A Survey and Evaluation of Mesh Reduction Techniques

Burkhard Wünsche

Department of Computer Science, The University of Auckland
Private Bag 92019, Auckland, New Zealand
Email: `burkhard@cs.auckland.ac.nz`

Abstract: Large polygon meshes are a common entity in scientific and engineering science. Polygon meshes can be used for simplified geometric operations such as collision detection and surface analysis and permit hardware assisted rendering, an essential condition to achieve real-time display. In practice, meshes obtained by conventional polygonization techniques are often too big for efficient rendering and postprocessing. A mesh reduction technique is then employed to reduce the mesh size. This paper gives an overview of common mesh reduction techniques, classifies them according to their design methodology and compares and evaluates them.

Keywords: Mesh simplification, multiresolution meshes

1 Introduction

Large meshes are common in computer graphics, for example when using devices such as CT, MRI, range cameras, or satellite data. Since large meshes put a strain on storage capacity, communication, and rendering hardware a mesh reduction algorithm must often be applied. The chosen technique depends on the application. Whereas some techniques aim only to eliminate small and badly shaped polygons, other techniques try to achieve a maximum reduction of polygons. Often the user wants to get a fast preview of the mesh with the possibility to increase the resolution of the whole mesh (*multiresolution surface meshing*) or part of the mesh (*local level-of-detail control*). The latter technique is common in digital terrain modeling where far away objects can be approximated less accurately to improve rendering speed[1]. A survey of polygonization methods and optimization techniques is given by Wünsche [27]. An introduction into mesh reduction techniques is given by Schroeder [22] and a short overview and classification is contained in [19].

2 Simplification Methodologies

We identify six methodologies used for mesh reduction:

I. Polygon merging combines coplanar or nearly coplanar polygons into bigger ones. The original topology of the mesh is not changed.

The merging criteria are often based on the surface gradient (e.g., as given by the polygon normals) [11, 15] or can be perceptually driven as done by Reddy [19] who uses the human *Contrast Sensitivity Function* which considers both size of a polygon and surface curvature.

After merging the polygons, collinear or nearly collinear edges are eliminated [11] or polygon boundaries are approximated by straight lines with respect to a distance measures [15]. The such simplified polygons are then retriangulated.

II. Polygon elimination deletes polygons by collapsing their vertices to single points.

The methods using this technique vary mostly only in the criteria used for polygon elimination. The quality of a mesh can be improved by eliminating very small or very thin polygons [18] whereas a

[1] For a good overview of algorithms for the simplification of polyhedral terrains see [10].

larger polygon reduction is usually achieved by eliminating polygons according to surface curvature [9]. Gieng et al. [7] use triangle-collapse operations. For each triangle they generate a bivariate polynomial approximating the underlying surface in the area of the triangle. This function serves as a basis for finding an optimal point to which the triangles are collapsed. The elimination criterion for a triangle depends on its size and shape and the curvature of the generated polynomial.

III. Vertex elimination deletes vertices and retriangulates the resulting holes.

Typical criteria used for vertex elimination are planarity or curvature criteria [24] or global minimum distance errors to the initial mesh or, if known, the original surface [5].

Schroeder [23] improves the quality of the simplified mesh by preserving high-frequency information such as sharp edges and corners. The decimation algorithm visits each vertex and classifies the local geometry and topology in the neighbourhood of the vertex. A vertex is deleted if it fulfills a local planarity measure given by the distance to a plane function for interior vertices and a distance to line function for boundary vertices.

Renze and Oliver [20] present a robust triangulation method to fill the holes resulting from vertex elimination. Ciampalini et al. [3] minimizes the approximation error of a triangulation by a series of edge flipping operations. They consider for this the approximated volumetric error, the sampled local error, and the triangle aspect ratio and area difference. The authors mention that by using edge flipping they obtain a better approximation with nearly one-half of the triangles used without flipping.

