社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2005－CG－120（9）
2005／8／19

Edge Collapse と Sizing Function を用いた等方的メッシュへの簡略化法

森口 昌樹　　杉原 厚吉

東京大学大学院 情報理工学系研究科
数理情報学専攻
{Masaki_Moriguchi, sugihara}@mist.i.u-tokyo.ac.jp

本研究では高密度メッシュから，指定された頂点数を持つ等方的な簡略化メッシュを作成する方法を提案する．提案法ではまず，高密度メッシュに edge collapse 操作を繰り返し適用して簡略化メッシュを作成する．このとき，sizing function（簡略化メッシュの理想的な辺長を表す関数）を考慮に入れた辺長をコスト関数に用いることにより，比較的等方性の高いメッシュを高速に作成することができる．そしてそのメッシュを元にして，高密度メッシュ上に制約付き重心ボロノイ図を作成することにより，高品質な簡略化メッシュを作成する．本手法は特徴の保存を行うことができ，任意のトポロジーを持つメッシュを簡略化することもできる．

Combining Edge Collapse Heuristics and Sizing Functions for Isotropic Mesh Coarsening

Masaki Moriguchi and Kokichi Sugihara

Department of Mathematical Informatics,
Graduate School of Information Science and Technology,
The University of Tokyo

In this research, we propose an efficient mesh simplification algorithm which simplifies dense triangular meshes into isotropic meshes conforming to given vertex budgets and sizing functions. First, dense meshes are simplified to moderate quality isotropic meshes by a sequence of edge collapse operations until the desired vertex count is achieved . We measure the cost by the edge length with respect to the sizing function. This enables to efficiently produce moderate quality isotropic meshes. Then, the quality of these meshes is further refined using constrained centroidal Voronoi diagrams. Our algorithm can preserve surface features and can process meshes with arbitrary topology.

1　はじめに

三角形メッシュは，コンピュータグラフィックスや数値計算などにおいて，３次元モデルを表現する手段として広く用いられている．そして有限要素法などの数値計算においては，高品質なメッシュが非常に重要な役割を果たす．本研究は表面メッシュを対象としており，メッシュの品質は三角形の形・大きさで測られる．三角形の形は正三角形に近い（等方的）ほど質が高く，大きさに関しては，辺の長さが sizing function で指定されている辺長に近いほど質が高い．

近年では３次元スキャナなどの技術の進歩により，数千万個ものポリゴンを持つ非常に高密度なメッシュが得られるようになってきた．しかしそのようなメッシュは非常に複雑なので，処理にかかる計算コストが膨大になってしまう．また３次元スキャナなどから作成されたメッシュは，細長い三角形を数多く持っているなど，質が悪い．そのため，

高密度メッシュから質の良い簡略化メッシュを作成することが望まれており，我々はその効率的なアルゴリズムを提案する．

メッシュの簡略化には，

- 簡略化メッシュの要素数を固定し，その中で出来るだけ誤差の少ない・質の良いメッシュを作成する，
- 簡略化メッシュにおける誤差・質の閾値を定めて，その範囲内で出来るだけ要素数の少ないメッシュを作成する，

という二通りの方針があるが，本研究では前者を採用する．つまり，指定された数の頂点数を持つ簡略化メッシュを作成することを目的としており，誤差や質の保証はない．

提案法はまず，高密度メッシュに sizing function を利用した edge collapse 簡略化法を適用して，比較的質の良い簡略化メッシュを作成する．そしてそのメッシュを元に，高密度メッシュ上に制約付き重心ボロノイ図を作成することにより，高品質な簡略化メッシュを作成する．本研究は，[16] のように，入力が高密度な三角形メッシュで，出力は入力メッシュと比較して十分に小さな要素数のメッシュ（例えば入力メッシュの数十分の一の頂点数を持ったメッシュ）である場合を想定している．

