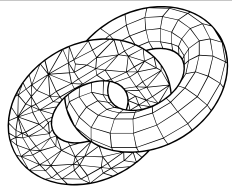


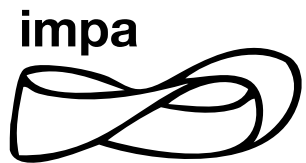
M4G - A Surface Representation for Adaptive CPU-GPU Computation

André Maximo



VisgrafLab

Vision and Graphics Lab
Institute of Pure and Applied Mathematics



Trimester Program on
Computational Manifolds and Applications

November 22, 2011
Rio de Janeiro, Brazil

introduction

introduction

Manifolds-for-GPUs (M4G)

introduction
representation
final remarks

Goal

Surface representation

Dynamic and **adaptive**

CPU-GPU computation

Source

Scanned triangular **meshes**

Application (e.g.)

Progressive **visualization**

introduction

Manifolds-for-GPUs (M4G)

introduction
representation
final remarks

Goal

Surface representation

Dynamic and **adaptive**

CPU-GPU computation

Source

Scanned triangular **meshes**

Application (e.g.)

Progressive **visualization**

Considerations

Ongoing work

Make several assumptions

Use many known techniques

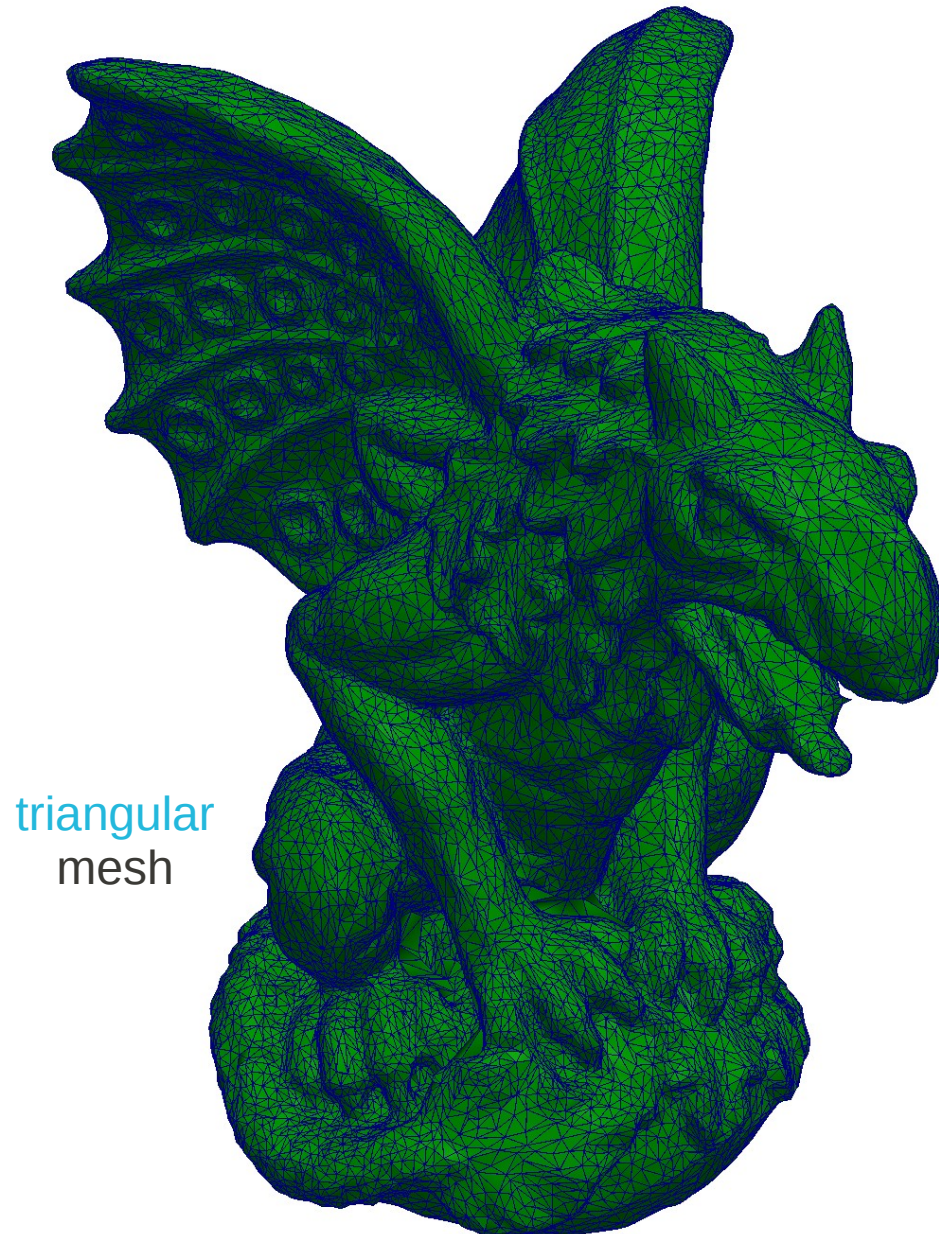
The "**manifold**" word is used lightly

introduction

basic :: representation



surface



triangular
mesh

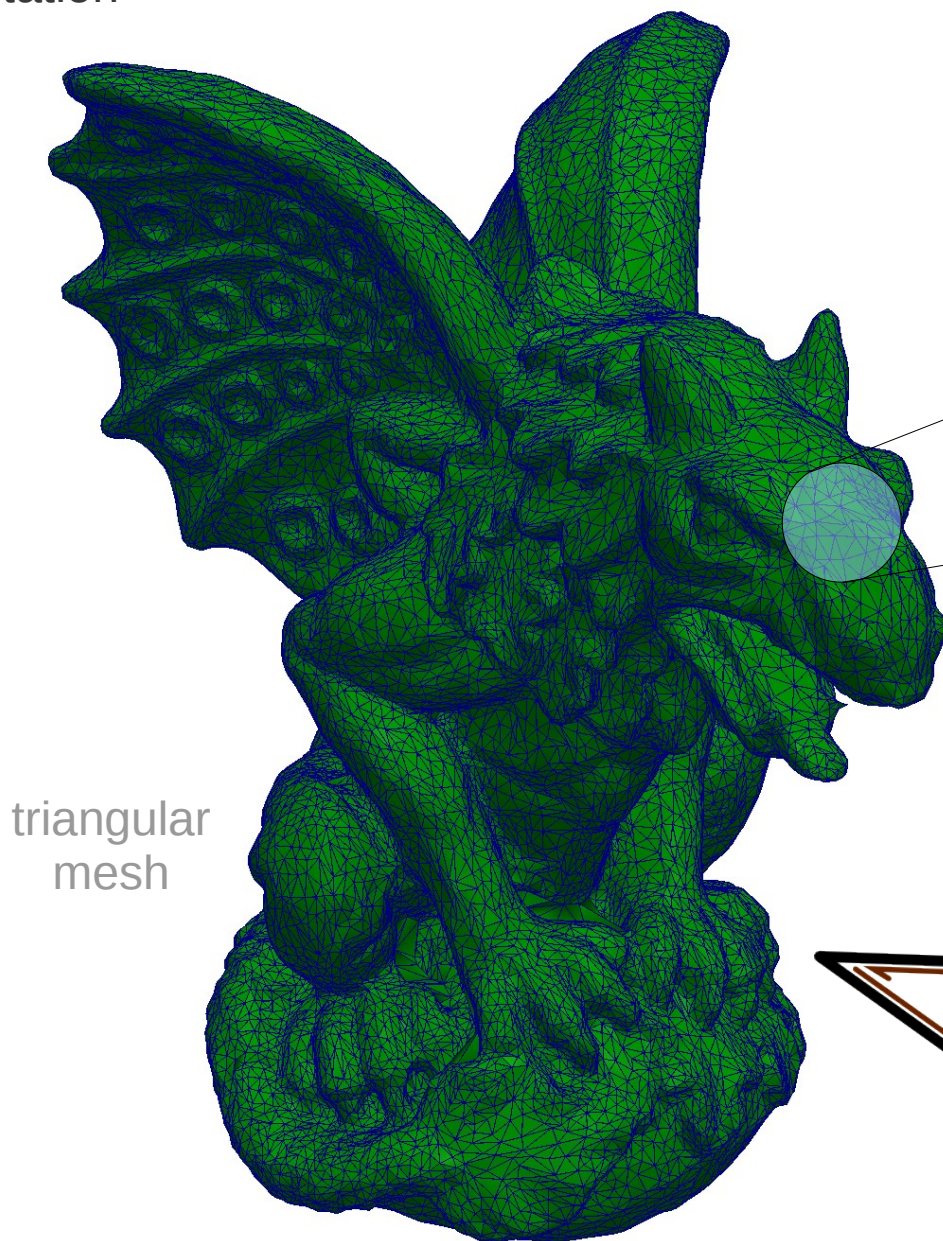
introduction
representation
final remarks

introduction

basic :: representation

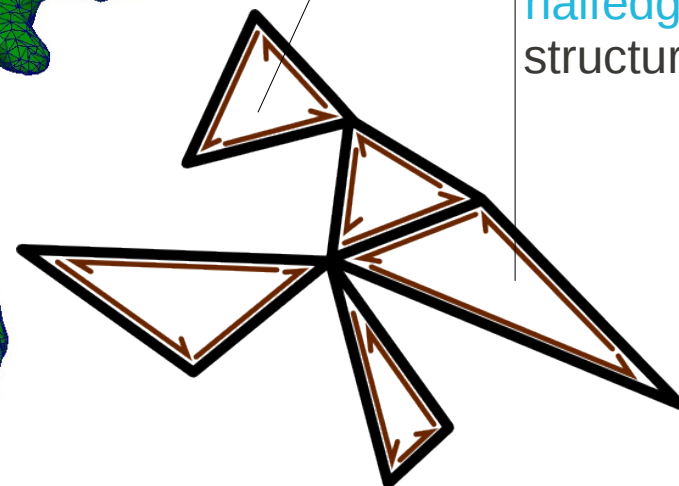
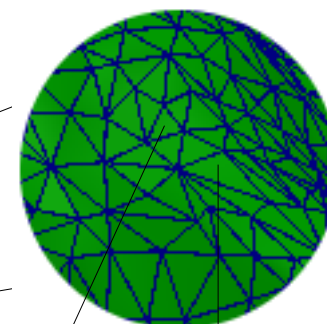