IV. Edge collapsing replaces edges by vertices, obviating the need for retriangulation.

An edge can be replaced either by its center [1] or by moving one endpoint onto another [28, 21]. The first solution sometimes yields a better approximation whereas the latter solution retains the original vertex set which proves useful in the definition of efficient multiresolution meshes (see section 3). Cohen et al. [4] choose the new vertex in a two dimensional projection of the local mesh in order to avoid self-intersection and optimize in the third dimension to minimize the error.

Hoppe et al. [14] use as a criteria for edge collapsing an energy function over a mesh to minimize both the distance of the approximating mesh from the original, as well as the number of approximating vertices. In [12] he extends his method in order to not only preserve the geometry of the original mesh but also the overall appearance defined by its attributes such as colour, normals, and texture coordinates. Hoppe achieves this goal by using an energy metric such that an edge collapse for an edge with different attributes has a low priority.

For view-dependent refinement it is possible to develop criteria depending on viewing direction, lighting, and the screen-space projection area [28] or view frustum, surface orientation, and screen-space geometric error [13].

Cohen et al. [4] develop a piece-wise linear mapping function for each simplification operation which is used to compute tight error bounds for the simplified mesh. The mapping function can also be used to compute appropriate texture coordinates, colour, or surface normals for the simplified mesh and is used for geomorphing and the computation of tight error bounds on colour and overall appearance.

V. Retiling replaces the entire mesh with a coarser one using some error criteria. The original polygonization is used only to provide the error measure.

An example is given by Turk [26] who distributes a set of points on a mesh by point repulsion, with density weighted by estimates of local curvature and uses these points for a new polygonization.

VI. Multiresolution Wavelet Analysis (MRA) is used to decompose a simple function into a low resolution part and so-called wavelet coefficients necessary to recover the original function. To apply multiresolution analysis to mesh reduction the mesh is expressed as a parametric function. The low resolution part of the function gives then a reduced mesh in which the new vertices are weighted averages of the original vertices [17, 6].

View dependent refinement can be achieved by filtering the wavelet space directly [8] whereas reduction of other mesh attributes such as colour can be achieved by combining a different multiresolution presentation for them with the representation for the geometry at display time [2].

3 Multiresolution Meshes

In many applications the optimal mesh simplification level is known only at run-time. For example, in Virtual Reality applications parts of the scene close to a moving viewer or being highlighted must be displayed in a higher resolution than parts far away or lying in a shadow. A *multiresolution representation* provides a solution. It is a data structure that allows a compact representation of different levels-of-detail (LOD) with a size of the same order as the data size [3].

Multiresolution meshes are becoming increasingly popular since they usually also offer progressive storage, transmission, and reconstruction (starting with a low resolution base mesh). Two types of multiresolution meshes are described in the literature:

Multiresolution wavelet analysis (MRA), already introduced in the previous section, stores a mesh as a base mesh and a series of wavelet coefficients. *Progressive Meshes* [14, 12, 23, 28] are a series of triangle meshes connected by reversible simplification operations. Usually the edge collapse operation is chosen, which according to Schroeder [23] is fast and leads to a representation smaller than the original mesh representation. A similar scheme using triangle collapse operations with an inverse vertex split was developed by Gieng et al. [7].

Hoppe [12] and Ciampalini et al. [3] mention that in contrast to MRA their approaches have lossless representation, manage feature edges or discontinuities with higher precision and are more efficient in construction and reconstruction. Multiresolution analysis also has the disadvantage that it requires remeshing and resampling [3]. Furthermore the wavelet approach can be somewhat conservative and for a given error bound, algorithms based on vertex removal and edge collapse [5, 12] have been found to provide more simplification (in terms of reducing polygon count). Hoppe [12] lists as advantages of MRA that it allows approximations with a guaranteed global error [6] and the independent compression of geometry and attributes [2].

4 Comparison

We have introduced a variety of mesh reduction techniques and provided some comparisons of decimation methodologies and decimation criteria. In order to convey better understanding of the methods we now present an extended and updated version of a classification framework previously introduced by us in [27]. We then summarize some comparisons of speed and achieved mesh quality published in the existing literature.