2 Sizing Function を利用した Edge Collapse 簡略化法

この節では edge collapse および sizing function の説明を行った後に，我々が提案するこれらを組み合わせた簡略化法について述べる．

2.1 Edge Collapse

Edge collapse とは辺を一点に縮約させる操作のことを言い，縮約した辺は頂点となり，その辺に接続していた全ての三角形は消去される [10]．Edge collapse でつくられる新しい頂点の位置が，もとの二つの頂点（縮約される辺の二つの頂点）位置に制限されているものは half edge collapse と呼ばれる．Edge collapse を繰り返し適用しメッシュの簡略化を行う方法は，メッシュ簡略化において最も多く用いられている．さらに，その中の大部分は次のような貪欲アルゴリズムを用いている [9, 8]．

1. 全ての辺に対して edge collapse のコストを計算し，優先順序付きキューに挿入する．
2. 最小のコストを持つ edge collapse をキューから取り出し，実行する．
3. 縮約された辺の近傍において，edge collapse のコストを再計算する．
4. 指定された要素数が達成されるまで，もしくは誤差が閾値をこえるまで，ステップ 2・3 を繰り返す．

ここで，コストとは edge collapse を適用することによって引き起こされる誤差を評価する関数である．

多くの場合，edge collapse によってつくられる新しい頂点はコスト関数を最小にする位置に置かれる．また，どのようにコストを計算するかで，アルゴリズムの特徴付けを行うこともできる．

2.2 Sizing Function

Sizing function は，高密度メッシュ上で定義される関数で，その点における簡略化メッシュの理想的な辺長を表している．通常，sizing function はメッシュの頂点のみに対して値が与えられ，他の点では各三角形上で補間を行って値を求める．Sizing function の値はメッシュの幾何的性質や数値計算の誤差評価などによって決定される．

Sizing function が定数値関数の場合は，一様メッシュ（等方的で，全ての三角形の大きさがほぼ等しいメッシュ）が作成され，曲率に依存した関数の場合は曲率適応的メッシュが作成される．また sizing function は edge collapse, edge split, edge swap とともに，メッシュの質の最適化にも用いられている [7]．実際の辺の長さが理想長（sizing function の値）よりも短ければ edge collapse を，長ければ edge split を適用して，sizing function との整合度を高くし，edge swap を用いて等方性を高くしている．

また注意すべき点として，sizing function の変化が大きい場合，適応度の高いメッシュが作成されるのだがそれと同時に三角形の等方性が低下してしまうことが挙げられる．これを防ぐには，sizing function の変化を抑える必要がある [1]．

2.3　提案法

提案法の説明をする前に，辺長をコスト関数とした edge collapse 簡略化法について述べる（以下，辺長簡略化法と略す）．この簡略化法はコストの計算に最適化が必要ないので，高速である．さらに，この単純な方法は比較的質の良い一様メッシュを作成できることが確認されている．これは頂点の密度を均一化するように edge collapse が働くからである [15]．我々はこの方法を，sizing function と組み合わせることにより，適応的メッシュを作成できるように拡張した．

本研究では，頂点数が指定された数と一致するように簡略化を行うので，一般に，簡略化メッシュが sizing function と整合する（実際の辺長と sizing function がほぼ同じ値を取る）ように簡略化を行うのは無理である．そこで，このような状況で sizing function と整合性に意味を持たせるために，これらを以下のように定義し直す．

まず，sizing function を辺の絶対長を表すものではなく，相対長を表すものだと見なし，それを relative sizing function (rsf) と呼ぶことにする．rsf はメッシュ上の点に対して定義されるものだが，辺に対しても定義することができる．辺 e 上における rsf の平均値を $\mathrm{rsf}(e)$ と定めるのである．

