Computers & Graphics xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Computers & Graphics

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

Special Section on VRIPHYS 2017

Volumetric subdivision for consistent implicit mesh generation

Q1 Christian Altenhofen[a,b,*], Felix Schuwirth[a,b], André Stork[a,b], Dieter Fellner[a,b,c]

[a] TU Darmstadt, Interactive Graphics Systems Group, Fraunhoferstr. 5, Darmstadt 64283, Germany
Q2 [b] Fraunhofer IGD, Fraunhoferstr. 5, Darmstadt 64283, Germany
[c] Institute of Computer Graphics and Knowledge Visualization, Graz University of Technology, Inffeldgasse 16c, Graz 8010, Austria

A R T I C L E I N F O

Article history:
Received 13 June 2017
Revised 12 September 2017
Accepted 20 September 2017
Available online xxx

Keywords:
Volumetric modeling
Volumetric subdivision
Mesh generation
Tetrahedral meshing
Physically-based simulation
Finite element method

A B S T R A C T

In this paper, we present a novel approach for a tighter integration of 3D modeling and physically-based simulation. Instead of modeling 3D objects as surface models, we use a volumetric subdivision representation. Volumetric modeling operations allow designing 3D objects in similar ways as with surface-based modeling tools, while automatic checks and modifications of inner control points ensure consistency during the design process. Encoding the volumetric information already in the design mesh drastically simplifies and speeds up the mesh generation process for simulation. The transition between design, simulation and back to design is consistent and computationally cheap. Since the subdivision and mesh generation can be expressed as a precomputable matrix-vector multiplication, iteration times can be greatly reduced compared to common modeling and simulation setups. Therefore, this approach is especially well suited for early-stage modeling or optimization use cases, where many geometric changes are made in a short time and their physical effect on the model has to be evaluated frequently. To test our approach, we created, simulated and adapted several 3D models. We measured and evaluated the timings for generating and applying the matrices for different subdivision levels. Additionally, we computed several characteristic factors for mesh quality and mesh consistency. For comparison, we analyzed the tetrahedral meshing functionality offered by CGAL for similar numbers of elements. For changing topology, our implicit meshing approach proves to be up to 70 times faster than creating the tetrahedral mesh only based on the outer surface. Without changing the topology and by precomputing the matrices, we achieve a speed-up of up to 2800, as all the required information is already available.

1. Introduction

In computer graphics and animation, subdivision surfaces are widely used to create visually appealing 3D models. When aiming for a plausible physical behavior of those models for 3D animation or games, physically-based simulation comes into play. In many cases, it takes several loops of design and simulation to adjust a 3D geometry so that it shows the intended behavior in the simulation. Although subdivision surfaces have proven to be useful tools in animation, they show some of the same hurdles as other representation schemes to be overcome for simulation. Usually, the geometric mesh has to be transformed into a volumetric mesh to enable simulation (hereinafter referred to as *meshing* process). As the surface mesh just describes the outer boundary of the object, volumetric meshing tends to be a time-consuming process that might even require manual interaction, often to be re-done with every geometric change. Since there is no direct correlation between the design mesh and the simulation mesh, feedback and conclusions from the simulation results have to be derived manually to improve the design.

While the engineering community tries to solve this problem with *Iso Geometric Analysis (IGA)* presented by Hughes et al. [1] in 2005, we propose a method more suited for computer animation. By using subdivision volumes instead of subdivision surfaces for creating the initial 3D object, we encode the volumetric information into the model directly in the design phase. Volumetric modeling operations allow the manipulation of the geometry similarly to existing modeling tools, which use subdivision surfaces, while at the same time keeping the volumetric representation consistent underneath. For its diversity and its ability to handle control meshes of arbitrary topology, we chose the Catmull–Clark solid

* Corresponding author at: Fraunhofer IGD, Fraunhoferstr. 5, 64283 Darmstadt, Germany.
E-mail address: christian.altenhofen@igd.fraunhofer.de (C. Altenhofen).

https://doi.org/10.1016/j.cag.2017.09.005

subdivision scheme by Joy and MacCracken [2] for our approach. Since the outer limit surface is identical to the limit surface of Catmull–Clark subdivision surfaces as presented by Catmull and Clark [3] in 1978, there is no visual difference when designing 3D models with either Catmull–Clark surfaces or solids. For simulating, the volumetric subdivision scheme is applied to the control mesh multiple times until the desired mesh resolution is reached. Afterwards, the mesh can be converted into a purely tetrahedral mesh if necessary. To run the simulation, we use a custom GPU-based FEM solver based on the method presented by Weber et al. [4] in 2013. However, as we are able to create a hexahedral or tetrahedral mesh, it does not require a special solver to be simulated and could also be fed into open-source or commercial solvers as well. Due to the initial volumetric representation, many vertices are shared between the design mesh and the simulation mesh and the simulation results can be visualized (also volumetrically) directly on the design mesh. This allows for clear hints on where to adapt/improve the model if necessary.