surface



triangular
mesh

introduction
representation
final remarks



halfedge
structure

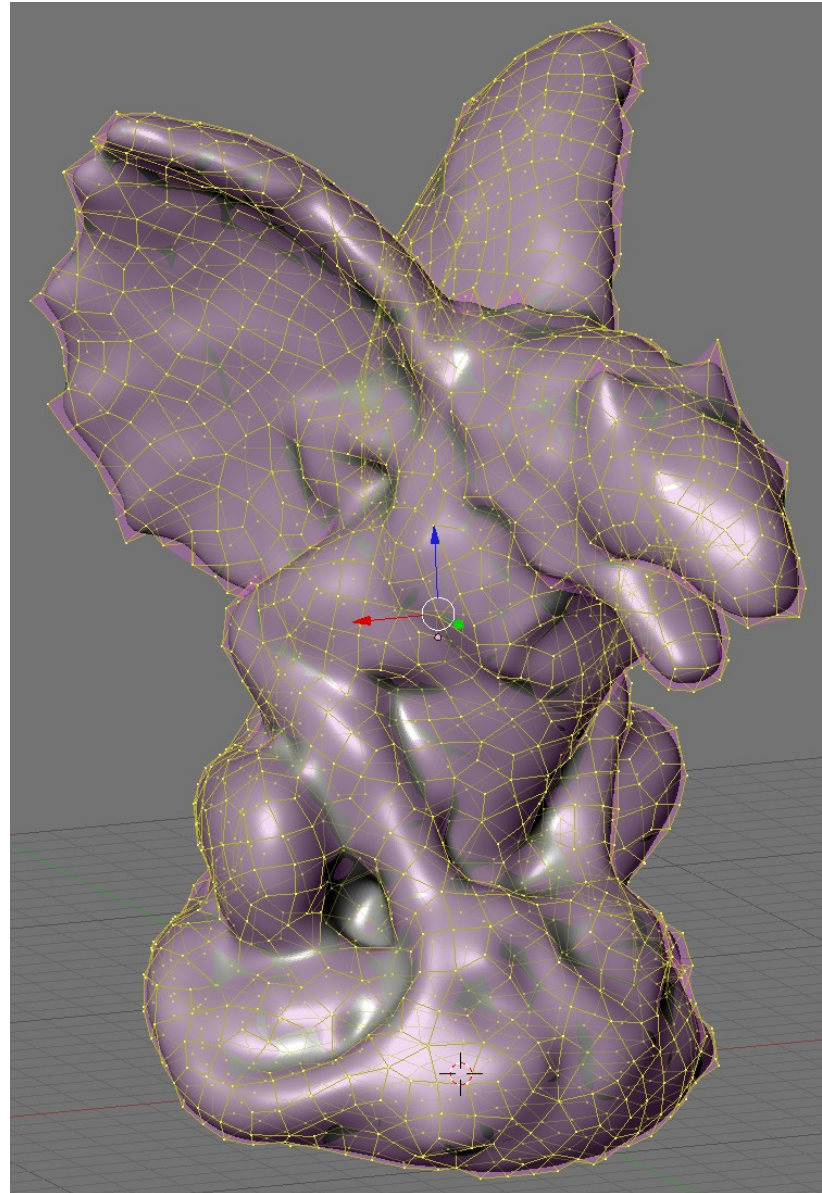
introduction

basic :: subdivision



surface

quadrangular
base mesh



introduction
representation
final remarks

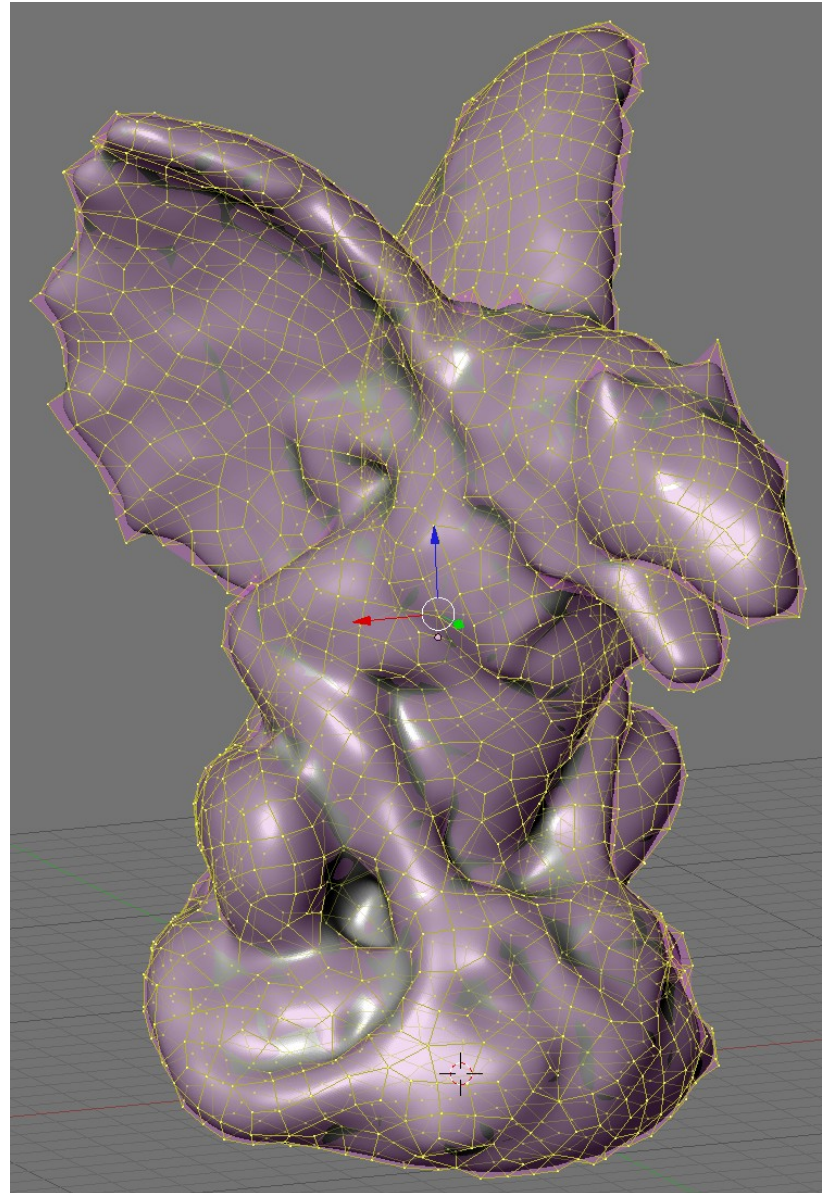
introduction

basic :: subdivision

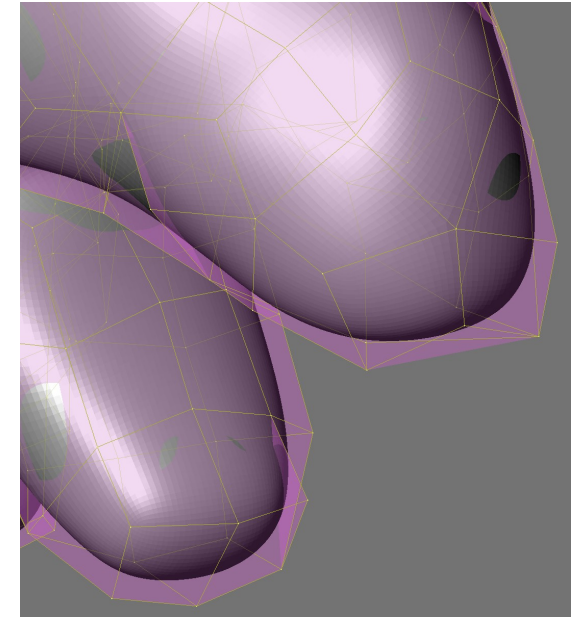


surface

quadrangular
base mesh



introduction
representation
final remarks



Catmull-Clark
subdivision

introduction

our goal



surface

introduction
representation
final remarks

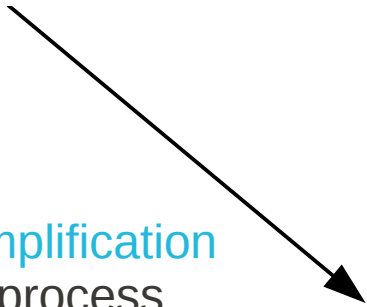
introduction

our goal



surface

simplification
process



base mesh

introduction
representation
final remarks

introduction

our goal



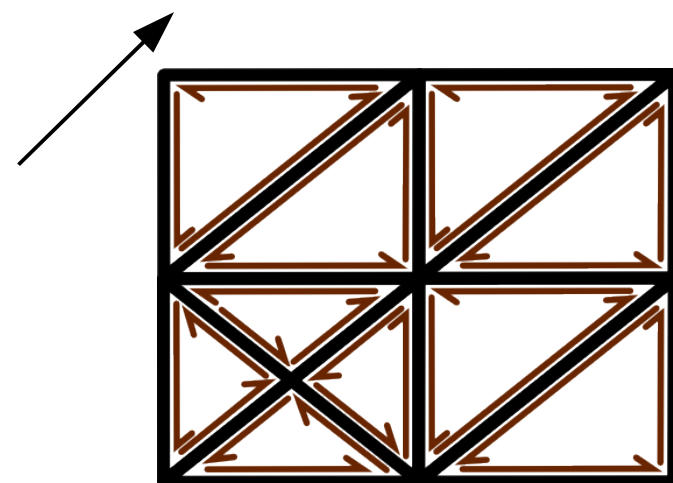
surface



base mesh

introduction
representation
final remarks

4-8 mesh



introduction

our goal

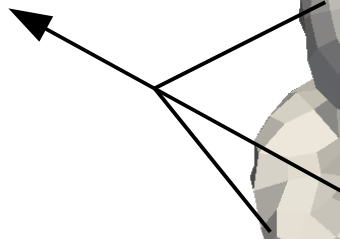


surface



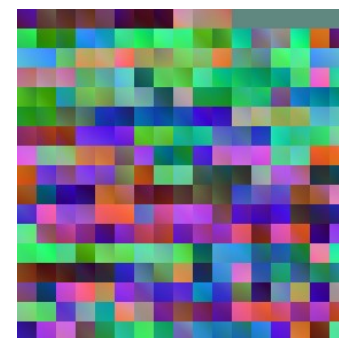
base mesh

collection of
charts

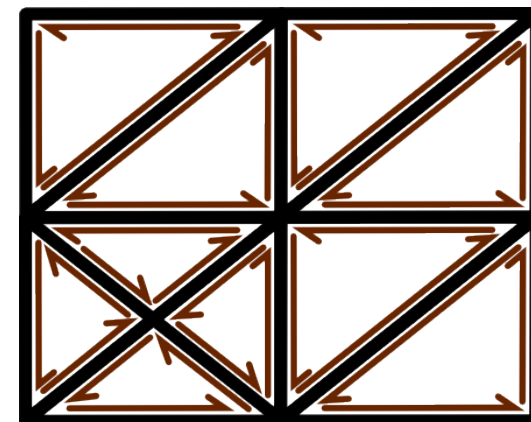


introduction
representation
final remarks

atlas



4-8 mesh

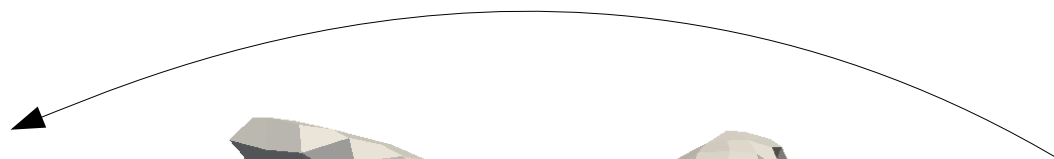


introduction

our goal

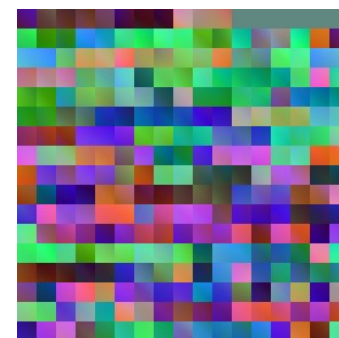


surface

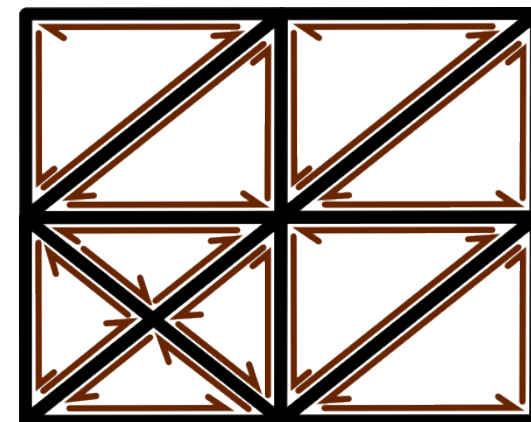


introduction
representation
final remarks

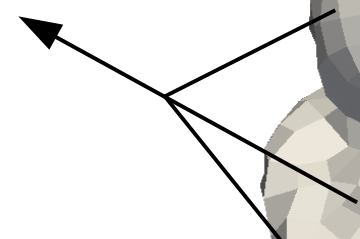
atlas



4-8 mesh



collection of
charts



base mesh

representation

representation

problem statement



surface

+



base mesh

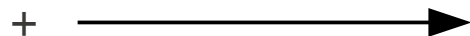
introduction
representation
final remarks

representation

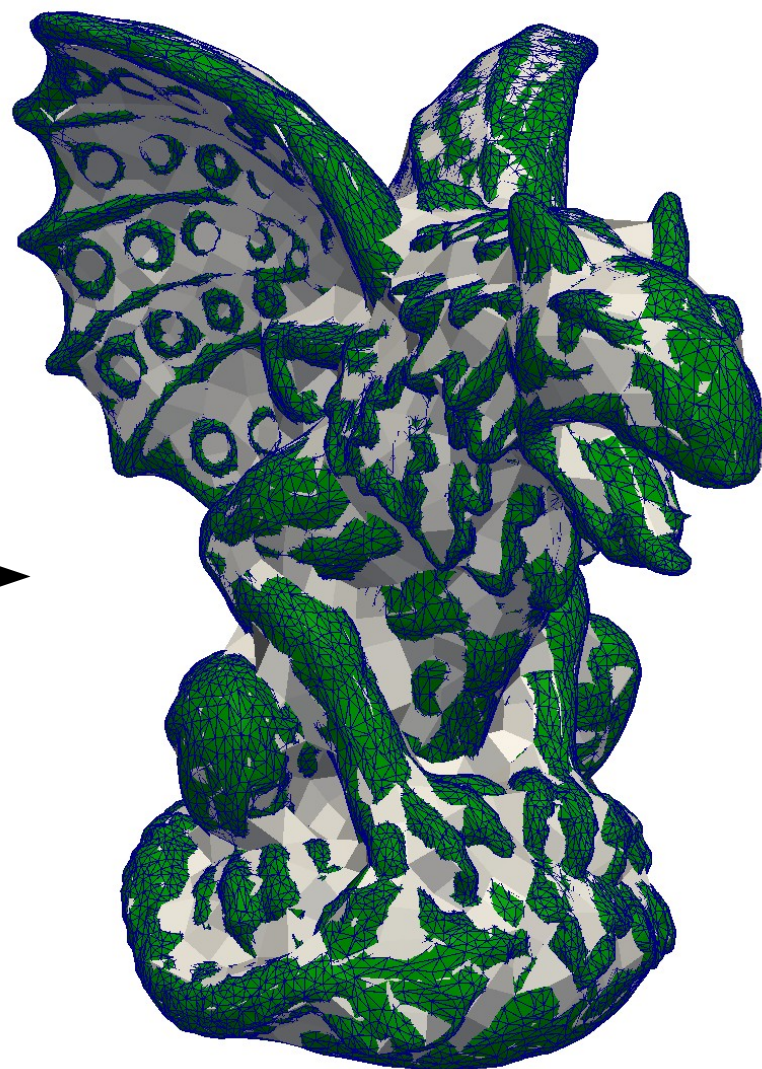
problem statement



surface



base mesh



how to build a [correspondence](#)?

