Approximate Convex Decomposition*

Jyh-Ming Lien Nancy M. Amato
{neilien,amato}@cs.tamu.edu
Parasol Lab., Department of Computer Science
Texas A&M University

1. INTRODUCTION

Decomposition is a technique commonly used to simplify complex models into smaller sub-models that are easier to handle. Convex decomposition divides models into convex components which are of interest because many algorithms perform more efficiently on convex objects than on non-convex objects.

One issue with convex decompositions, however, is that they can be costly to construct and can result in representations with an unmanageable number of components. In many applications, the detailed features of the model are not crucial and in fact considering them only serves to obscure important structural features and adds to the processing cost. In such cases, an approximate representation of the model, such as our proposed approximate convex decomposition, that captures the key structural features would be preferable.

2. OUR APPROACH

In this work, we propose a partitioning strategy that decomposes a given model into *"approximately convex"* pieces. Our motivation is that for many applications the approximately convex components of this decomposition provide similar benefits as convex components, while the resulting decomposition is both significantly smaller and can be computed more efficiently.

Our goal is to generate τ-*approximate* convex decompositions. For a given model P, P is said to be τ-*approximate* convex if $\text{concave}(P) < \tau$, where $\text{concave}(\rho)$ denotes the concavity measurement of ρ. Here, τ represents a tunable parameter denoting the non-concavity tolerance for the application. A τ-*approximate* convex decomposition of P, $\text{CD}_\tau(P)$, is defined as a decomposition $D(P)$ that contains only τ-*approximate* convex components; i.e.,

$$\text{CD}_\tau(P) = \{C_i \mid C_i \in \text{D}(P) \text{ and } \text{concave}(C_i) \leq \tau\}. \quad (1)$$

The significance of a feature is measured by its concavity. We recursively remove (resolve) concave features in order of decreasing significance until all remaining components have concavity less than some desired bound.

Let $\text{retract}(x, H_P, t) : \partial P \to H_P$ denote the function defining the trajectory of a point $x \in \partial P$ when x is retracted from its original position to ∂H_P, where H_P denotes the convex hull of P. Assuming that this retraction exists for x, $\text{dist}(x, H_P)$ is the integration of the function $\text{retract}(x, H_P, t)$ from $t = 0$ to 1. We define the concavity measurement of x as:

$$\text{concave}(x) = \text{dist}(x, H_P) \quad (2)$$

Then, the concavity of P is defined as the maximum concavity of its vertices.

We proposed three methods to compute the concavity of x, straight-line concavity (SL), shortest path concavity (SP), and hybrid concavity (HC). SL-concavity measures the straight line distance from x to H. This straight line may intersect the boundary ∂P of P. SP-concavity measures the shortest path distance from x to H without intersecting ∂P. The hybrid concavity measure uses SL-concavity as the default, but uses SP-concavity when SL-concavity may fail to ensure monotonically decreasing concavity measurements.

For polygons, a notch (concave feature) x must be enclosed by exactly one line segment β of the convex hull. To compute the concavity, we simply measure the retraction distance from x to β. For polyhedra, a notch x may be enclosed by more than one facet of the convex hull. To identify which facet x should be retracted to, we project facets onto P. Vertices under the projection of a facet will be retracted to that facet. See Figure 1.

After the concavity is measured, the model will be decomposed if it concavity is untolerable. To decompose a polygon, a diagonal is added to the vertex with maximum concavity. To decompose a polyhedron into solid parts,

*This research supported in part by NSF Grants ACI-9872126, EIA-9975018, EIA-0103742, EIA-9805823, ACR-0081510, ACR-0113971, CCR-0113974, EIA-9810937, EIA-0079874, and by the Texas Higher Education Coordinating Board grant ATP-000512-0261-2001.

SCG'04, June 8–11, 2004, Brooklyn, New York, USA.