そして，次のような条件を満たすときメッシュと rsf は整合性を持つ、と定義する：メッシュの全ての辺 e_i に対して，$d(e_i) \approx \alpha\,\mathrm{rsf}(e_i)$ を満たすような定数 α が存在する（ここで d は実際の辺長を表す）．また $\delta(e) = d(e)/\mathrm{rsf}(e)$ とすると，この条件は次のように書き換えられる：メッシュの全ての辺 e_i に対して，$\delta(e_i)$ はほぼ同じ値を取る．

つまり，δ が全ての辺に対して等しくなるように簡略化メッシュを作成すれば，rsf に整合したメッシュが得られることになる．δ は，実際の辺長が理想長と比較してどの程度“短い”かを表しており，提案法ではこの δ をコスト関数に用いた．辺長簡略化法では，コスト関数に d を用いて，d が均一化されるようにし，提案法では，コスト関数に δ を用いて，δ が均一化されるようにしている．また，これは control space [1] において，辺長簡略化法を適用することに相当する．

Edge collapse でつくられる新しい頂点の位置は次のように定める．まず高密度メッシュの各頂点に，次節で説明される式を用いて，密度関数を割り当てる（この関数は rsf から計算される）．そして，各 dual face に対して重さ（密度関数を dual face 上で面積分したもの）を計算し，それを各頂点の重さとする．そして，新しい頂点の位置を，縮約される辺の二つの頂点の重心に定める．重心の計算には先ほど求めた重さを用いる．新しい頂点の重さは，二つの頂点の重さの和を割り当てる．Edge collapse を利用して簡略化されたメッシュの頂点は，高密度メッシュ上においてその頂点に対応する dual face のクラスタがある [11]．上のように計算された頂点の位置は，その頂点に対応するクラスタの重心と一致している．

図 1 から見て取れるように，提案法は比較的質の高い簡略化メッシュを作成することができる．

プログレッシブメッシュ　簡略化を行うときにプログレッシブメッシュ[9] を作成すれば，任意の頂点数を持つ簡略化メッシュが，それから抽出できるようになる．つまり，一つのモデル・一つの sizing function に対して，簡略化は一度だけ行えば十分である．

特徴の保存　Edge collapse は単純で局所的な操作であるため，特徴の保存を容易に行える．特徴を含む辺および特徴と隣接する辺に簡単な制限を設けるだけでよい．

トポロジー　入力メッシュが多様体の場合はトポロジーを保ったまま簡略化を行うことが可能だが，edge collapse は非多様体メッシュの簡略化を行うこともできる．さらに，連結していないメッシュを融合させたい場合，およびメッシュの接続関係が与えられていない場合（ポリゴンスープ）でも，virtual edge [8] を利用すれば簡略化を行うことができる．

3　制約付き重心ボロノイ図

制約付き重心ボロノイ図を利用すれば，前節の方法で得られた簡略化メッシュから，高品質な簡略化メッシュを作成することができる．この方法は，ただの重心ボロノイ図ではなく制約付き重心ボロノイ図を利用することにより，Valette and Chassery の提案した方法 [16] を特徴の保存も行えるように拡張したものである．