Our criteria for a classification are:

1. Does the algorithm achieve the approximation with a given number of vertices or polygons (*bounded number approximation*)?
2. Does the algorithm achieve the approximation with a given maximum error of ϵ (*bounded ϵ approximation*)?
3. Does the algorithm provide *multiresolution surface meshing* to obtain meshes of different resolution?
4. Does the algorithm provide a *geomorph* to interpolate smoothly between different levels-of-detail?
5. Does the algorithm use a subset of the original mesh vertices (vs. resampling)?
6. Does the algorithm preserve the mesh topology? Methods which allow a controlled change of the topology are marked with an asterisk in table 1.
7. Does the algorithm allow a *local level-of-detail control*? This is useful if only selected parts of a model must be approximated in detail.
8. Does the algorithm work for arbitrary meshes (i.e., for triangular 2-manifold meshes)?

9. Which error measure/removal criteria is used: 1. distance measure 2. polygon size 3. surface curvature (estimated by normal angle) 4. wavelet coefficient .

Note that preservation of topology is important in applications such as FEM analysis, medical imaging, or molecular modelling but in many other applications such as Virtual Reality it presents an undesired constrain impacting on interactivity and speed. An example for a desired change in topology would be a cubical plate with many small holes which, if topological change is allowed, can be approximated by just six polygons. The results of classifying the presented algorithms with these criteria are shown in table 1.

		Criteria								
		1.	2.	3.	4.	5.	6.	7.	8.	9.
I.	[11]	No	No	No	No	Yes	Yes	No	Yes	3
	[15]	Yes	No	No	No	Yes	Yes	No	Yes	1+3
II.	[9]	Yes	Yes	No	No	No	Yes	No	Yes	3
	[7]	Yes	No	Yes	Yes	No	Yes	No	Yes	2+3
III.	[24]	Yes	No	No	No	Yes	Yes	No	Yes	1
	[5]	No	Yes	No	No	Yes	Yes	Yes	Yes	1
	[23]	Yes	Yes	Yes	No	Yes	No*	No	Yes	1
	[3]	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	1
V.	[14]	Yes	No	No	No	No	Yes	No	Yes	1
	[12]	Yes	No	Yes	Yes	No	No*	Yes	Yes	1
	[28]	Yes	No	Yes	No	Yes	No	Yes	Yes	5
	[21]	Yes	Yes	Yes	No	Yes	No*	No	Yes	1
	[1]	No	No	No	No	No	Yes	No	Yes	3
	[4]	Yes	Yes	No	Yes	No	Yes	Yes	Yes	1
VI	[25]	Yes	No	No	Yes	No	Yes	No	Yes	-
VI.	[17]	Yes	Yes	Yes	Yes	No	Yes	No	No	4
	[6]	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	4
	[8]	Yes	Yes	Yes	No	Yes	Yes	Yes	No	4
	[2]	Yes	Yes	Yes	Yes	No	Yes	No	Yes	4

Table 1: Comparison of mesh reduction techniques.

Currently the *Mesh Optimization* algorithm from Hoppe et al. [14] and the *Mesh Decimation* method from Schroeder et al. [24] appear the most popular and have been the basis for numerous new algorithms [12, 13, 4, 23, 3].

Several authors [22, 15, 3] suggest that *Retiling* [26] is well suited for curved, round objects and that *Mesh Optimization* [14] appears to give best results, but both algorithms are relatively slow. Mesh Decimation [24] and *Geometric Optimization* [11] appear to be fastest. Schroeder [23] states the time complexity of Mesh Decimation as $O(n)$.

Ciampalini et al. [3] report that their algorithm is about 20–30 times faster than Mesh Optimization, but about 10 times slower than Mesh Decimation. Given a target number of vertices their global error is 4 – 10 times lower than with Mesh Decimation and only slightly worse than Mesh Optimization.

Finally we note that most methods achieve simplification by a local error criteria such that the global error for a simplification step is only given as an incremental and hence rather inexact error. Efficient global error control is achieved by [5, 16, 23, 4].

5 Conclusion

Polygon meshes allow conventional storage of implicit surfaces, permit hardware assisted rendering, and are used for numerous simplified geometric operations and other applications.