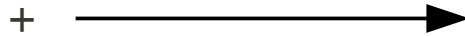
introduction
representation
final remarks

representation

problem statement



surface



base mesh



introduction
representation
final remarks

Multi-resolution structure

Low-level in the CPU

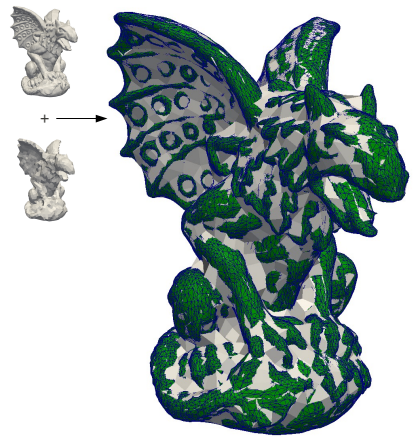
High-level in the GPU

Dynamic controllable

how to build a correspondence?

representation

problem statement

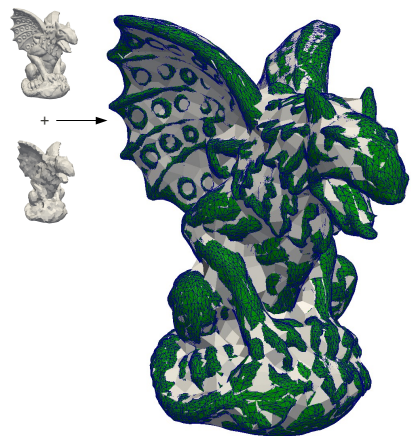


correspondence

introduction
representation
final remarks

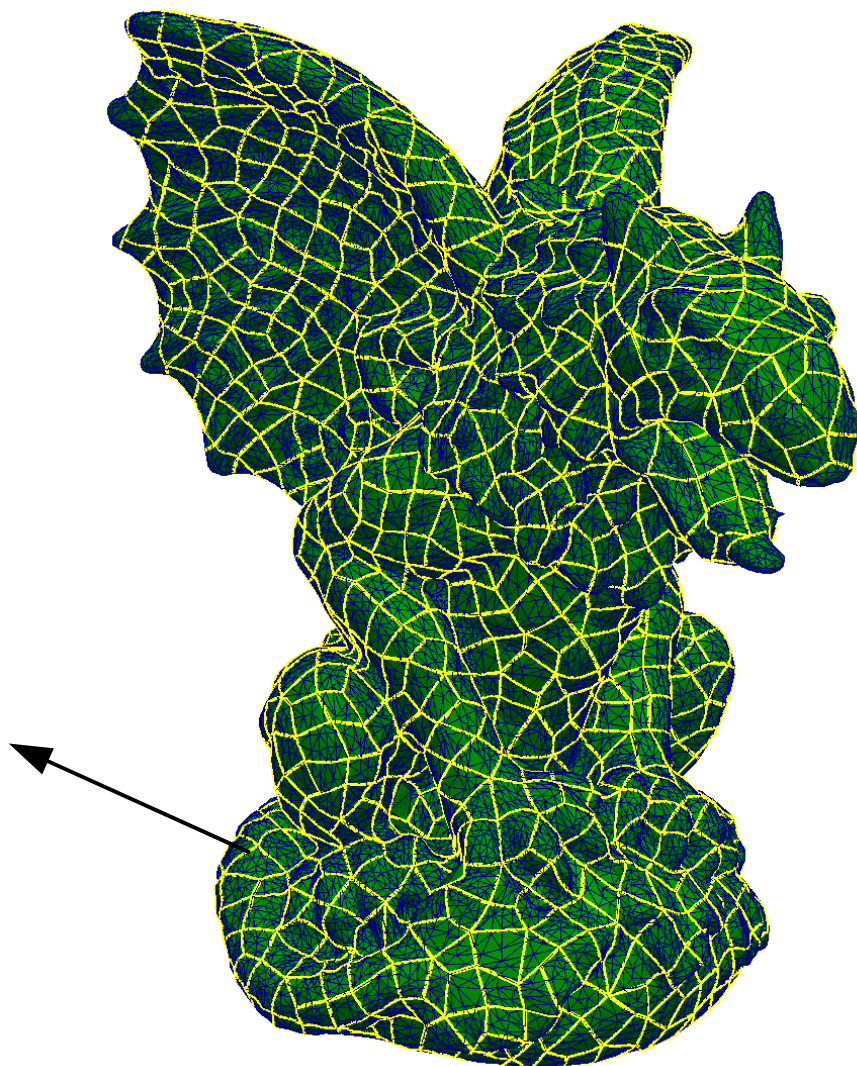
representation

problem statement



correspondence

chart
boundaries

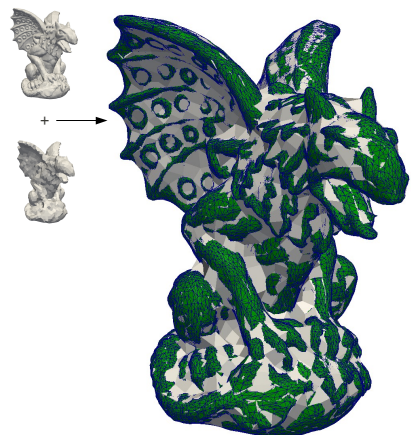


build an atlas

introduction
representation
final remarks

representation

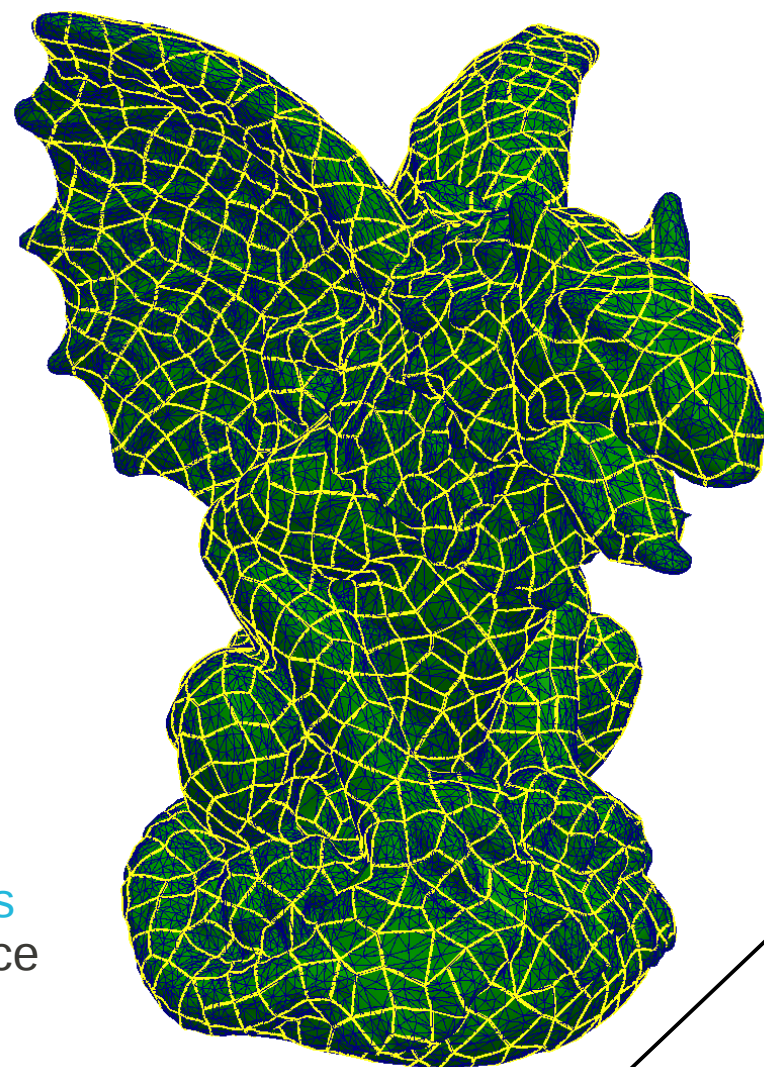
problem statement



correspondence

chart
boundaries

finding **curves**
over the surface



build an atlas

parameterizations:
curves (**1D**) and
interior (**2D**)

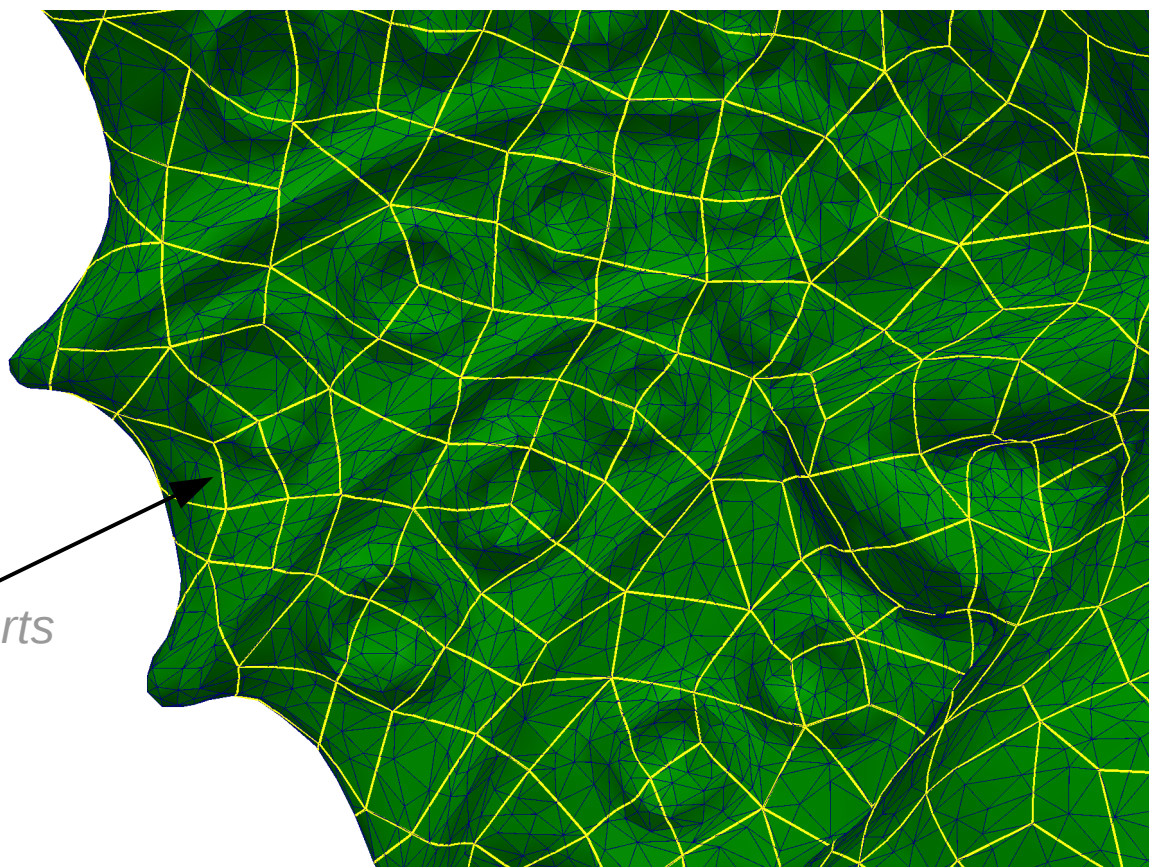
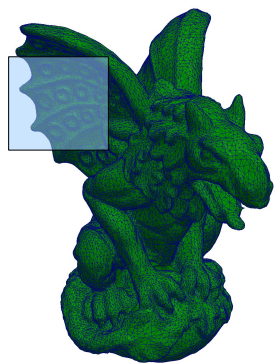
introduction
representation
final remarks

representation

finding curves over the surface

first thought :: [geodesics](#)