Since both the subdivision steps along with the conversion into a tetrahedral mesh can be expressed as one precomputable matrix-vector multiplication, our meshing process is much faster than those of commonly used meshing tools such as CGAL [5] or TetGen [6] and therefore allows much faster iterations of design and simulation. Additionally, due to the volumetric structure and the choice of the volumetric modeling operations, the mesh is guaranteed to be manifold except for self-intersections. These are resolved in an automatic smoothing step, averaging the positions of the inner control points based on their neighbors. At the same time, this smoothing improves the overall quality of our simulation mesh.

Our main contributions are:

- A new approach for generating 3D models suited for design and simulation alike.
- A tighter integration of modeling and simulation with shorter iteration times and more insightful feedback.

These lead to the following benefits:

- Fast and consistent meshing due to an existing volumetric structure and precomputable mesh generation matrices.
- A partly relation between the geometric model and the simulation results.
- An ensured manifold volumetric mesh representation while modeling, automatically resolving self-intersections of inner vertices.

This paper is an extension to the conference paper “Implicit Mesh Generation using Volumetric Subdivision” by Altenhofen et al. [7] as presented at the 13th Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS 2017. The extensions focus on evaluating and improving the quality and consistency of the simulation meshes created by our implicit mesh generation approach. Mesh uniformity in terms of cell sizes and angles as well as inconsistent topologies due to self-intersections are discussed. We present a modified version of Laplacian smoothing on inner control points to resolve self-intersections and even out cell sizes and angles. Additionally, we show an alternative approach for converting the volumetric subdivision mesh into a tetrahedral mesh, using adapted Catmull–Clark subdivision rules also in this step.

The paper is structured as follows: Section 2 summarizes existing work that is related or fundamental to our approach. Section 3 explains the approach in detail, showing its individual components and their connections. In Section 4, we describe our prototypical implementation. Section 5 shows the accomplished results, as well as benchmarks for the major steps and comparison with other algorithms. Section 6 wraps up the paper, reflects on our results and points out the advantages and possibilities for improvements of our approach. Finally, Section 7 gives an outlook on future improvements.

2. Related work

This section overviews existing methods that are related to our approach and briefly discusses their benefits and drawbacks.

2.1. Modeling

In the field of modeling three-dimensional objects, the main categories are computer-aided design software (CAD) like Solid Works [8], Rhino [9] or Fusion 360 [10] and polygonal modeling known from tools like Maya [11] or Blender3D [12].

CAD software in general uses implicit volumetric representations such as BReps or parametric surface descriptions such as Bézier, B-spline or NURBS surfaces. The latter define smooth surfaces by a set of control points and are well suited for engineering applications. However, they are hard to use when aiming for organic shapes such as often required in computer graphics and animation.

In the area of design for computer graphics, animation and games, polygonal modeling tools are very common because of their easy-to-use modeling techniques. In contrast to CAD software, polygonal modeling tools like Blender3D offer subdivision surface algorithms to design organic 3D models. They provide a control mesh with a relatively low amount of degrees of freedom to model a smooth limit surface (see Section 2.2). A commercial example for heavily using subdivision techniques in different areas of 3D modeling and computer animation is Pixar [13]. Nevertheless, all these tools typically offer surface modeling only.

In terms of volumetric modeling, there are only few approaches. Fairly new modeling techniques arise with the field of additive manufacturing, using modeling techniques on a voxel basis. An example is the software Monolith [14] which is a voxel-based modeling engine for multi-material 3D printing. It can describe inner structures and material properties. Another volumetric modeling technique is sculpting presented by McDonnell et al. [15] in 2001, which allows the designer to initially model a rough shape and then define details like a sculptor by adding and removing parts locally from the shape. Analogously to subdivision surfaces, subdivision volumes exist, but they are mostly used in the context of simulation as described in Section 2.2.

2.2. Subdivision

Subdivision surfaces are widely used in computer graphics and computer animation. They provide smooth surfaces while at the same time only having a small number of degrees of freedom to define those. Due to the iterative or precomputable refinement process, memory consumption for a subdivision-based 3D object is much lower than for a finely tessellated model. It also allows for dynamic tessellation and level-of-detail approaches to improve the performance for rendering. For many years, different subdivision schemes have been developed. Some of them require purely triangular control meshes like the ones by Loop [16] and Dyn et al. [17], while others work on quad-based meshes or are able to handle control meshes with arbitrary topology by authors such as Doo [18] and Catmull and Clark [3]. Depending on the subdivision scheme and the topology of the control mesh, the limit surface has different continuity C (C^0, C^1, C^2).

As an extension to the existing subdivision schemes for surfaces, volumetric subdivision algorithms have been developed. They are mostly used in the engineering environment for global or local refinement of the simulation mesh as described by

Please cite this article as: C. Altenhofen et al., Volumetric subdivision for consistent implicit mesh generation, Computers & Graphics (2017), https://doi.org/10.1016/j.cag.2017.09.005