In practice meshes obtained by conventional polygonization techniques are often too big for efficient rendering and postprocessing. A mesh reduction technique is then employed to reduce the mesh size. We have introduced a variety of mesh reduction techniques, have classified them into six groups and have given an evaluation and comparison between them. The field of multiresolution meshes deserves special attention and we plan to publish a more detailed review and evaluation shortly.

We hope that our work forms a good basis for the decision making about which method to implement and for the development of new mesh reduction techniques.

References

[1] María-Elena Algorri and Francis Schmitt. Mesh simplification. In Jarek Rossignac and François Sillion, editors, *Computer Graphics Forum (Eurographics '96)*, volume 15, pages C77 – C86. Eurographics Association, University Press, Cambridge, UK, 1996. Futoroscope - Poitiers, France August 26 – 30, 1996.

[2] Andrew Certain, Jovan Popović, Tony DeRose, Tom Duchamp, David Salesin, and Werner Stuetzle. Interactive multiresolution surface viewing. In Holly Rushmeier, editor, *SIGGRAPH '96 Conference Proceedings*, Annual Conference Series, pages 91–98. ACM SIGGRAPH, Addison-Wesley Publication Company Inc, August 1996. Held in New Orleans, Louisiana, 04-09 August 1996.

[3] A. Ciampalini, P. Cignoni, C. Montani, and R. Scopigno. Multiresolution decimation based on global error. *The Visual Computer*, 13(5):228 – 246, May 1997.

[4] Jonathan Cohen, Dinesh Manocha, and Marc Olano. Simplifying polygonal models using successive mappings. In Roni Yagel and Hans Hagen, editors, *Proceedings of IEEE Visualization '97*, pages 395 – 402, 1997.

[5] Jonathan Cohen, Amitabh Varshney, Dinesh Manocha, Greg Turk, Hans Weber, Pankaj Agarwal, Frederick P. Brooks, Jr., and William Wright. Simplification envelopes. In Holly Rushmeier, editor, *SIGGRAPH '96 Conference Proceedings*, Annual Conference Series, pages 119–128. ACM SIGGRAPH, Addison Wesley, August 1996. Held in New Orleans, Louisiana, 04-09 August 1996, URL: `http://www.cs.unc.edu/~geom/envelope.html`.

[6] Matthias Eck, Tony DeRose, Tom Duchamp, Hugues Hoppe, Michael Lounsbery, and Werner Stuetzle. Multiresolution analysis of arbitrary meshes. In Robert Cook, editor, *SIGGRAPH '95 Conference Proceedings*, Annual Conference Series, pages 173–182. ACM SIGGRAPH, Addison Wesley, August 1995. Held in Los Angeles, California, 06-11 August 1995, URL: `ftp://ftp.cs.washington.edu/tr/1995/01/UW-CSE-95-01-02.d`.

[7] Tran S. Gieng, Bernd Hamann Kenneth I. Joy, Gregory L Schussman, and Issac J. Trotts. Smooth hierarchical surface triangulations. In Roni Yagel and Hans Hagen, editors, *Proceedings of IEEE Visualization '97*, pages 379 – 386, 1997.

[8] Markus H. Gross, Oliver G. Staadt, and Roger Gatti. Efficient triangular surface approximations using wavelets and quadtree data structures. *IEEE Transactions on Visualization and Computer Graphics*, 2(2):130 – 143, June 1996.

[9] Bernd Hamann. A data reduction scheme for triangulated surfaces. *Computer-Aided Geometric Design*, 11(2):197 – 214, 1994.

[10] P. S. Heckbert and M. Garland. Fast polygonal approximation of terrains and height fields. Technical report cmu-cs-95-181, School of Comp. Sci., Carnegie Mellon University, Pittsburgh, PA 15213, 1995. URL: `http://www.cs.cmu.edu/afs/cs.cmu.edu/user/garland/www/scape/scape.ps.gz`.