introduction
representation
final remarks



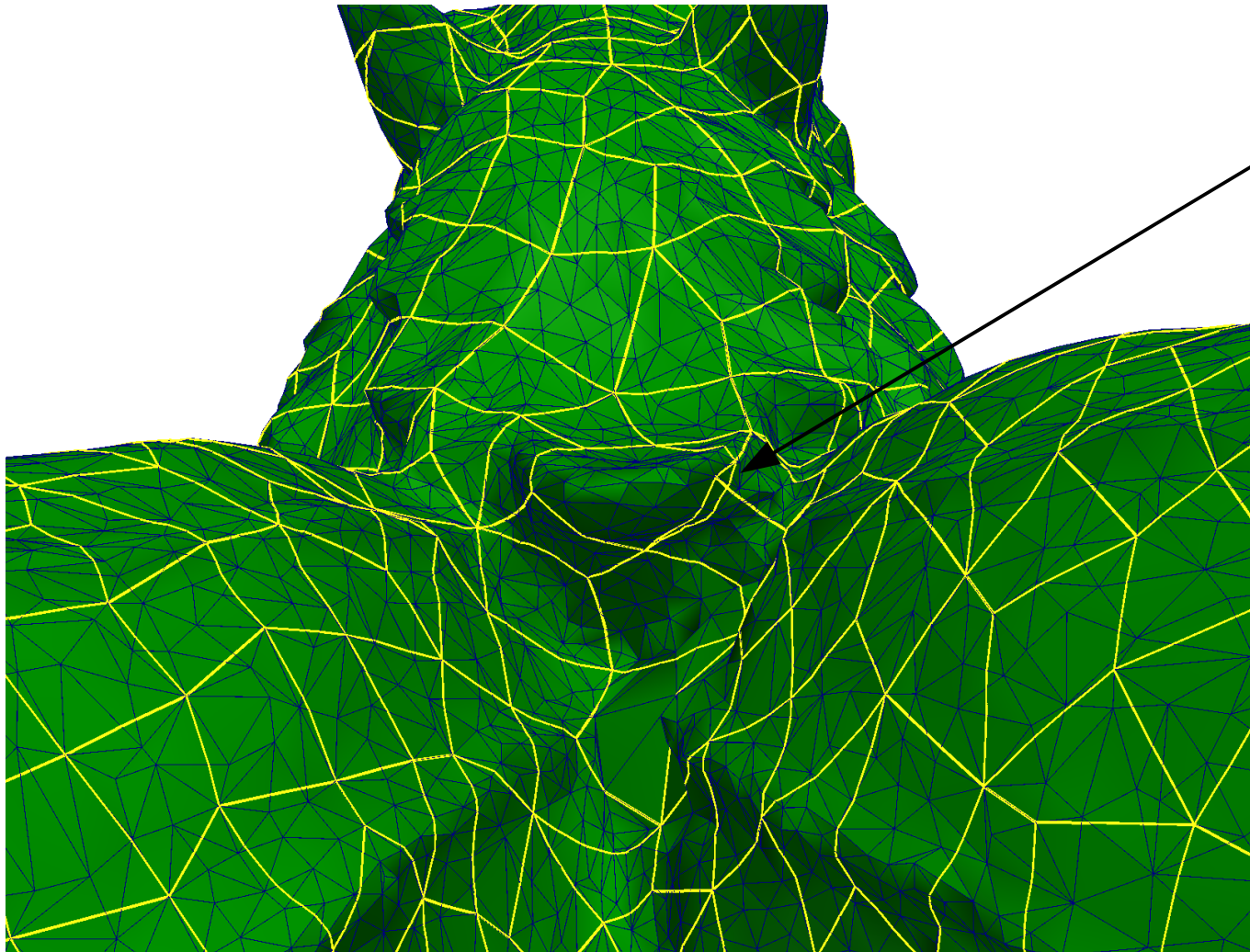
nice grid of charts

representation

finding curves over the surface

problem with **geodesics**

introduction
representation
final remarks



avoiding
high
frequencies



inconsistency
through chart
boundaries

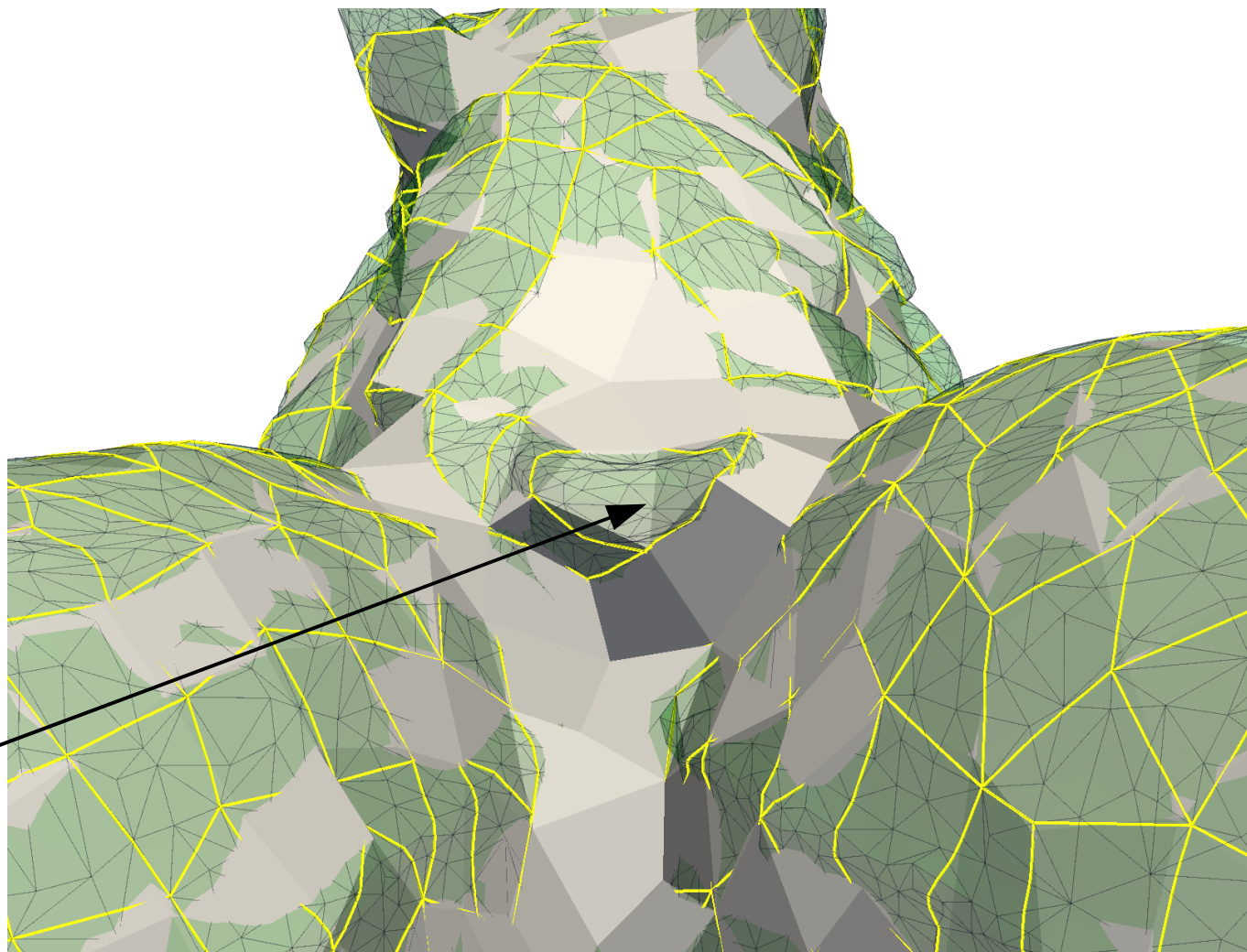
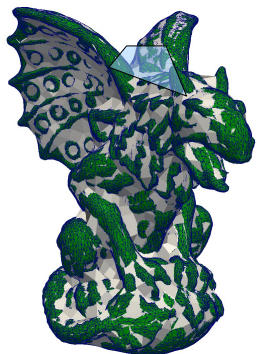
how to find
better **curves**
over the
surface?

