Generalized Adaptive Refinement for Grid-based Hexahedral Meshing

L. Pitzalis$^{1,2}$, M. Livesu$^3$, G. Cherchi$^1$, E. Gobbetti$^2$, R. Scateni$^1$

$^1$University of Cagliari, Italy
$^2$CRS4, Italy
$^3$CNR-IMATI, Italy
Generalized Adaptive Refinement for Grid-based Hexahedral Meshing

adaptive grid

conforming hexmesh
Grid-based Hexmeshing

start from an input surface

fit the input surface on its AABB
Grid-based Hexmeshing

refine the AABB according to a metric
Grid-based Hexmeshing

discard useless hexaedra and project the grid on the target surface
Grid-based Hexmeshing

A non-conforming mesh with hanging nodes is obtained.
Grid-based Hexmeshing

A non-conforming mesh with hanging nodes is obtained.

Goal: remove hanging nodes!
Remove hanging nodes
Remove hanging nodes
Remove hanging nodes

dualization is required!
Remove hanging nodes

mission complete: conforming hexmesh
Remove hanging nodes
Remove hanging nodes

**Balancing:**
the difference in the amount of refinement between adjacent cells cannot be greater than 1
Remove hanging nodes

**Balancing:**
the difference in the amount of refinement between adjacent cells cannot be greater than 1

**Pairing:**
hanging nodes must be taken in pairs to be removed
Remove hanging nodes

non conforming grid

conforming hexmesh
State of the art

[Gao et al. 2019]

[Livesu et al. 2021]
Pairing in Octree grids
Pairing in Octree grids
Pairing in Octree grids

grid pairing

tree pairing
Pairing in Octree grids
Pairing in Octree grids

grid pairing

NO tree pairing
Tree pairing vs NO our pairing

State-of-the-art tree pairing

Num cells
25 → 52

too much over-refinement 😞

our pairing

Num cells
25 → 28

less over-refinement! 😊
Our Contribution
Contributions

• Refinement on vertices

• Guarantee the pairing condition through an ILP
  • Linear objective function
  • All linear constraints

• Get rid of the tree structure constraints for balancing and pairing
ILP Formulation for Regular Grids

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th></th>
<th>1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
ILP Formulation for Regular Grids

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

`0 = \begin{array}{c} \end{array}`

`1 = \begin{array}{c} \end{array}`

every refined cell must fit into a non-overlapping 2x2 mask
ILP Formulation for Regular Grids

every refined cell must fit into a non-overlapping 2x2 mask
ILP Formulation for Regular Grids

every refined cell must fit into a non-overlapping 2x2 mask
ILP Formulation for Regular Grids

cells touching the center of the mask are refined
minimize the number of refined cells

\[
\min \sum_{c \in G} \left( \sum_{v \in c} r(v) - r(c) \right)
\]

s.t.

\[
\forall c \in G, \quad \sum_{v \in c} r(v) \geq r(c)
\]

\[
\forall i,j \in N_p \quad r(v_i) + r(v_j) \leq 1
\]
ILP Formulation for Adaptive Grids

\[
\min_{r(v)} E = \sum_{c \in G} \left( \sum_{v \in c} r(v) - r(c) \right)
\]

subject to:

\[
\forall c \in G, \sum_{v \in c} r(v) \geq r(c)
\]

\[
\forall i, j \in N_P \quad r(v_i) + r(v_j) \leq 1
\]

the refinement assigned to a cell must always be lesser or equal to the refinement assigned to its vertices
ILP Formulation for Adaptive Grids

\[
\min_{r(v)} E = \sum_{c \in G} \left( \sum_{v \in c} r(v) - r(c) \right)
\]

s.t.

\[
\forall c \in G, \quad \sum_{v \in c} r(v) \geq r(c)
\]

\[
\forall i, j \in N_P \quad r(v_i) + r(v_j) \leq 1
\]

two vertices whose minors overlap cannot be both refined
2x2x2 minors can’t overlap → pairing guarantee
ILP Formulation for Adaptive Grids

\[
\min_{r(v)} E = \sum_{c \in G} \left( \sum_{v \in c} r(v) - r(c) \right)
\]

s.t.

\[
\forall c \in G, \quad \sum_{v \in c} r(v) \geq r(c)
\]

\[
\forall i,j \in N_p \quad r(v_i) + r(v_j) \leq 1
\]
ILP Formulation for Adaptive Grids

balanced grid

global grid update

binary sub-grid

ILP solution
Results
Results

More than half the grid size if compared to [Gao et al. 2019]

- [Gao et al. 2019] wins (0/202 models)
- Our method wins (194/202 models)
- Same size (8/202 models)
Results

<table>
<thead>
<tr>
<th>Our method relative growth</th>
<th>[Livesu et al. 2021] method relative growth</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>200%</td>
<td>200%</td>
</tr>
<tr>
<td>300%</td>
<td>300%</td>
</tr>
<tr>
<td>400%</td>
<td>400%</td>
</tr>
<tr>
<td>500%</td>
<td>500%</td>
</tr>
<tr>
<td>600%</td>
<td>600%</td>
</tr>
<tr>
<td>700%</td>
<td>700%</td>
</tr>
<tr>
<td>800%</td>
<td>800%</td>
</tr>
</tbody>
</table>

[Livesu et al. 2021] wins (0/202 models)

Our method wins (194/202 models)

same size (8/202 models)

~ half the grid size if compared to [Livesu et al. 2021]!
Results

- Input adaptive grid
  - [Livesu et al. 2021] 4.1X
  - [Gao et al. 2019] 5.3X
  - Ours 2.9X
Results

input adaptive grid

[Livesu et al. 2021]
4.2X

[Gao et al. 2019]
5.1X

ours
2.3X
Results

- input adaptive grid
- [Livesu et al. 2021] 4.1X
- [Gao et al. 2019] 4.1X
- ours 1.9X
nose of Max Planck model collapsed on a face

details are completely lost
Applications

nose of Max Planck model collapsed on a face

some details are preserved but at the cost of an excessive refinement
Applications

nose of Max Planck model collapsed on a face

details are preserved
Applications

fingers collapsed on a face

8^3 uniform grid
32^3 uniform grid
adaptive grid
Applications

not restricted to cube shapes
Conclusion
Conclusion
Future Works

Input paired grid

1 1 1 1
1 1
1 1
1 1
1 1 1 1

Our output

1 1 1 1
1 1 1
1 1 1
1 1 1
1 1 1 1
Code and Demo are available!

https://github.com/cg3hci/Gen-Adapt-Ref-for-Hexmeshing
Optimal Dual Schemes for Adaptive Grid Based Hexmeshing

M. Livesu, L. Pitzalis, G. Cherchi

*ACM Transaction on Graphics 2021*
Thanks for your attention

Presented by
Luca Pitzalis