[11] P. Hinker and C. Hansen. Geometric optimization. In G. M. Nielson and D. Bergeron, editors, *Proceedings of Visualization '93*, pages 189 – 195, Los Alamitos, California, 1993. IEEE, Computer Society Press.

[12] Hugues Hoppe. Progressive meshes. In Holly Rushmeier, editor, *SIGGRAPH '96 Conference Proceedings*, Annual Conference Series, pages 99–108. ACM SIGGRAPH, Addison-Wesley Publication Company Inc, August 1996. Held in New Orleans, Louisiana, 04-09 August 1996, URL: `http://www.research.microsoft.com/research/graphics/hoppe/pm_sig96_bw.ps.gz`.

[13] Hugues Hoppe. View-dependent refinement of progressive meshes. In *SIGGRAPH '97 Conference Proceedings*, Annual Conference Series, pages 189–198. ACM SIGGRAPH, Addison-Wesley Publication Company Inc, August 1997. Held in Los Angeles, California, August 03-08, 1997, URL: `http://www.research.microsoft.com/research/graphics/hoppe`.

[14] Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzel. Mesh optimization. In *Proceedings SIGGRAPH '93*, pages 19 – 26. ACM SIGGRAPH, 1993. URL: `ftp://ftp.cs.washington.edu/tr/1993/01/UW-CSE-93-01-01.PS.Z`.

[15] Alan D. Kalvin and Russell H. Taylor. Superfaces: Polygonal mesh simplification with bounded error. *IEEE Computer Graphics and Applications*, 16(3):64–77, May 1996. ISSN 0272-1716.

[16] Reinhard Klein, Gunther Liebich, and Wolfgang Straßer. Mesh reduction with error control. In Roni Yagel and Gregory M. Nielson, editors, *IEEE Visualization '96*. IEEE, October 1996.

[17] Michael Lounsbery. *Multiresolution Analysis for Surfaces of Arbitrary Topological Type*. PhD thesis, Department of Computer Science and Engineering, University of Washington, September 1994. URL: `ftp://cs.washington.edu/pub/graphics/LounsPhd.ps.Z`.

[18] D. Moore and J. Warren. Compact isocontours from sampled data. In D. Kirk, editor, *Graphic Gems III*, pages 23 – 28. Academic Press, 1992.

[19] M. Reddy. SCROOGE: Perceptually-driven polygon reduction. *Computer Graphics Forum*, 15(4):191 – 203, October 1996.

[20] Kevin J. Renze and James H. Oliver. Generalized unstructured decimation. *IEEE Computer Graphics and Applications*, 16(6):24 – 32, November 1996.

[21] Rémi Ronfard and Jarek Rossignac. Full-range approximation of triangulated polyhedra. *Computer Graphics Forum (Proceedings of Eurographics '96)*, 15(3):C67 – C76, 1996. Futoroscope - Poitiers, France August 26 – 30, 1996.

[22] William J. Schroeder. Polygon reduction techniques. Course notes # 30, ACM SIGGRAPH, 1995.

[23] William J. Schroeder. A topology modifying progressive decimation algorithm. In Roni Yagel and Hans Hagen, editors, *Proceedings of IEEE Visualization '97*, pages 205 – 212, 1997.

[24] William J. Schroeder, Jonathan A. Zarge, and William E. Lorensen. Decimation of triangle meshes. In Edwin E. Catmull, editor, *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 65 – 70, July 1992.

[25] Greg Turk. Generating textures on arbitrary surfaces using reaction-diffusion. *Computer Graphics*, 25(4):289 – 298, 1991. (SIGGRAPH '91).

[26] Greg Turk. Re-tiling polygonal surfaces. In Edwin E. Catmull, editor, *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 55–64, July 1992.

[27] Burkhard C. Wünsche. A survey and analysis of common polygonization methods & optimization techniques. *Machine Graphics & Vision*, 6(4):451 – 486, 1997.

[28] Julie C. Xia, Jihad El-Sana, and Amitabh Varshney. Adaptive real-time level-of-detail-based rendering for polygonal models. *IEEE Transactions on Visualization and Computer Graphics*, 3(2):171 – 183, April 1997.