representation

finding curves over the surface

problem with **geodesics**

introduction
representation
final remarks



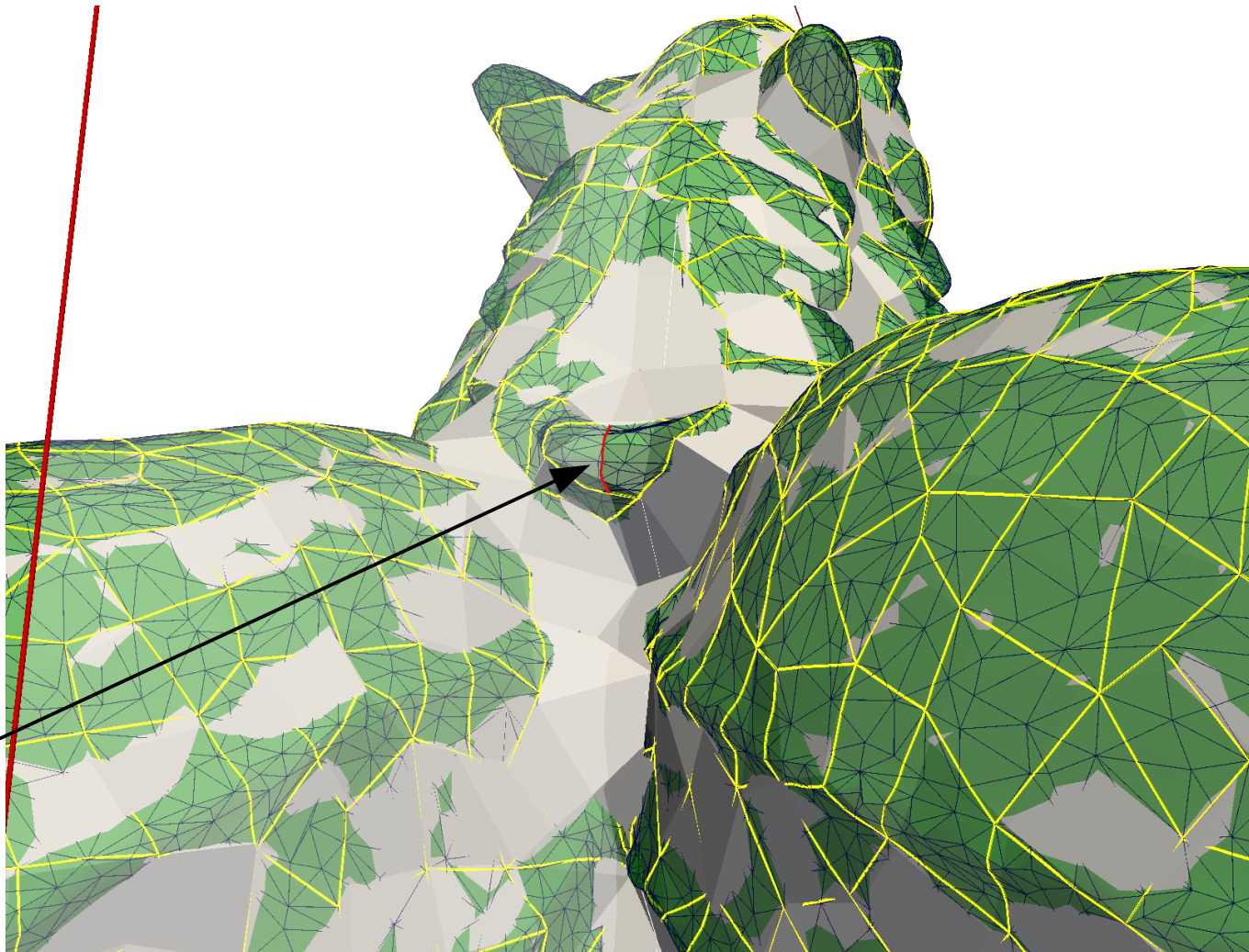
corresponding
boundary
curve

representation

finding curves over the surface

work in **progress** :: new algorithm

introduction
representation
final remarks

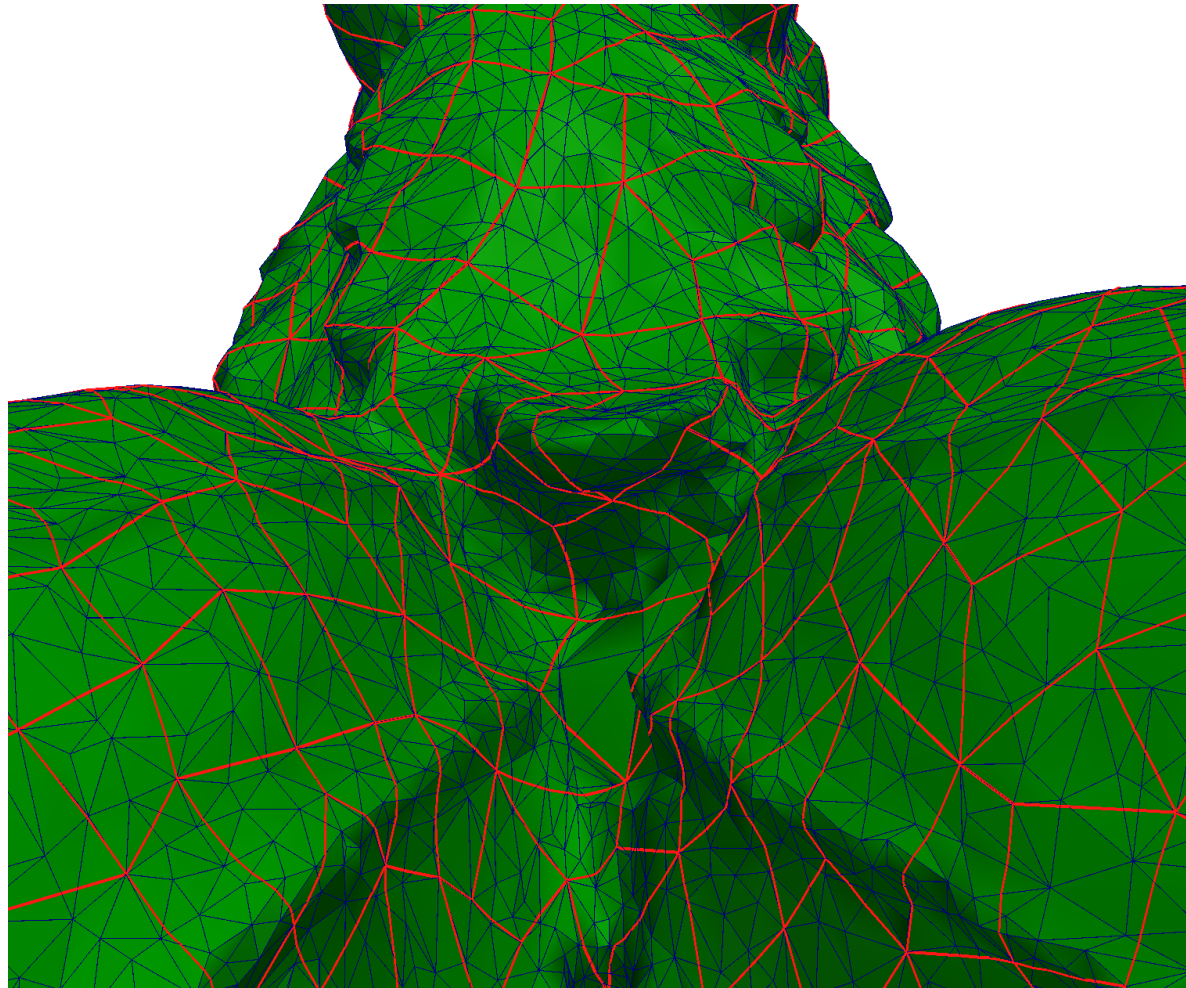


boundary curve
based on
cutting plane

representation

finding curves over the surface

work in progress :: new algorithm



introduction
representation
final remarks

assign an
"edge normal"
based on
the base mesh
faces

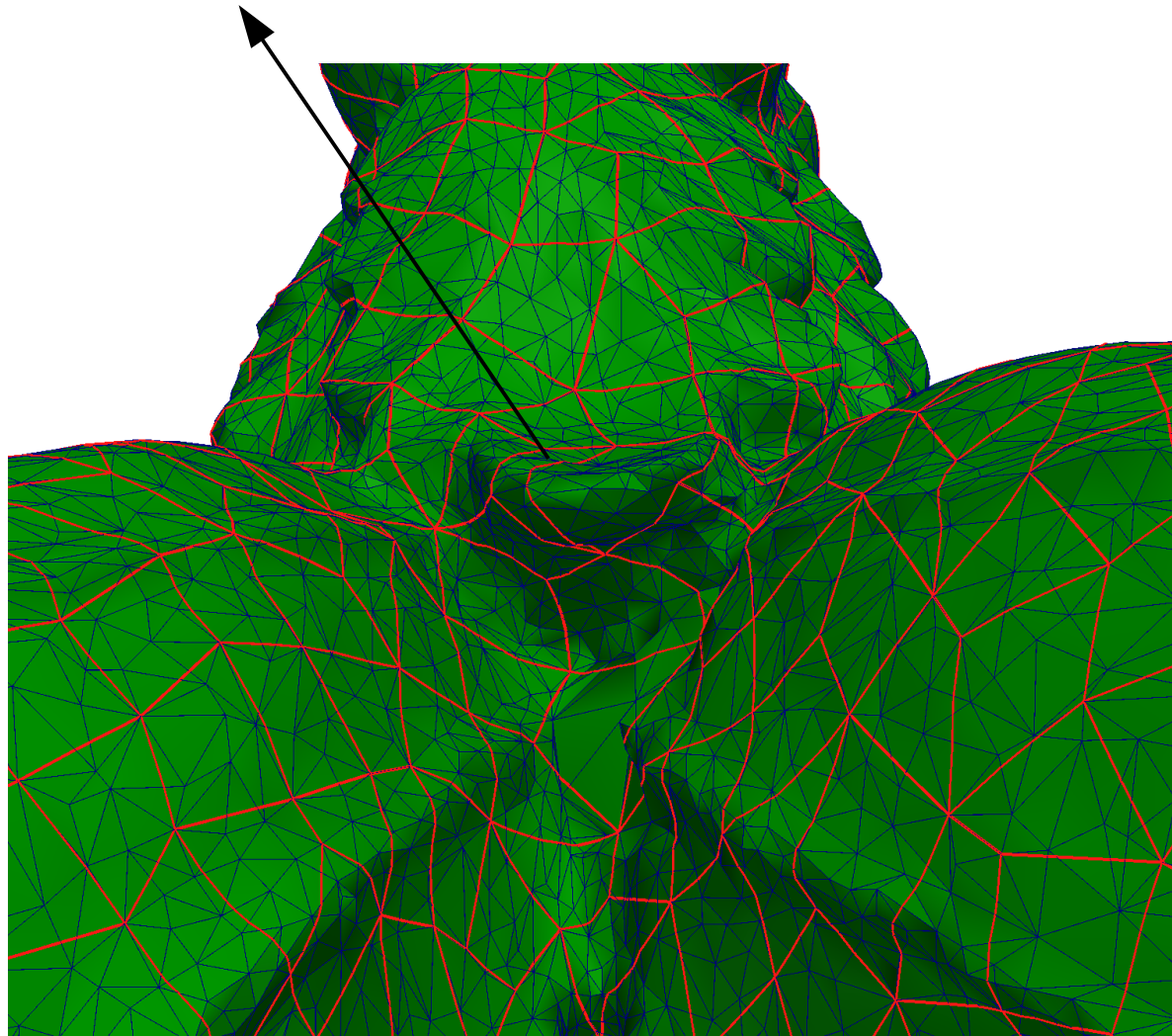
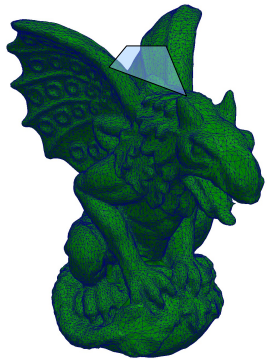


define cutting
planes for each
chart boundary
(base mesh edges)

representation

finding curves over the surface

new algorithm :: problem solved?



introduction
representation
final remarks

assign an
"edge normal"
based on
the base mesh
faces

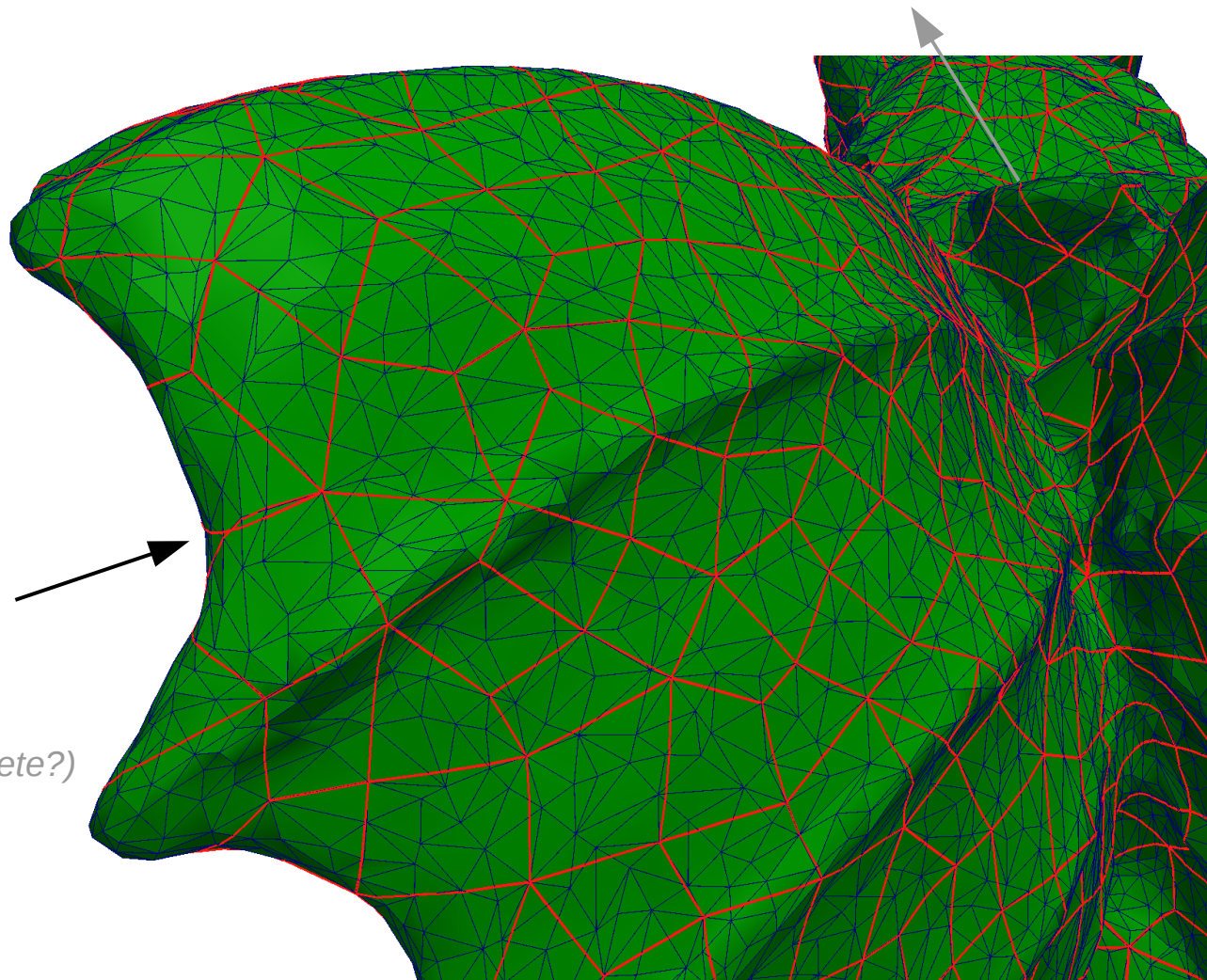


define cutting
planes for each
chart boundary
(*base mesh edges*)

representation

finding curves over the surface

problem with **new algorithm**



matching all cutting planes to **agree** without intersection on curves is difficult
(maybe even NP-Complete?)

introduction
representation
final remarks

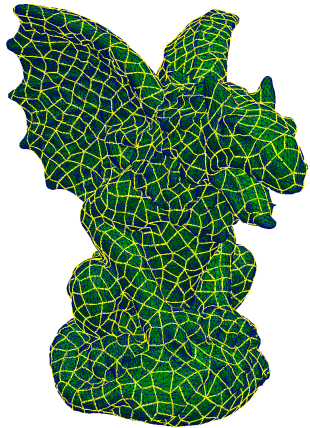
computing without considering others



inconsistency through chart boundaries

representation

building an atlas

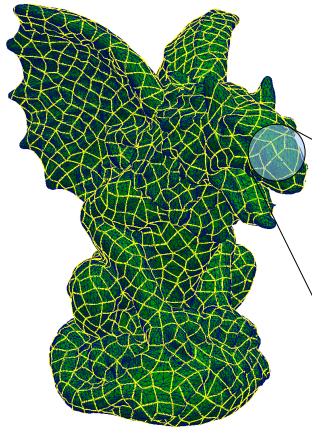


surface
and curves

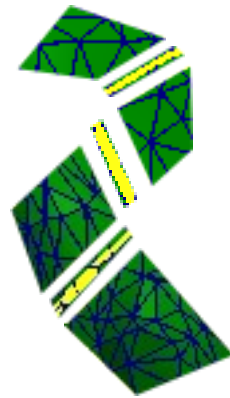
introduction
representation
final remarks

representation

building an atlas



surface
and curves

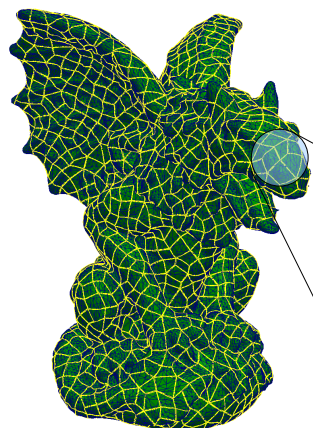


introduction
representation
final remarks



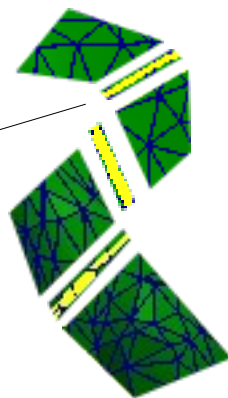
representation

building an atlas



surface
and curves

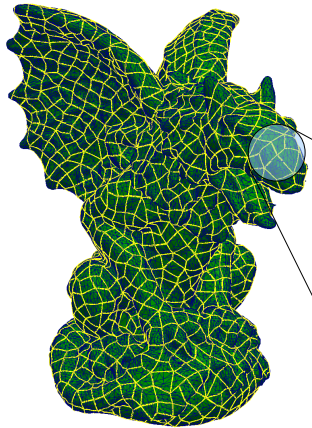
parameterize
curves by
arc-length



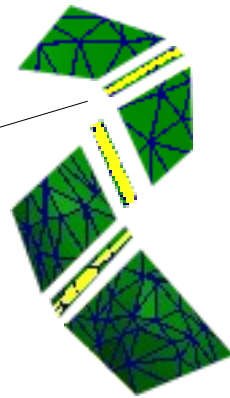
introduction
representation
final remarks

representation

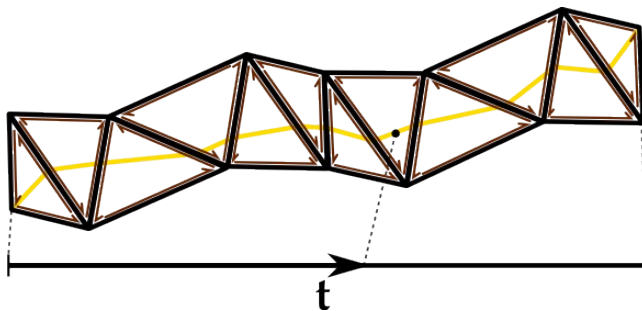
building an atlas



surface
and curves



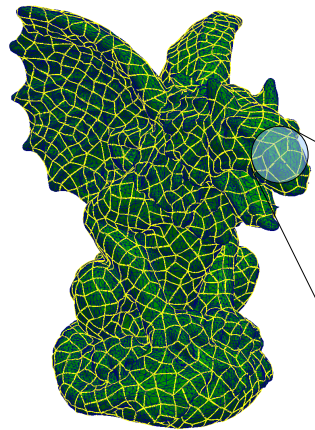
parameterize
curves by
arc-length



introduction
representation
final remarks

representation

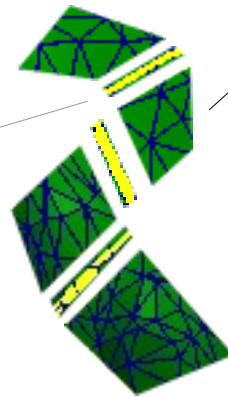
building an atlas



surface
and curves

parameterize interior
using a known technique
(such as [Floater, 2003](#))

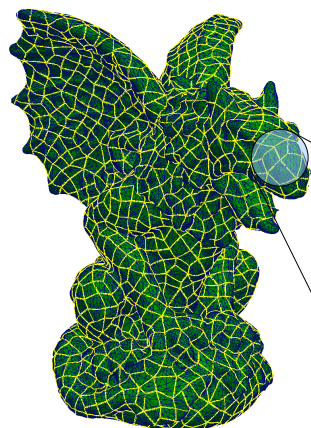
parameterize
curves by
arc-length



representation

building an atlas

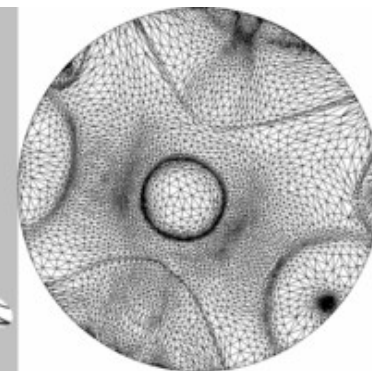
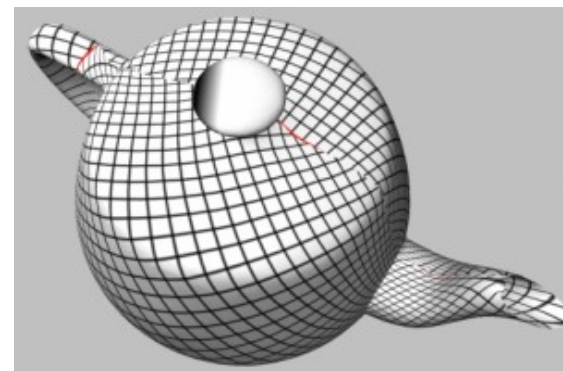
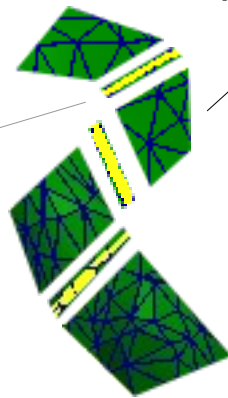
introduction
representation
final remarks



surface
and curves

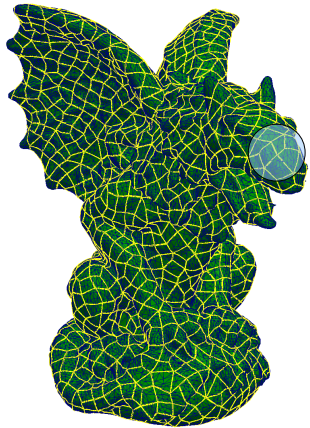
parameterize interior
using a known technique
(such as [Floater, 2003])

