CSE 528 Computer Graphics
Instructor: Hong Qin

Hima Bindhu Busireddy **Project Report** **110747994**

Topic: Quadrilateral Mesh Simplification

3D models are represented as meshes needs to re meshed or simplified as they help in Level of detail (LOD) rendering.

In Computer graphics Level of detail plays a vital role as it involves complexity reduction for 3D models. Though LOD seems to be applicable to geometry detail only, but it can be generalized. These LOD techniques includes shader management and applicable to texture maps.

Based on the shape of the elements constituting meshes, Meshes can be of different kinds like Triangular, Quad-Dominant and Quadrilateral meshes. This project aims at simplifying quadrilateral mesh structures.

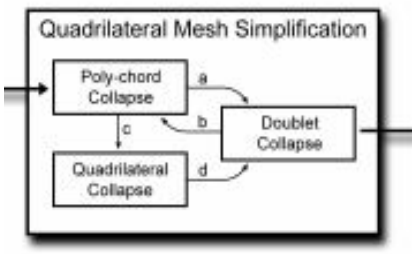
Main Idea:

The purpose of this project is to implement a quadrilateral mesh simplification (QMS) technique presented in paper https://vgc.poly.edu/~csilva/papers/siggraph-asia2008.pdf QMS simplification algorithm is a mesh structure independent and uses dual of a mesh in a controlled manner. The set of operations like poly-chord and quad-based collapse are applied on dual meshes in a balanced way, thus helping to generate quality meshes.

QMS Algorithm:

The algorithms starts with dual of mesh construction using any method.

Once dual is obtained then QMS algorithm starts with poly-chord collapse method in iterative way choosing the best deletion method possible. Iteration is shown as (looping between state transitions a and b). When the poly-chord collapses are done i.e. iteration is terminated then algorithm chooses quad-based collapse for element deletion shown as (state transition c). Consequently in the next step algorithm returns to poly-chord collapse (via state transition d and b), to continue processing on the new mesh structure. Doublets are removed after every collapse step (state transitions a and d)



CSE 528 Computer Graphics
Instructor: Hong Qin

Hima Bindhu Busireddy **Project Report** **110747994**

Implementation Details:

Implementation involved following steps as mentioned below:

- Loading and Rendering a mesh from ply file
- Dual mesh processing and rendering
- Polychord detection and collapse
- Quadrilateral collapse
- Doublet collapse
- Applying QMS algorithm and rendering final mesh

Loading and Rendering a mesh from ply file

The input mesh file supplied to the QMS system is a .ply file which has the structure as described below.

ply file consists of set of vertices and faces for any model and we also get the other extra information like total number of vertices and total number of faces. Each vertex is indicated with three coordinates x,y and z respectively. Each face is indicated with four set of vertices beginning with number '4' indicating that all the faces are quadrilaterals. An example input file is as shown in the below figure.

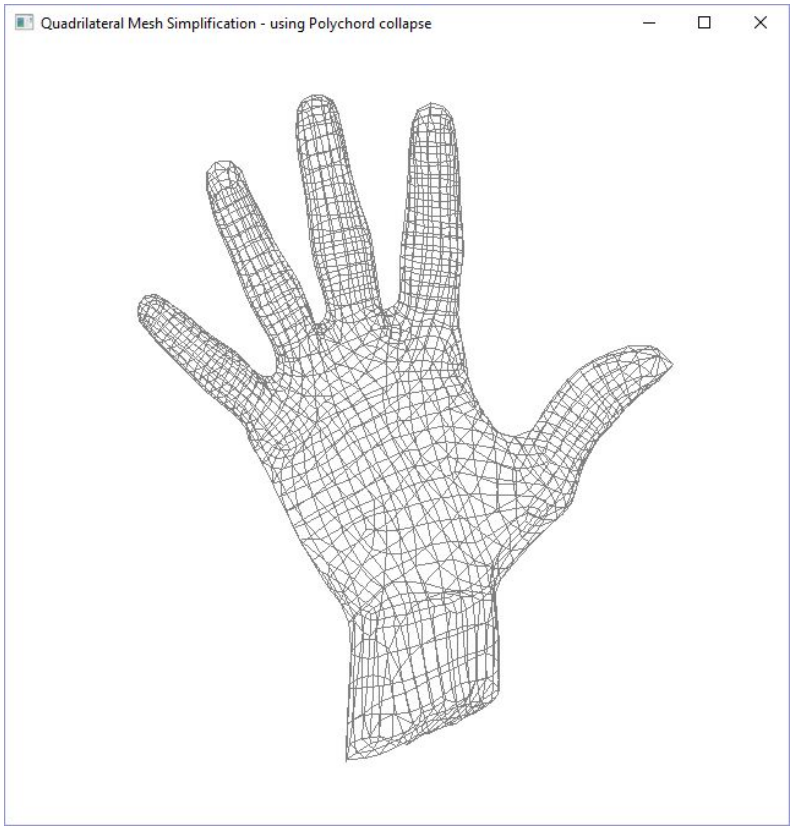
```
1  ply
2  format ascii 1.0
3  comment Licensing: This code is distributed under the GNU LGPL license.
4  element vertex 56
5  property float32 x
6  property float32 y
7  property float32 z
8  element face 48
9  property list uint8 int32 vertex_index
10 end_header
11 -2 0 0
12 -1.7 1.7 0
13 0 2 0
14 1.7 1.7 0
15 4 0 1 9 8
16 4 1 2 10 9
17 4 2 3 11 10
18 4 3 4 12 11
19 4 4 5 13 12
20 4 5 6 14 13
```

As part of the first step ply file is read as input from the user and processed to form edges and

polygons. Also the vertices and edges are stored for the input mesh.In order to render the input mesh the faces are traversed one by one iteratively and edges composing faces are drawn.

Hima Bindhu Busireddy **Project Report** **110747994**

Example output is shown in the below figure where a hand model mesh is rendered:



Dual mesh processing and rendering

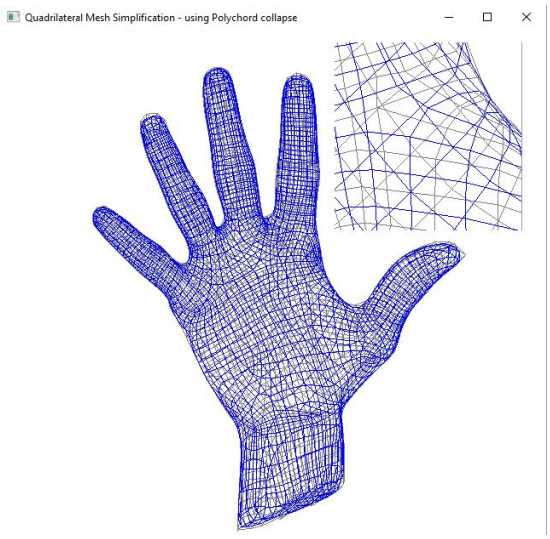
Dual for a given mesh is computed following the approach mentioned in paper and other papers referred by the author for the same.

- The dual of a quadrilateral element is its centroid
- The dual of a quadrilateral edge is the chord that connects the centroids of neighboring quadrilaterals
- The dual of a vertex is the polygon formed by connecting the centroids, in a cyclic order, of neighboring quad elements.

Though dual implementation is not part of the paper chosen for the project and since i could not find any dual models provided by author or available online, I had to code the logic computation by self.

Traversing through each face iteratively calculated the centroids and these are the duals for faces as per the dual theory. Then picked a random vertex and processed the surrounding faces from the face containing these centroid. The number of connected faces obtained in the above step is not constant since it varies with the connectivity of a picked quad. The project is about dealing with quad only meshes, Hence this number is fixed to a constant three and a dual face is formed with the four dual centroids chosen.

The dual mesh is then rendered in the similar way as mesh with computed dual vertices and faces.An example dual is shown in the below figure. Actual input mesh is in grey color and dula is in blue color as clearly seen in the zoomed part.



Polychord detection and collapse

This a very important and primary step for the QMS implementation where rings or polychords are detected in the dual mesh and are deleted. This way dual mesh after this global step will be loosing a good number of polygons with one step and will be optimized unlike other quad deletion operations where only one or closely connected edges are collapsed.

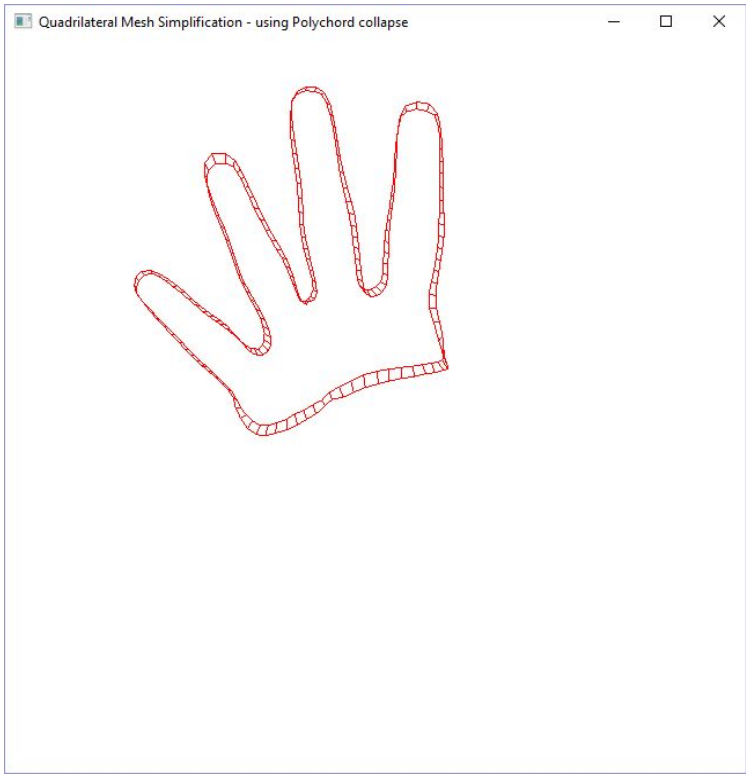
Polychord detection method: A random edge is picked and the opposite edge is computed for the respective polygon or quad face. Starting at the opposite edge the connected edges are traversed iteratively. If the traversal reaches the original edge then a ring is detected And the same is repeated till all the polygons in mesh are covered. If no loop is not detected then the mesh has not resulted in any polychords for the chosen random edge. In this case user will not see any polychords and message is displayed on the console saying to rerun for a different

edge trial. This approach is picked from the reference paper about polychord mentioned by authors.

Challenges: The paper has not presented any information about the decision on how many time polychord detection should be run as there is no information about number of polychords available. This challenge is addressed by taking a constant number of loop detections i.e 10.

Polychord edges thus detected are deleted from the mesh.

A polychord detection for a hand model is shown in the below figure:



Quadrilateral collapse

In this step quad is picked with opposing vertices is recognized and then collapsed from the dual mesh merging the connected elements.

Method: Polygon faces are traversed one by iteratively and for each face the edges are evaluated if each edge is connected to only one other polygon face. When this criteria is met for the chosen face then vertices sharing a diagonal in that face are deleted and the other vertices also identified connected edges are joined to the center point of the deleted vertices.

Hima Bindhu Busireddy **Project Report** **110747994**

Doublet collapse

In this step doublet is detected when two neighboring quadrilaterals that share two consecutive edges common edges are merged to single edge in the dual structure.

Method: Polygon faces are traversed one by iteratively and for each face the the neighbouring faces are evaluated if they have the same set of three vertices. All such doublets are stored and re-traversed where the two common edges formed by three vertices is collapsed to one edge with two vertices. A vertex which is in the middle of other two is chosen and removed from the dual resulting only one edge with rest two vertices.

Applying QMS algorithm and rendering final mesh

QMS Method: Given input mesh is loaded and dual is computed. polychord collapse method is then applied on the dual. This step is repeated until there are no more polychords in the resultant dual mesh. Then quadrilateral collapse is applied on the current dual mesh. As part of final step doublet collapse is applied on the dual mesh and the resultant mesh is rendered as final mesh after QMS.

User Interface:

This project presents an interactive user interface to the user with the following features

- User can press key ‘m’ to toggle of show/hide of mesh
- User can press key ‘d’ to toggle of show/hide of dual mesh
- User can press key ‘c’ to toggle of show/hide of poly chords
- User can press key ‘x’ to Zoom Out the view
- User can press key ‘z’ to Zoom In the view

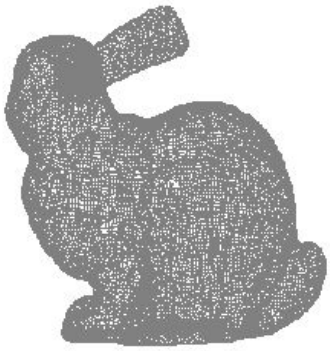
Testing:

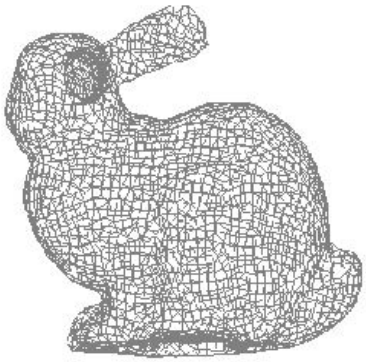
This project is tested against various quadrilateral mesh models like Hand, Bunny, Rampant, Moai and Tondo etc.For a example model say ‘bunny’ QMS is applied.

CSE 528 Computer Graphics
Instructor: Hong Qin

Hima Bindhu Busireddy **Project Report** **110747994**

Input bunny mesh file has 20k polygons and after QMS the resultant mesh model is composed of around 11k polygons. This output mesh is again fed as input to the QMS system and the resultant simplified mesh obtained is composed of 3k polygons as shown in the below figures respectively.





References:

http://www.opengl-tutorial.org/beginners-tutorials/tutorial-7-model-loading/

http://stackoverflow.com/questions/23298435/obj-loader-for-opengl-glut

https://www.researchgate.net/figure/230538831_fig10_Figure-11-Illustration-of-a-doublet-collapse-optimization-operation

http://www.cs.utah.edu/~jdaniels/research/papers/sgp2009_qCoarsen_present.pdf

http://dsk.ippt.pan.pl/docs/abaqus/v6.13/books/usi/default.htm?startat=pt06ch64s06s05.html

http://sci.utah.edu/~jdaniels/Joel_Daniels_Homepage/Publications.html

https://en.wikipedia.org/wiki/Dual_graph

CSE 528 Computer Graphics

Instructor: Hong Qin

Hima Bindhu Busireddy **Project Report** **110747994**