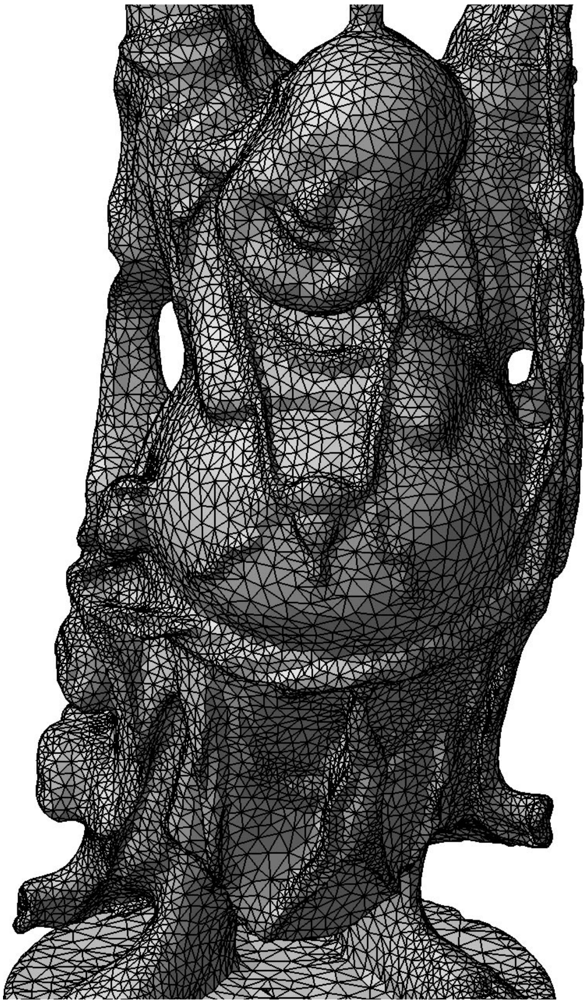

図 1: Buddha モデル（543,652 頂点）に edge collapse 簡略化法を適用して得られた曲率適応的メッシュ（25,038 頂点）．

本節では，はじめに重心ボロノイ図および制約付き重心ボロノイ図についての説明を行う．それから，効率的に制約付き重心ボロノイ図を作成することができる提案法について述べる．

3.1　重心ボロノイ図

重心ボロノイ図とは，各ボロノイ領域の母点と重心が一致するという特殊な母点配置を持ったボロノイ図のことである [3]．重心ボロノイ図の各ボロノイ領域は，正六角形に近い形をしており，その双対図形であるドロネー三角形分割の各三角形は，正三角形に近い形をしている．すなわち，重心ボロノイ図は等方的メッシュの作成に用いることができるのである．メッシュを作成したい領域上に重心ボロノイ図を作成し，その双対を取ることにより等方的メッシュが得られる．重心ボロノイ図は次に述べる Lloyd 法と呼ばれる反復法を用いて計算されることが多い．

Lloyd 法 [13] は，まず重心ボロノイ図を作成したい領域に母点をランダムに配置する．そして，収束条件を満たすまで次の二つの操作を繰り返す：

1. 与えられた母点のボロノイ図を作成する．
2. 母点をそれぞれのボロノイ領域の重心に更新する．

この操作がメッシュの作成にどのような効果を与えているかを考えてみよう．つまり，これらの操作がドロネー三角形分割にどのような影響を与えているかを考えてみよう．母点はドロネー三角形分割における頂点，ボロノイ領域同士の接続関係はドロネー三角形分割の辺に対応していることを考えると，ステップ 1 は接続関係の最適化に対応しており，ステップ 2 は頂点位置の最適化に対応していることが分かる．この二つの操作を繰り返し適用することにより，メッシュは等方的になっていく．

3.2　制約付き重心ボロノイ図

曲面上のメッシュ生成，特徴の保存などには重心ボロノイ図よりも制約付き重心ボロノイ図の方が適している．制約付き重心ボロノイ図では，(一部もしくは全ての）重心が曲面上や特徴上に制約される [4, 5]．制約されている重心を制約付き重心と呼

ぶ．この制約によって曲面上のメッシュ生成や特徴の保存が可能になっている．

測地距離などを用いれば曲面上に重心ボロノイ図を作成してメッシュ生成を行うこともできるが，計算コストが大きいという欠点を持つ．一方，Du らが提案した制約付き重心ボロノイ図 [5] は距離をユークリッド距離で計算しているので，コストは高くない．

制約集合 S に制約されている制約付き重心は，制約無し重心から S への最近点であることに注意してほしい．この最近点を求める操作をプロジェクションと呼ぶ．制約付き重心ボロノイ図も Lloyd 法を用いて計算できるが，ステップ 2 が，制約無し重心の計算してさらにプロジェクションを行って制約付き重心を求める，という二段階構成になる．

3.3 提案法

我々は [16] と同じく，離散近似を行って効率的にボロノイ図を計算をした．離散近似を行った理由は，各ボロノイ領域を連結に保つことが容易で，さらに計算が高速・ロバストに行えるからである．この方法は，入力メッシュが高密度で出力メッシュの要素数が十分に小さいときに効果的で，それ以外の場合は離散化誤差が大きくなるので，あまり質の良くない簡略化メッシュが得られるようになる．離散近似ボロノイ図を作成することをクラスタリング，離散近似ボロノイ領域のことをクラスタと呼ぶこともある．

提案法も Lloyd 法を用いて制約付き重心ボロノイ図の計算を行った．母点の初期配置にはランダムサンプリングを利用するのではなく，前節で得られた簡略化メッシュの頂点を利用する．ただ，この簡略化メッシュの頂点は高密度メッシュ上に乗っていないため，まずプロジェクションを行う必要がある．それから Lloyd 法の反復に入る．ここでは次のような操作を行う：

1. 高密度メッシュ上でボロノイ図を計算する（クラスタリングを用いる）．
2. ボロノイ領域の境界の調整を行って離散化誤差を減らす．
3. 制約無し重心を計算し，それからプロジェクションを行い母点を更新する．

クラスタリング　特徴の保存を行うために，高密度メッシュの dual face をセルと定め離散近似ボロノイ図を作成した（dual face クラスタリング）．三角形をクラスタリングして離散近似ボロノイ図を作成することもできるが（face クラスタリング），この方法だと特徴を保存する方法が複雑になってしまう．Klein らはメッシュ上の測地ボロノイ図を dual face クラスタリングを用いて計算している [12]．

クラスタリングには次のように region growing アルゴリズムを用いた [14, 2]：

1. 各母点に対し，母点に最も近い dual face を seed dual face と定める．
2. Seed dual face を含むクラスタとそれに隣接する dual face のうち最も距離の小さい組をクラスタリングする．
3. 全ての dual face がクラスタリングされるまでステップ 2 を繰り返す．

このアルゴリズムは，制限を設けた（母点と接続している辺しか縮約できない）half edge collapse 簡略化法と同等であり，特徴の保存（位相的データ）は容易に行える．

境界の調整　Dual face クラスタリングによる離散化誤差を少なくするために，高密度メッシュの各三角形内でクラスタ間の境界を実際のボロノイ図の境界に整列させる．

プロジェクション　特徴の保存（幾何的データ）を行うために，特徴を含むクラスタの母点を特徴上に制約する．それ以外の母点は高密度メッシュ上に制約する．重心を計算するときに密度関数 $\rho(\boldsymbol{x})$ が必要になるが，それは次の式を用いて計算する [6]：$\rho(\boldsymbol{x}) = \mathrm{rsf}(\boldsymbol{x})^{-4}$．

4 結果

我々はシャープエッジを持った機械部品のモデルや 1 千万個以上ものポリゴンを持ったモデルに対して実験を行った．実験結果によれば，Lloyd 法での反復は 10 回程度で行えば，高品質な簡略化メッシュが得られることが分かった．そして，提案法は特徴の保存を行うことも可能であり（図 2），任意

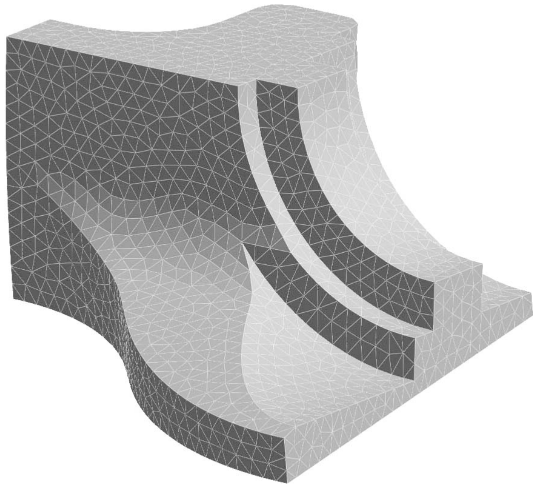

図 2: 特徴を保存した一様な簡略化メッシュ.