parameterize
curves by
arc-length

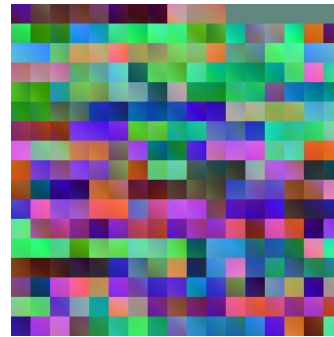


representation

building an atlas



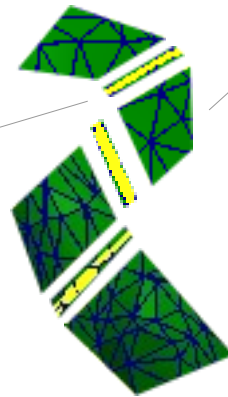
surface
and curves



composed both in
the **M4G atlas**

parameterize interior
using a known technique
(such as [Floater, 2003])

parameterize
curves by
arc-length

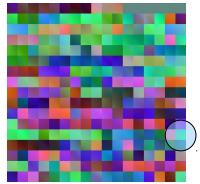


introduction
representation
final remarks

representation

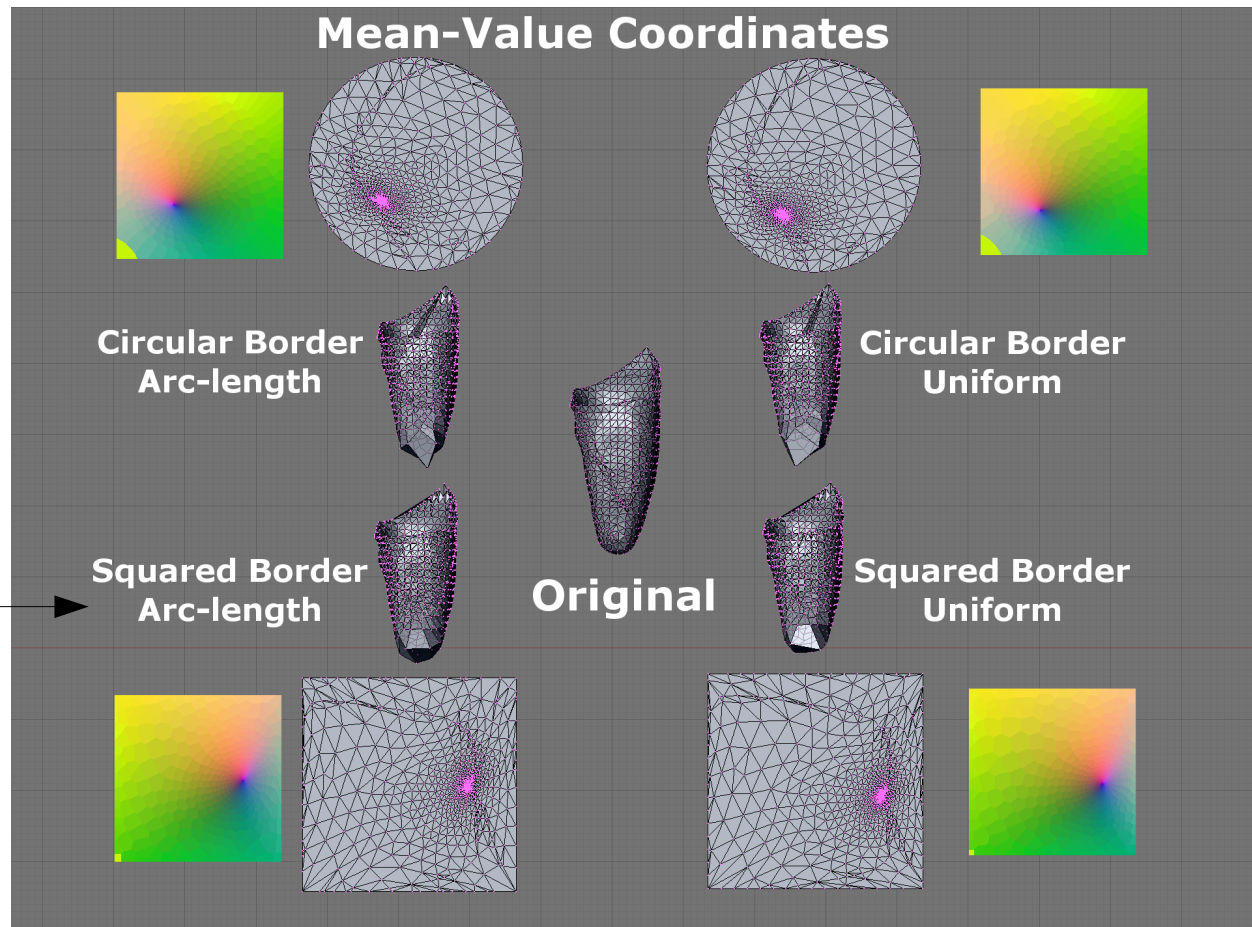
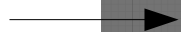
comparing parameterizations

introduction
representation
final remarks



[Floater, 2003]

chosen
result



representation

4-8 mesh



base mesh

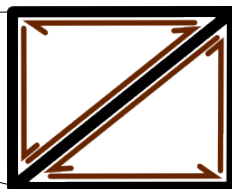
introduction
representation
final remarks

representation

4-8 mesh



tile



(triquad: triangulated quadrangulation)

base mesh

introduction
representation
final remarks

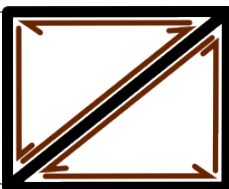
representation

4-8 mesh

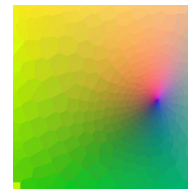


base mesh

tile



chart



introduction
representation
final remarks

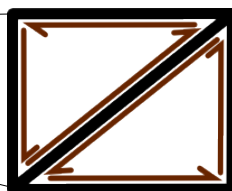
representation

4-8 mesh

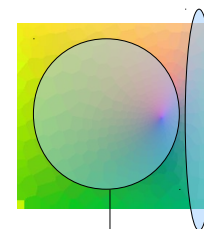


base mesh

tile



chart



1D shared
parameterization

2D separate
parameterization

introduction
representation
final remarks

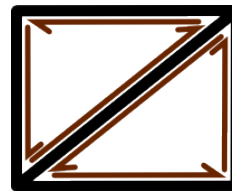
representation

4-8 mesh



base mesh

tile



chart

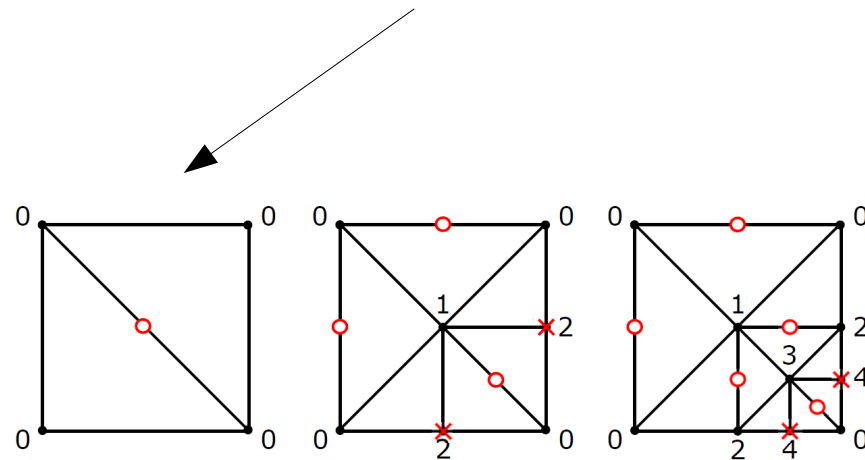
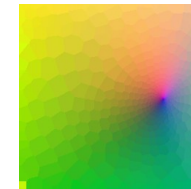


Figure 3: Illustrative example of the 4–8 structure with the corresponding vertex levels. The M4G tiling is refined (from left to right) creating intermediary adaptive meshes. At each resolution level, only a set of edges can be split (with a circle) and a set of vertices can be welded (with a cross).

introduction
representation
final remarks

representation

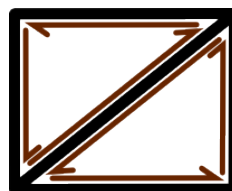
4-8 mesh



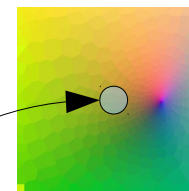
base mesh

introduction
representation
final remarks

tile



chart



lookup surface properties
in the interior of chart

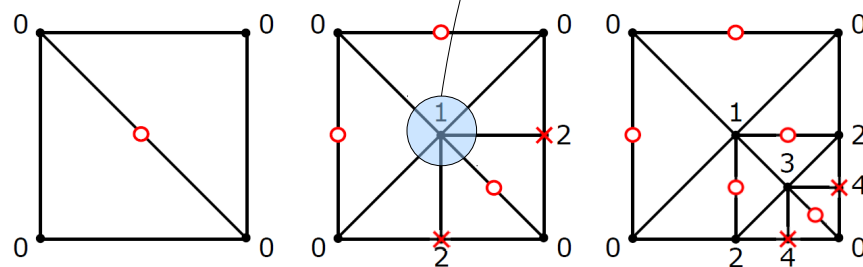


Figure 3: Illustrative example of the 4–8 structure with the corresponding vertex levels. The M4G tiling is refined (from left to right) creating intermediary adaptive meshes. At each resolution level, only a set of edges can be split (with a circle) and a set of vertices can be welded (with a cross).

representation

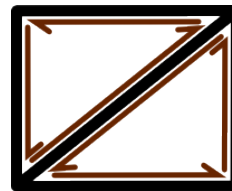
4-8 mesh



base mesh

introduction
representation
final remarks

tile



chart



lookup in the
boundary of chart

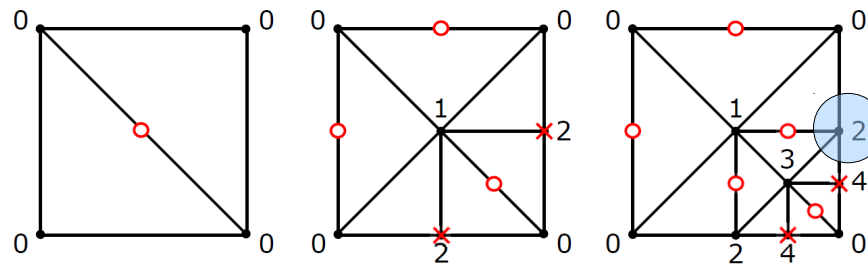


Figure 3: Illustrative example of the 4–8 structure with the corresponding vertex levels. The M4G tiling is refined (from left to right) creating intermediary adaptive meshes. At each resolution level, only a set of edges can be split (with a circle) and a set of vertices can be welded (with a cross).

representation

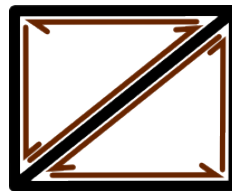
4-8 mesh



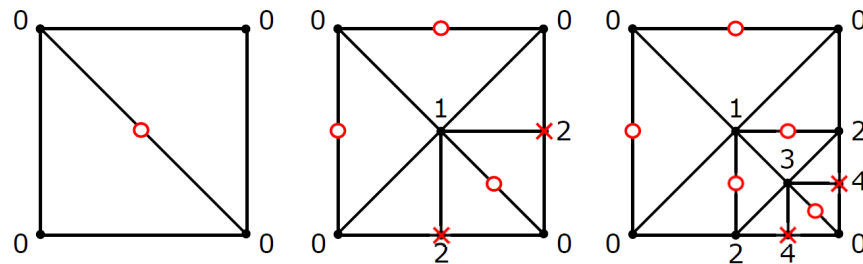
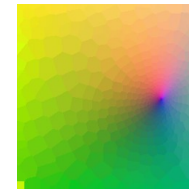
base mesh

introduction
representation
final remarks

tile



chart



refinement and
simplification occur
in the CPU



Figure 3: Illustrative example of the 4–8 structure with the corresponding vertex levels. The M4G tiling is refined (from left to right) creating intermediary adaptive meshes. At each resolution level, only a set of edges can be split (with a circle) and a set of vertices can be welded (with a cross).