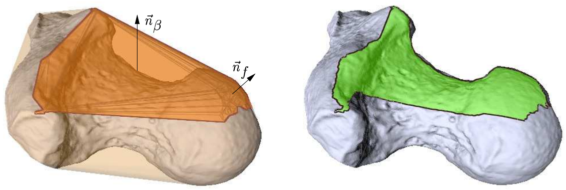


Figure 1: Associating convex hull facets (left) with vertices (right) of the polyhedron P. A set of facets are grouped and projected onto P together.

a cut plane incident to the most concave notch will bisect the model. To decompose a polyhedron into approximately convex surfaces, we identify concave features from boundary lines of convex hull facet projections. Figure 2 shows this process. Then, these features are connected through regions of high concavity. The model is decomposed along the path that connects these features. See Figure 3.

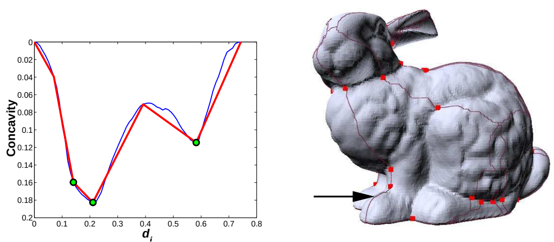
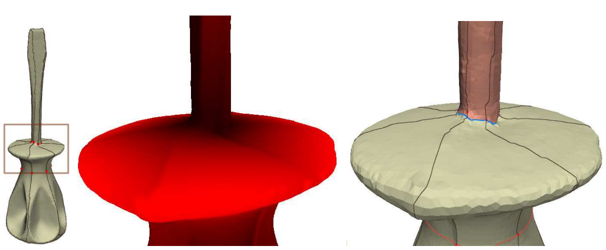


Figure 2: A line (indicated by an arrow) on the Stanford Bunny is mapped to the plot (left). Features are identified from this line and are marked as dots. Red dots on the bunny indicate all features found.



(a) (b) (c)

Figure 3: (a) Red dots are identified features. (b) Darker area indicate high concavity regions. (c) Features are connected through high concavity areas. The model is then decomposed along the connection.

More detailed information about our algorithm can be found in [1, 2].

3. RESULTS

In the submitted video, we demonstrate the results of approximate decomposition for polygons with or without holes and, for polyhedra, we show that our strategy can be applied to both solid and surface convex components. From our experimental results, we observe that if an application is willing to sacrifice a little convexity, then our algorithm can produce fewer components than the optimal approach in less time. Figure 4 shows the difference between exact and approximate convex surface decomposition.

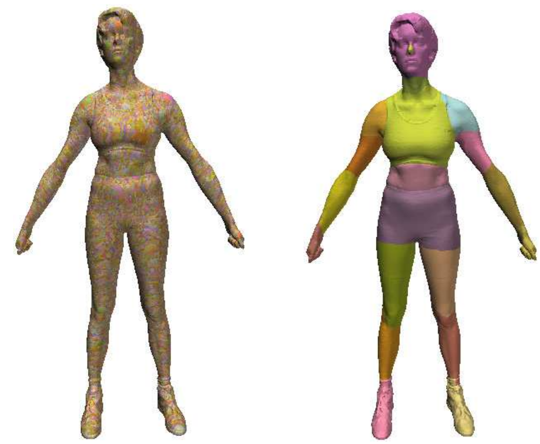


Figure 4: This model has 243,442 triangles and 141,837 notches. Left: Exact convex surface decomposition – 44,461 components. Right: Approximate convex surface decomposition – 20 components with concavity < 0.01.

Another important feature of approximate convex decomposition is its ability to reveal significant structural information about a given model. For instance the Stanford Bunny and the elephant model in Figure 5 are decomposed into sub-models that reflect anatomical structures.

Figure 5: Both models are decomposed into 0.1-approximate convex components. The elephant is decomposed into 14 components and the bunny is decomposed into 10 components.

References

[1] J.-M. Lien and N. M. Amato. Approximate convex decomposition. Technical report tr03-01, PARASOL Lab., Dept. of Computer Science, Texas A & M University.

[2] J.-M. Lien and N. M. Amato. Approximate convex decomposition for polygons. In *Proceedings of the 20th Annual ACM Symposium on Computational Geometry (SoCG'04)*, pages 173–182, June 2004. to appear.