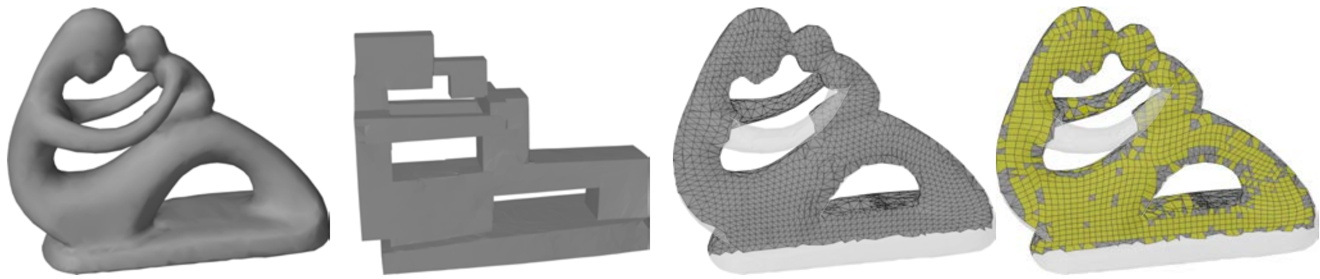
PolyMerge: a fast approach for hex-dominant mesh generation

C. Antonio Sánchez* and Sidney Fels*
University of British Columbia, Vancouver, Canada



(a) *Initial Surface* **(b)** *PolyCube* **(c)** *Tetrahedralization* **(d)** *Final Hex-Dominant Mesh*

Figure 1: *An initial surface mesh (a) is deformed to fit its nearest PolyCube (b) using a finite-element framework. A grid of new nodes are distributed in the PolyCube, then mapped back to the original volume. A tetrahedral mesh is generated (c) using the new nodes and the original surface. This is converted to a hex-dominant mesh (d) by merging groups of tetrahedra.*

1 Introduction

Hex-dominant meshes are often preferred for finite element simulations due to their superior convergence properties and lack of numerical artifacts (e.g. locking). Unfortunately, constructing such meshes for arbitrary shapes is still a challenging task. One recent approach is to use a low-distortion map between a target shape and its nearest PolyCube form (union of rectangular prisms) [Livesu et al. 2013]. A hex mesh can be constructed quite easily in the deformed space, then transformed back to the original coordinate system. However, the distortion may be quite large, leading to poorly-conditioned elements (highly stretched/skewed). Another option to is to create an initial tetrahedral mesh, then convert this to a hex-dominant one by merging groups of tetrahedra. In [Yamakawa and Shimada 2003], a graph search is used to identify potential tet groupings that can be merged to form hex elements. Finding hexes this way leads to a combinatorial problem, which is computationally expensive. It also suffers from redundant searches, checking the same patterns of tets multiple times.

We present a new hex-meshing strategy that addresses these deficiencies. Our tet-merging algorithm is up to $40\times$ faster than the current state-of-the-art, allowing us to generate hex-dominant meshes (80% by volume) with more than 100k elements in less than one minute.

2 Approach

We harness the benefits of both approaches by combining PolyCube deformation to introduce structure, with a new tetrahedral-merging algorithm to generate the final hex-dominant mesh (Figure 1). To deform the shape to a PolyCube, we set up a quadratic minimization problem that includes: (1) a penalty term that aligns each face with its nearest coordinate plane; and (2) a finite-element strain energy term that minimizes total deformation. This leads to a linear system, which we iterate until the shape converges. We distribute new nodes inside the PolyCube in a grid pattern, then transform them back to the original coordinate system. Using the original surface and the new nodes, we build a well-conditioned tetrahedral mesh via Delaunay triangulation. This eliminates the issue of poorly-conditioned elements due to high PolyCube distortion.

By examining all possible ways of splitting a hexahedron into tetrahedra, we have identified a set of seven unique patterns (plus their rotated/flipped versions) on which all potential hexes can be formed without adding or removing nodes. The central component to each of these patterns is either a tetrahedron in the centre ($\times 2$), or a diagonal edge connecting opposite corners of the hex with 3, 4($\times 2$), 5 or 6 tetrahedra sharing it. The algorithm proceeds by looping through each tet to search for the central-tet patterns; then looping through each edge to search for all diagonal-based patterns. With this algorithm, we find all potential hexes without a graph search, and with no redundancies. Once a potential hex is identified, it is added to a priority queue sorted by quality. When emptying the queue, the next hex is added if: (a) all its tets still exist in the mesh; and (b) certain face conformity constraints are met.

We are currently able to produce meshes that are approximately 80% hexes by volume. Since we do not modify the original surface mesh, this is where the remaining tets are concentrated. Our next step is to re-mesh the surface, again based on the PolyCube map, to make it more compatible with the interior nodes.

*email: {antonios, ssfels}@ece.ubc.ca

SIGGRAPH 2014, August 10 – 14, 2014, Vancouver, British Columbia, Canada.

ACM 978-1-4503-2958-3/14/08

References

LIVESU, M., VINING, N., SHEFFER, A., GREGSON, J., AND SCATENI, R. 2013. Polycut: Monotone graph-cuts for polycube base-complex construction. *ACM Trans. Graph. 32*, 6 (Nov.), 171:1–171:12.

YAMAKAWA, S., AND SHIMADA, K. 2003. Fully-automated hex-dominant mesh generation with directionality control via packing rectangular solid cells. *Int. J. for Num. Meth. in Eng. (IJNME) 57*, 15, 2099–2129.