representation

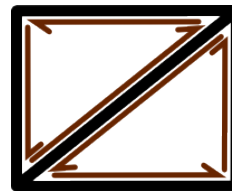
4-8 mesh



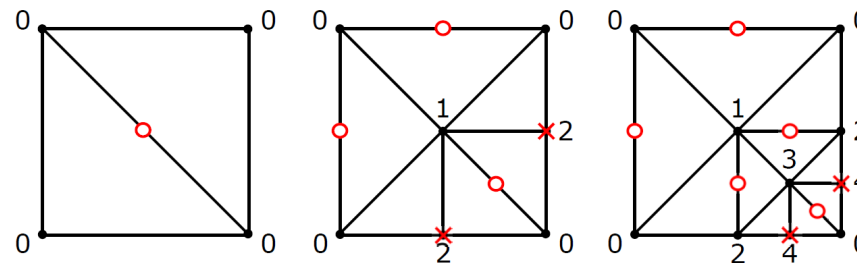
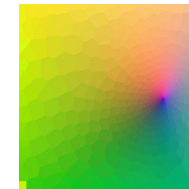
base mesh

introduction
representation
final remarks

tile



chart



refinement and
simplification occur
in the CPU

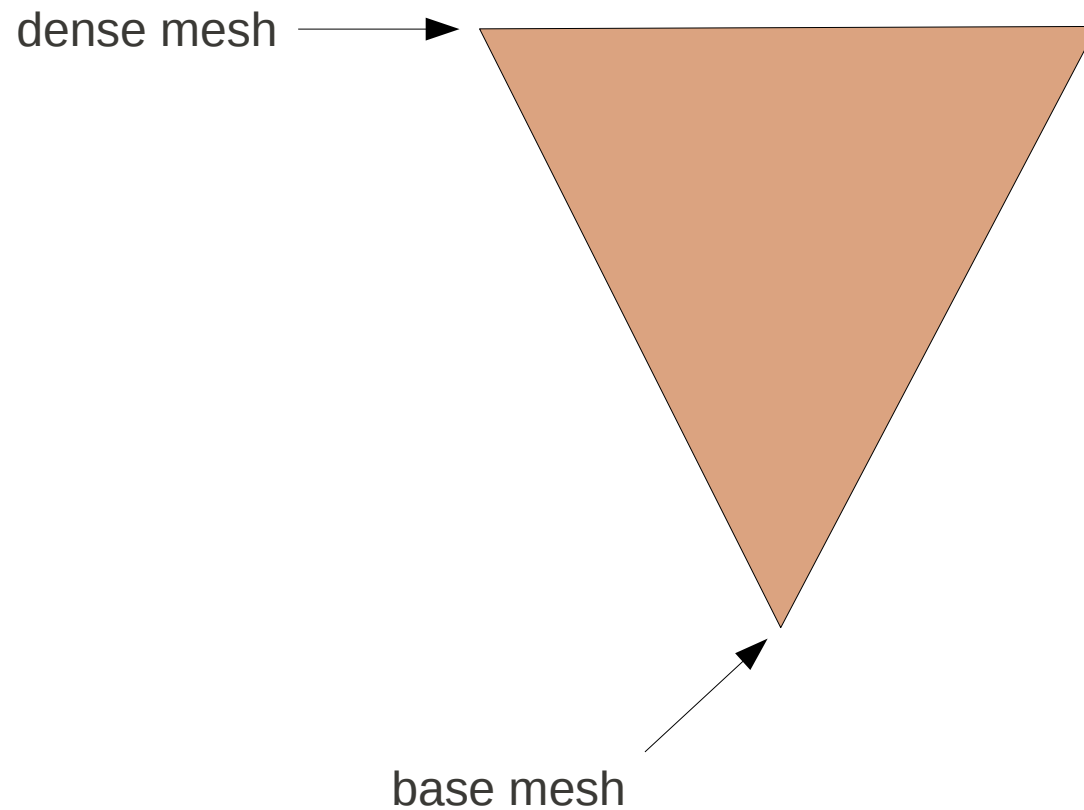


Figure 3: Illustrative example of the 4–8 structure with the corresponding vertex levels. The M4G tiling is refined (from left to right) creating intermediary adaptive meshes. At each resolution level, only a set of edges can be split (with a circle) and a set of vertices can be welded (with a cross).

representation

CPU-GPU coupled computation

introduction
representation
final remarks

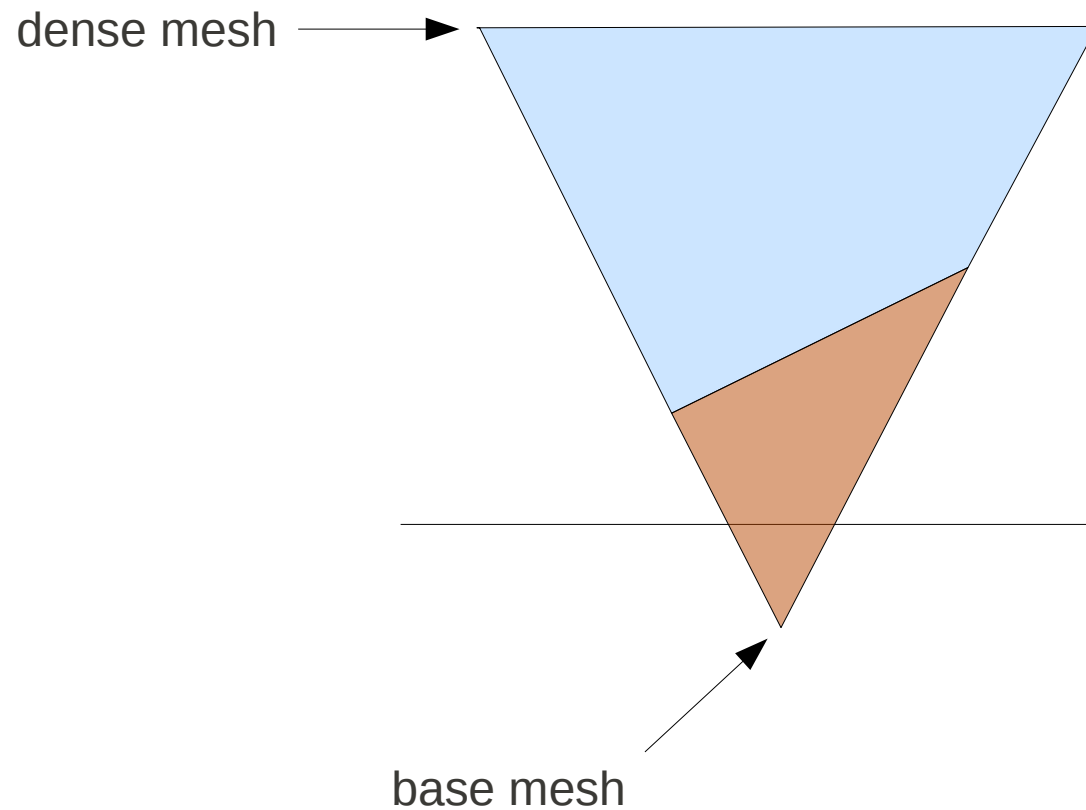


Multi-resolution structure

representation

CPU-GPU coupled computation

introduction
representation
final remarks



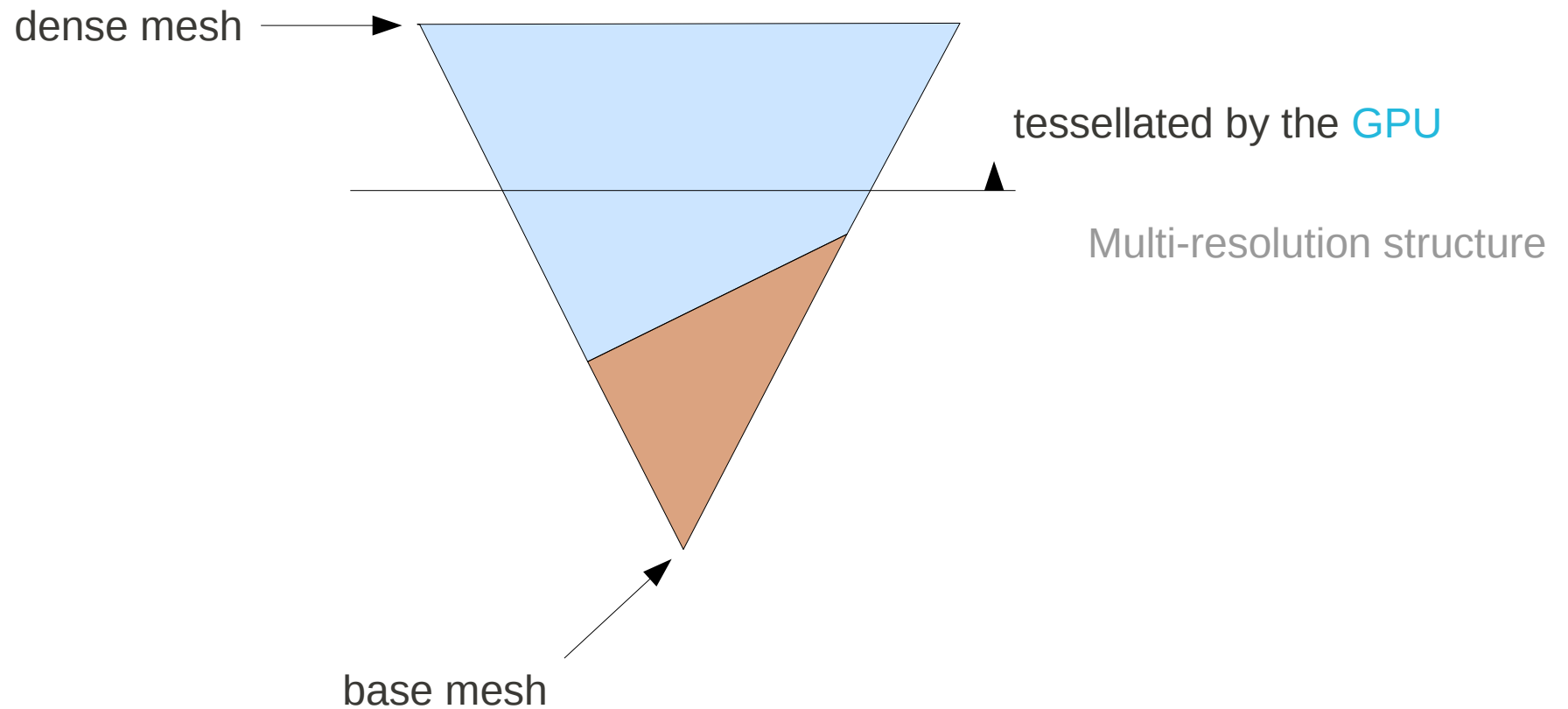
Multi-resolution structure

controlled by the CPU

representation

CPU-GPU coupled computation

introduction
representation
final remarks



representation

CPU-GPU coupled computation

introduction
representation
final remarks

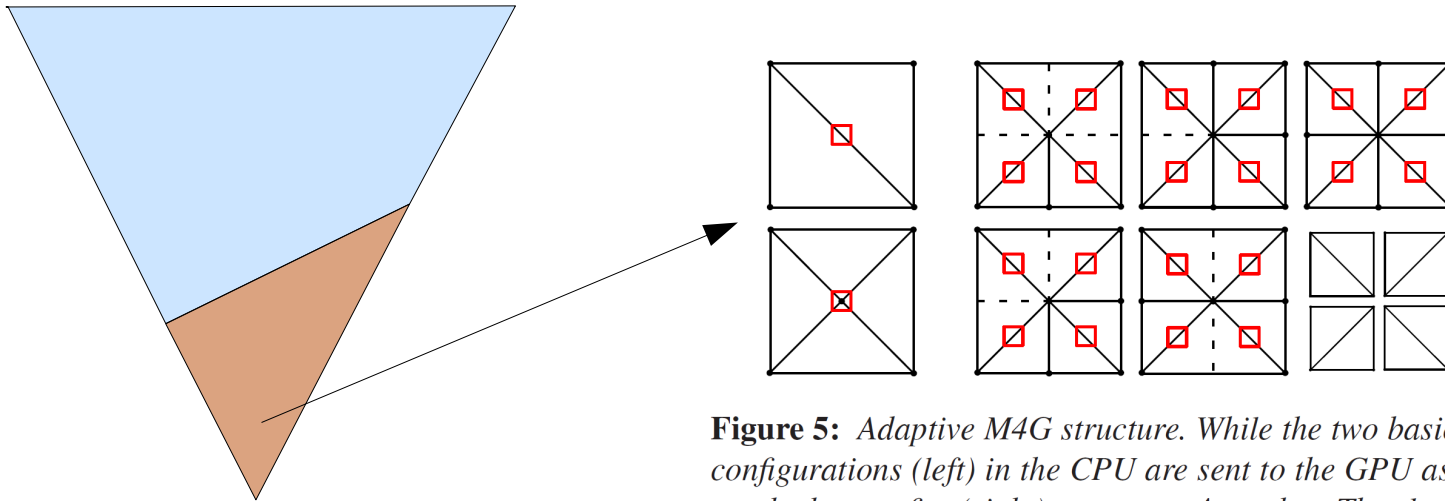


Figure 5: Adaptive M4G structure. While the two basic tile configurations (left) in the CPU are sent to the GPU as one patch, the next five (right) are sent as 4 patches. The elements (vertex or edge) in the CPU data structure representing a patch in the GPU data buffer are denoted with a square.

representation

CPU-GPU coupled computation

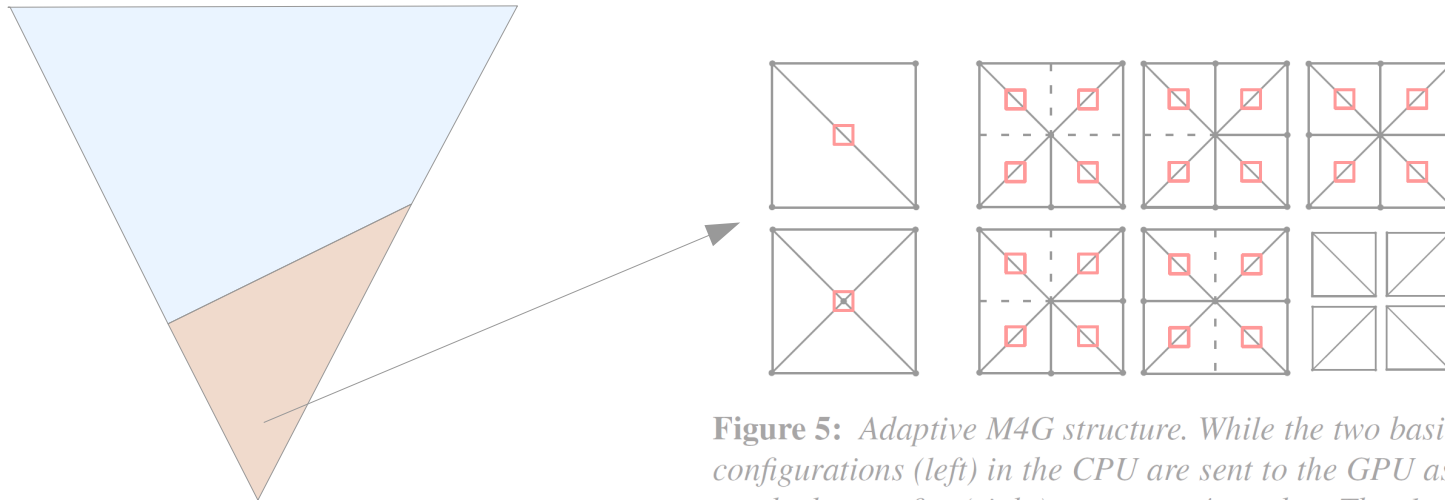
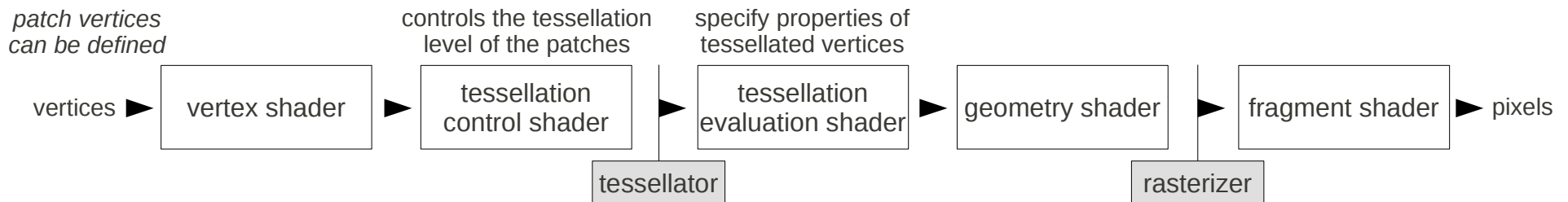