のトポロジーを持つメッシュの簡略化を行うことができる.

5 まとめ

本研究では，edge collapse と sizing function を組み合わせることにより，比較的質の良い簡略化メッシュを高速に作成する方法を提案した．またそのメッシュを元にして，制約付き重心ボロノイ図を利用することにより，高品質なメッシュを効率的に作成する方法も提案した．そして，いくつかの実験を通して提案法の有効性を確認した．

今後の課題として，等方的なメッシュだけはでなく非等方的なメッシュを作成できるようにすること，提案法を簡略化の枠組みからリメッシングの枠組みへと拡張することが挙げられる．

謝辞

本研究は，文部科学省 21 世紀 COE プログラム「情報科学技術戦略コア」，及び，科学研究費補助金基盤研究 (S)15100001 の援助を受けている．

参考文献

[1] Borouchaki, H., Hecht, F., and Frey, P. J. Mesh gradation control. *Int. J. Numer. Methods Engng*, **43** (6): 1143–1165, 1998.

[2] Cohen-Steiner, D., Alliez, P., and Desbrun, M. Variational shape approximation. In *Proceedings of SIGGRAPH*, 905–914, 2004.

[3] Du, Q., Faber, V., and Gunzburger, M. Centroidal Voronoi tessellations: applications and algorithms. *SIAM Review*, **41** (4): 637–676, 1999.

[4] Du, Q. and Gunzburger, M. Grid generation and optimization based on centroidal Voronoi tessellations. *Applied Mathematics and Computation*, **133** (2–3): 591–607, 2002.

[5] Du, Q., Gunzburger, M. D., and Ju, L. Constrained centroidal Voronoi tessellations for surfaces. *SIAM Journal on Scientific Computing*, **24** (5): 1488–1506, 2003.

[6] Du, Q. and Wang, D. Tetrahedral mesh generation and optimization based on centroidal Voronoi tessellations. *International Journal for Numerical Methods in Engineering*, **56** (9): 1355–1373, 2003.

[7] Frey, P. J. and Borouchaki, H. Geometric surface mesh optimization. *Computing and Visualization in Science*, **1** (3): 113–121, 1998.

[8] Garland, M. and Heckbert, P. S. Surface simplification using quadric error metrics. In *Proceedings of SIGGRAPH*, 209–216, 1997.

[9] Hoppe, H. Progressive meshes. In *Proceedings of SIGGRAPH*, 99–108, 1996.

[10] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., and Stuetzle, W. Mesh optimization. In *Proceedings of SIGGRAPH*, 19–26, 1993.

[11] Kim, J. and Lee, S. Transitive mesh space of a progressive mesh. *IEEE Transactions on Visualization and Computer Graphics*, **9** (4): 463–480, 2003.

[12] Klein, A., Certain, A., DeRose, A., Duchamp, T., and Stuetzle, W. Vertex-based Delaunay triangulation of meshes of arbitrary topological type. Technical Report, University of Washington, 1997.

[13] Lloyd, S. P. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, **28** (2): 129–137, 1982.

[14] Sander, P. V., Wood, Z. J., Gortler, S. J., Snyder, J., and Hoppe, H. Multi-chart geometry images. In *Proceedings of SIGGRAPH*, 146–155, 2003.

[15] Southern, R., Marais, P., and Blake, E. Generic memoryless polygonal simplification. In *Proceedings of AFRIGRAPH*, 7–15, 2001.

[16] Valette, S. and Chassery, J.-M. Approximated centroidal Voronoi diagrams for uniform polygonal mesh coarsening. In *Proceedings of Eurographics*, 381–389, 2004.