use a symbolic surface-surface composition as the final step to lift the generated quadrilaterals from the parametric space to the Euclidean space. Elber and Kim [6] discusses an algorithm for converting a trimmed-surface to tensor-product surfaces. However, the algorithm in [6] does not offer a general quadrangulation and hence is limited to simple topologies and can not handle industrial level trimmed surfaces. None of the above methods is capable of precisely handling general models with high order rational trimming curves and trimmed-surfaces. Also, previous methods, with the exception of [6], do not involve general symbolic computation to provide precise and robust partition of the parametric do-

[1] In this work, precise denotes a precision that approaches the accuracy of the hardware (machine precision).

Please cite this article as: F. Massarwi et al., Untrimming: Precise conversion of trimmed-surfaces to tensor-product surfaces, Computers & Graphics (2017), http://dx.doi.org/10.1016/j.cag.2017.08.009