Figure 5: Adaptive M4G structure. While the two basic tile configurations (left) in the CPU are sent to the GPU as one patch, the next five (right) are sent as 4 patches. The elements (vertex or edge) in the CPU data structure representing a patch in the GPU data buffer are denoted with a square.

introduction
representation
final remarks

footnote



representation

CPU-GPU coupled computation

introduction
representation
final remarks

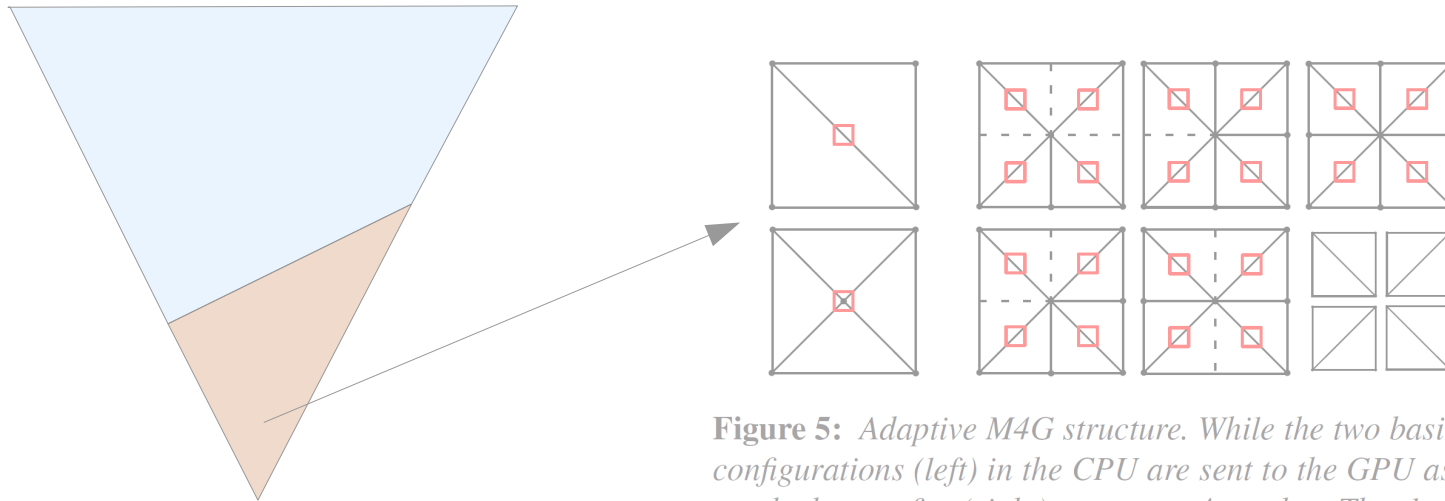
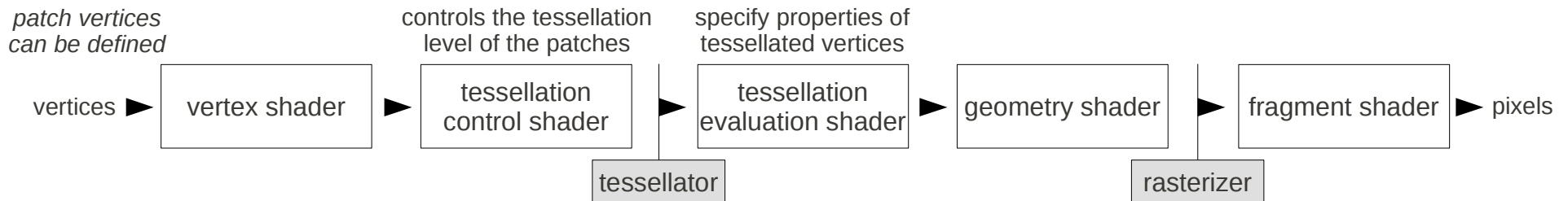


Figure 5: Adaptive M4G structure. While the two basic tile configurations (left) in the CPU are sent to the GPU as one patch, the next five (right) are sent as 4 patches. The elements (vertex or edge) in the CPU data structure representing a patch in the GPU data buffer are denoted with a square.

footnote



final remarks

final remarks

applications

progressive **visualization**

introduction
representation
final remarks

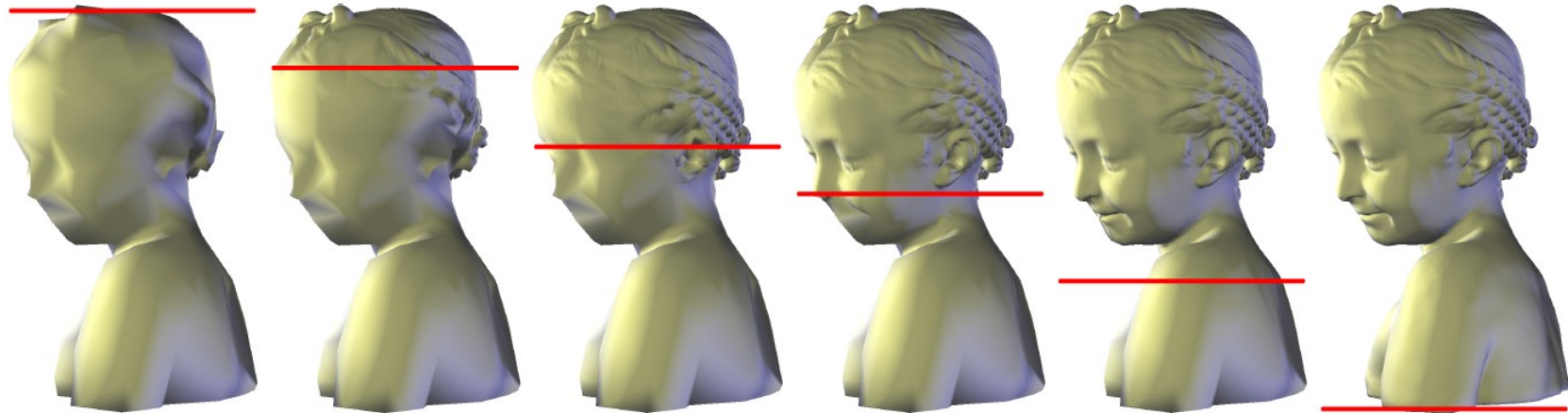


Figure 6: Example application of the Bimba dataset with variable mesh resolution. The surface features of the reconstructed mesh is revealed (from left to right) as the control plane (in red) moves from top to bottom.

final remarks

applications

progressive [visualization](#)

introduction
representation
final remarks

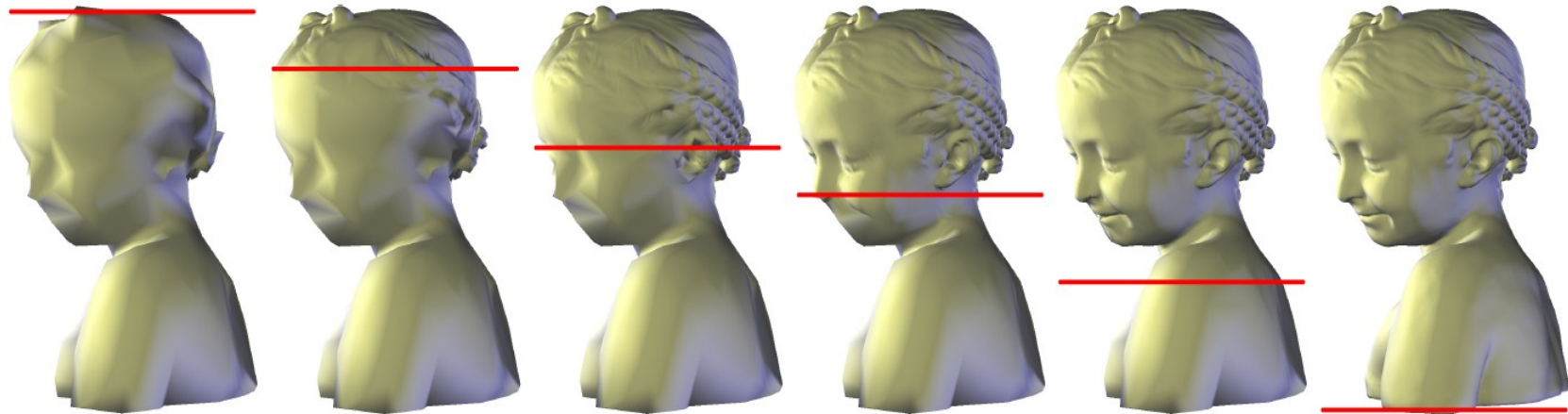


Figure 6: Example application of the Bimba dataset with variable mesh resolution. The surface features of the reconstructed mesh is revealed (from left to right) as the control plane (in red) moves from top to bottom.

final remarks

conclusions

introduction
representation
final remarks

An **adaptive** surface representation

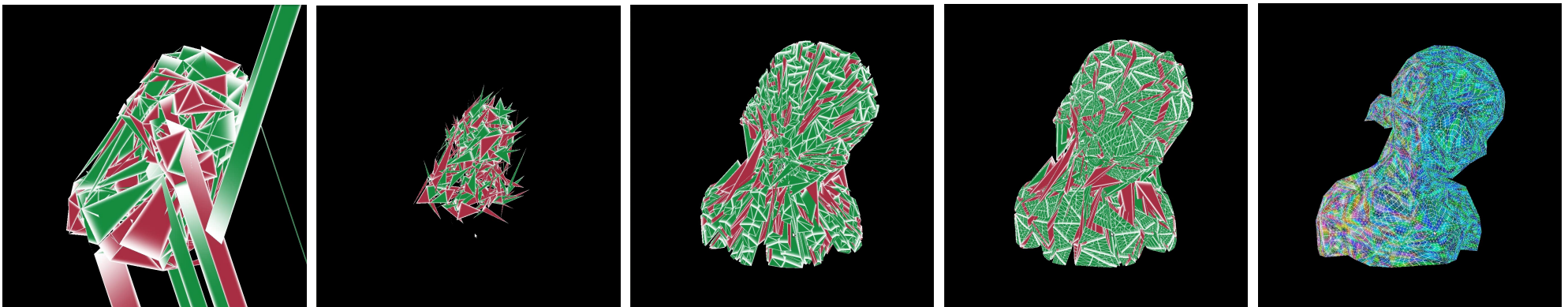
Controlled by both the **CPU** and the **GPU**

Boundary curves are still tricky

Atlas storage is naïve

More meaningful **applications**

making of ...

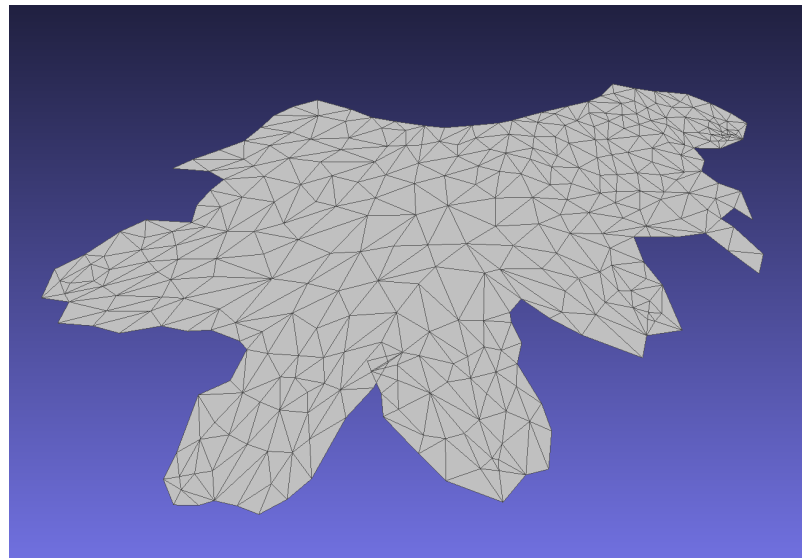
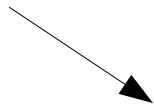
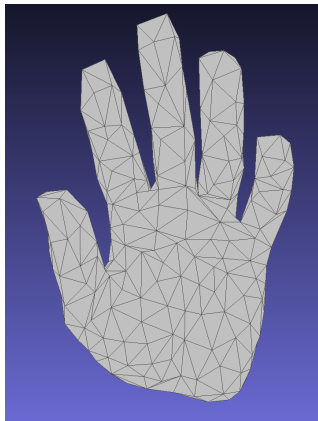


final remarks

finding boundary curves

using **global parameterization**
as initial guess

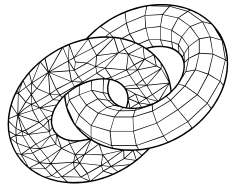
introduction
representation
final remarks



*base mesh on top of domain?
is it still a valid mesh?
on the interior?*

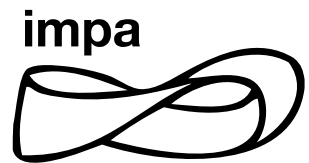
Thank You!

André Maximo



VisgrafLab

Trimester Program on
Computational Manifolds and Applications



andmax@gmail.com

<http://www.impa.br/~andmax>

acknowledgements:

