Level of Detail (LOD) Models

Part One

Outline

L O D M O D E L S

- Layered versus Multiresolution Models
- A general multiresolution surface model: The MultiTriangulation
- Basic spatial queries on multiresolution models
- Answering spatial queries at variable resolution
- Construction paradigms: an example on terrains
- Extensions to parametric surfaces and volume data

LOD Models

- **Layered models**
  - description of a sequence of few meshes each of which represents an object at a different resolution
- **Multiresolution models**
  - description of a virtually continuous set of meshes representing an object at increasing resolutions

Layered Models

- Each mesh is obtained through simplification
- Each mesh is associated with a range of levels of detail
- The range is used as a filter to select a mesh from the sequence

Standard technology in OpenInventor™ and VRML

...Layered Models...

Disadvantages of layered models:

- Each mesh is stored independently: the number of meshes must be small, otherwise the model becomes huge
- Modest possibility to adapt resolution to application needs
- Unpleasant “popping” effects during the transition between different levels
- Resolution of each mesh is uniform

Multiresolution Models

- They provide a virtually continuous range of meshes representing an object at different resolutions
- The number of different meshes, which can be extracted from the model, is not fixed a priori, but it is a function of the data size, and can be huge (e.g., combinatorial)
- Resolution of a mesh can be *variable* in different parts of the object

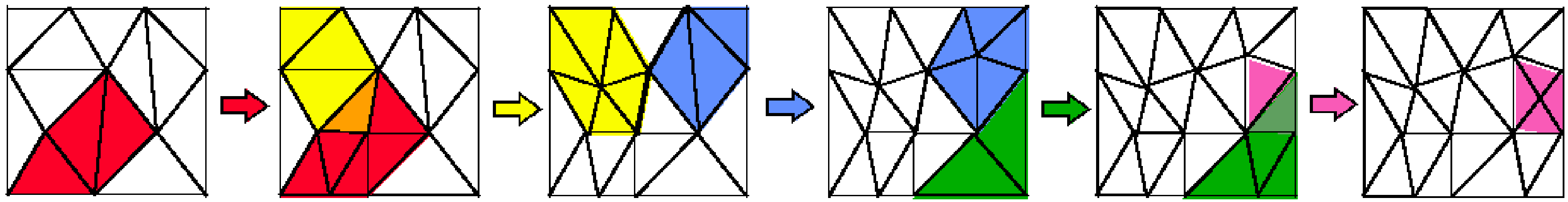
Requirements for a Multiresolution Model

- Support to efficient query processing (e.g., extraction of surface representations in real time)
- Size of the model not much higher than size of the maximum resolution representation
- No cracks or abrupt transitions within a single mesh
- Smooth transition between representations at close resolutions

L O D M O D E L S

Intuitive Idea behind Multiresolution

- Surface representations at different levels of detail (LODs) can be obtained as a sequence of local modifications on an initial mesh (by simplification or refinement)



- Some modifications depend on others
- Some modifications are mutually independent

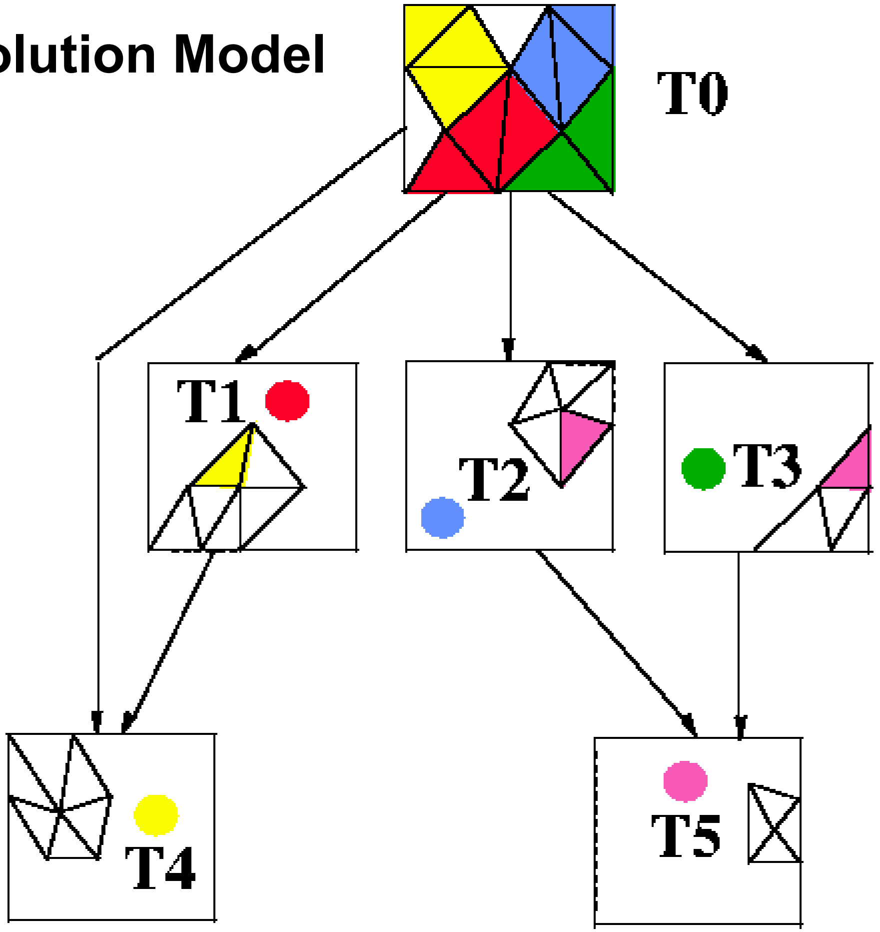
The "Heart" of a Multiresolution Model

- Initial mesh

  +

- collection of *local modifications* (triangle meshes) arranged into a *partial order*

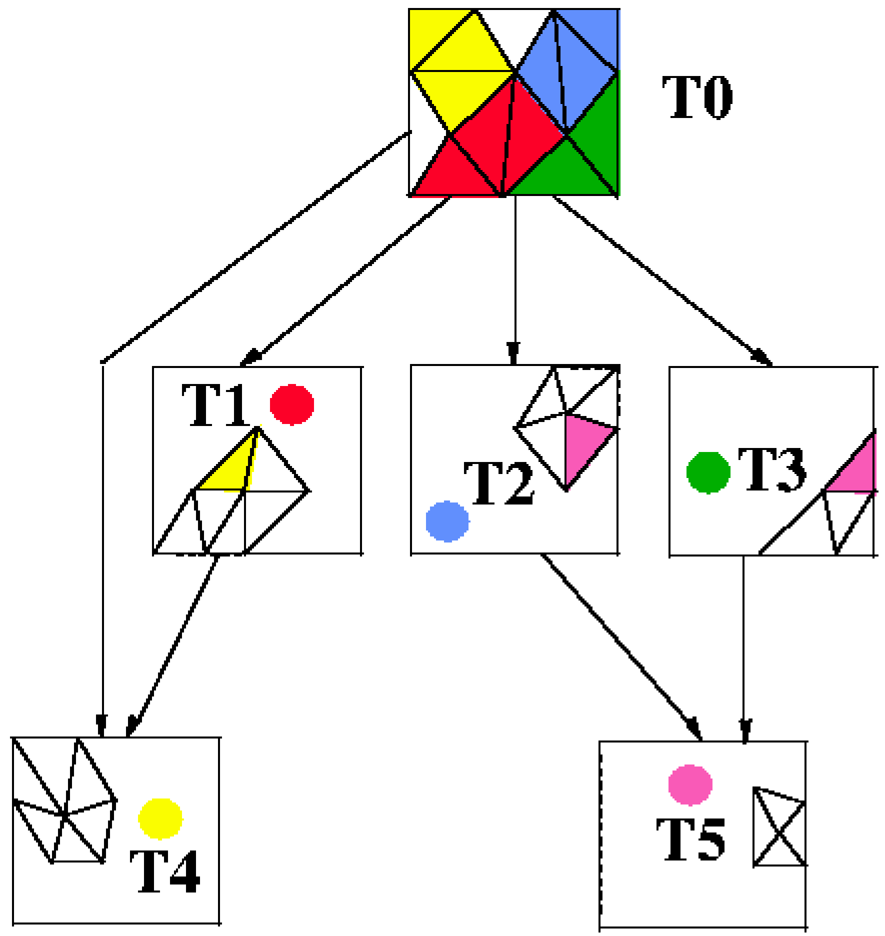
  (described by a DAG)



the MultiTriangulation

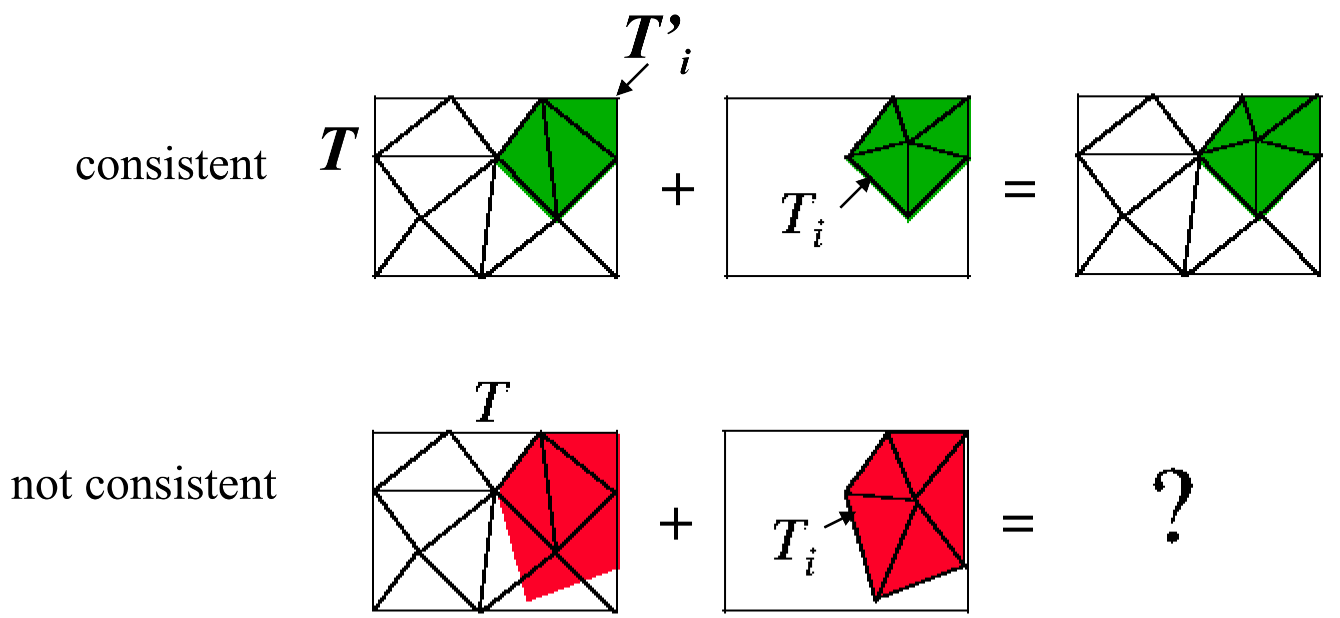
L O D M O D E L S

- A *MultiTriangulation* (MT) is a labeled DAG where
  - nodes are triangle meshes
  - arcs describe the partial order



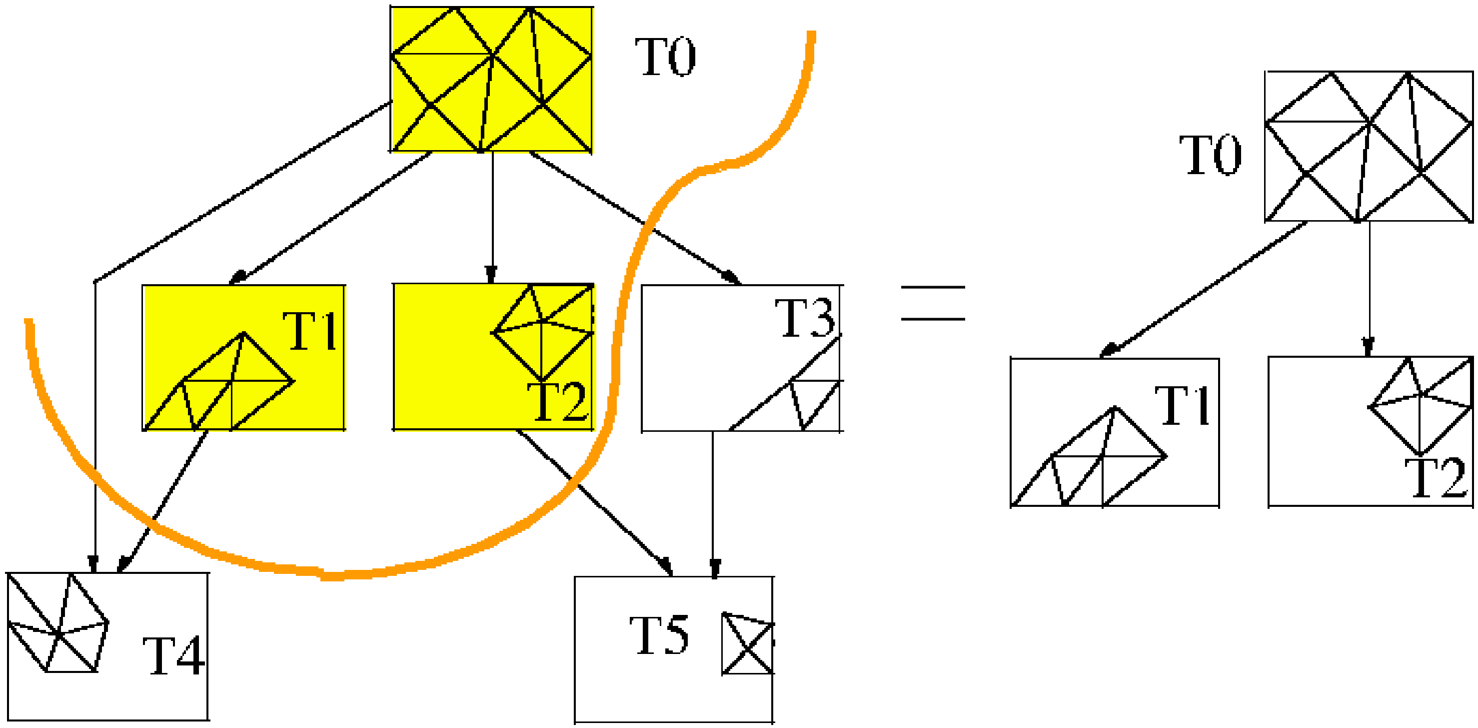
...A General Framework for Multiresolution...

- Each local modification must be consistent
- *Consistent* modification of a triangle mesh:
  - If T is a triangle mesh, a mesh T_i is a *consistent modification* of T iff T contains a submesh T'_i, such that T_i "covers" T'_i, and T_i has more triangles than T'_i
  - T'_i is called the *floor* of T_i,



Expressive Power of a MultiTriangulation

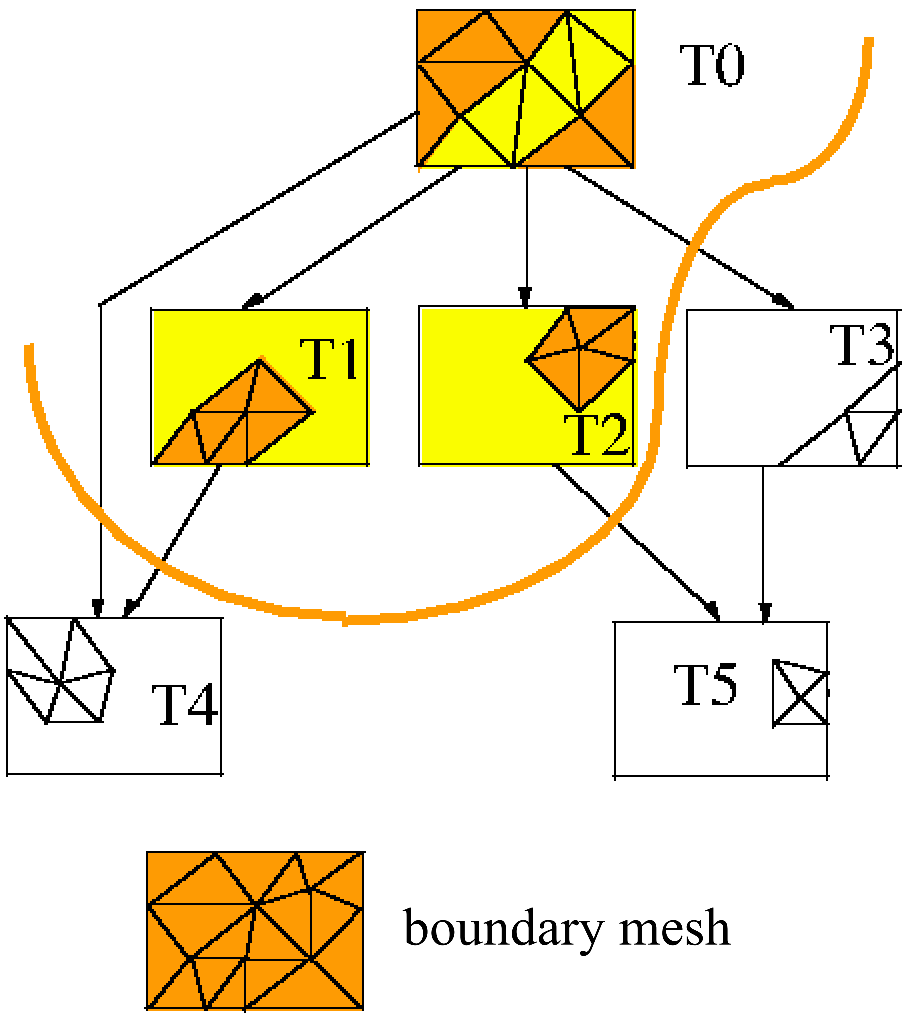
- A subMT of an MT ***M*** is a subgraph ***M'*** where
  - ***M'*** contains the root
  - If T_i belongs to ***M'***, then all parents of T_i belong to ***M'*** as well
- Every subMT is an MT



L O D M O D E L S

...Expressive Power of a MultiTriangulation...

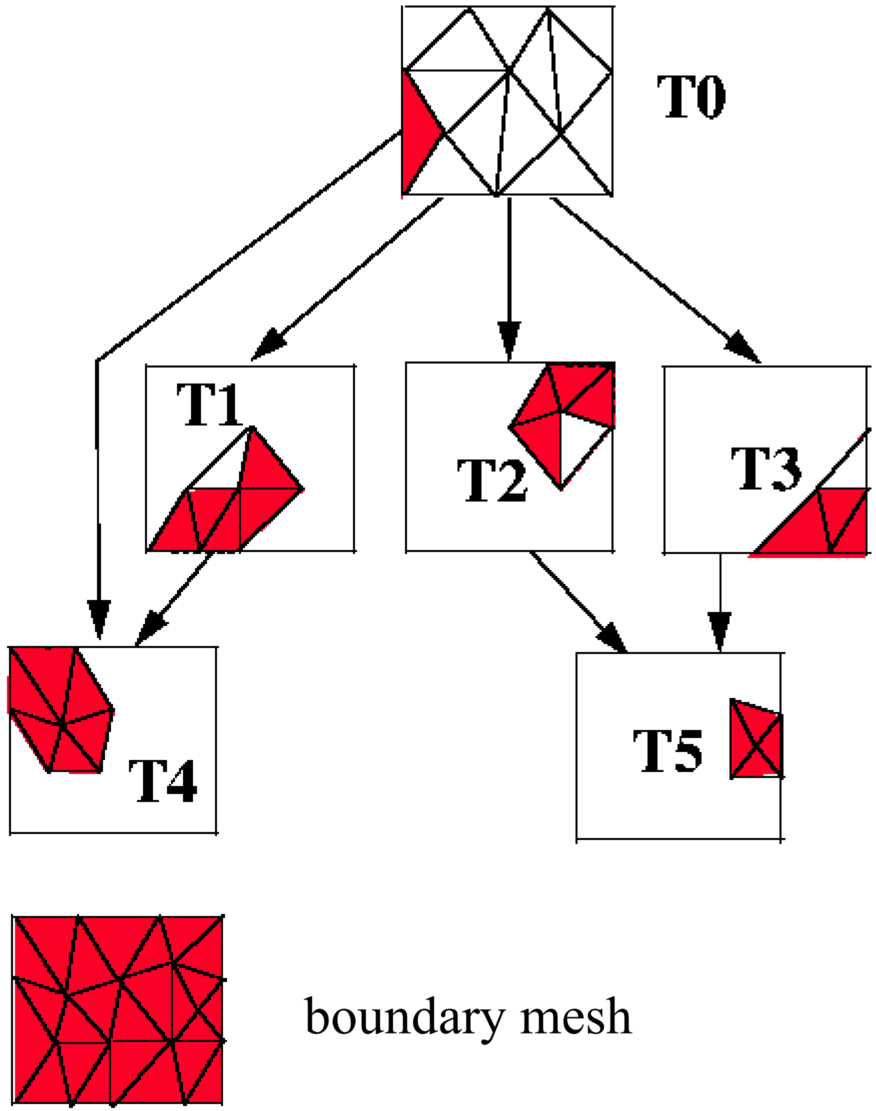
- Any mesh made of triangles in ***M*** is the boundary mesh of a subMT
- **Boundary mesh:** mesh obtained by applying all modifications in the subMT to the root mesh



...Expressive Power of a MultiTriangulation...

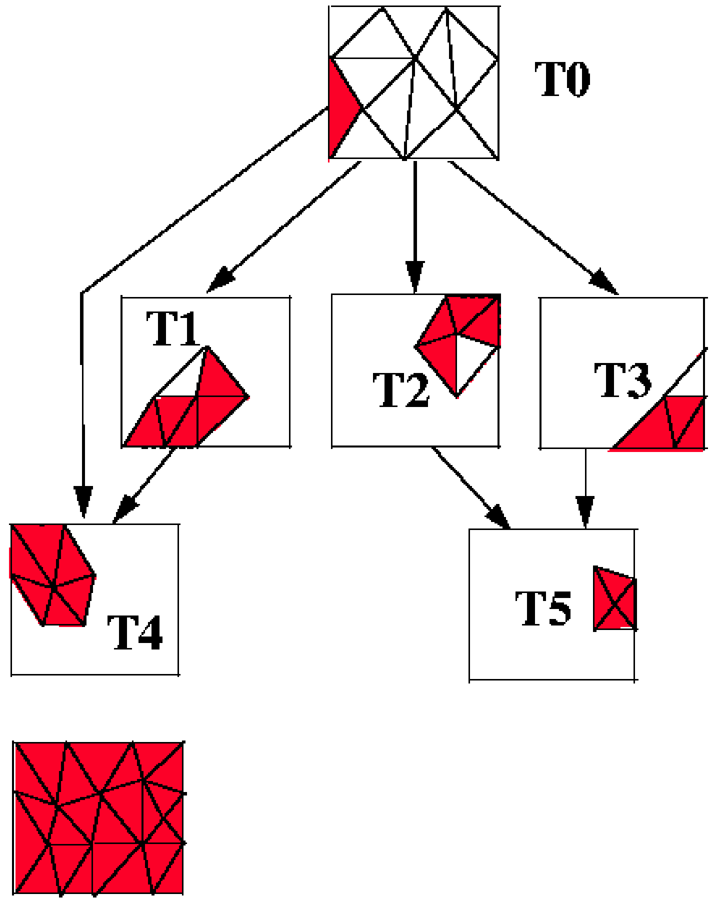
Mesh at Maximum Resolution

- boundary mesh associated with the MT itself



Desirable Properties for a MultiTriangulation

- Linear growth:
  - the number of triangles of the MT is linear in the number of triangles in its boundary mesh



...Desirable Properties for a MultiTriangulation...

- Bounded width:
  - the number of triangles in any MT mesh is bounded from above by a constant
- Logarithmic height:
  - the maximum path length is logarithmic in the total number of arcs of the MT

Remark: bounded width ==> linear growth

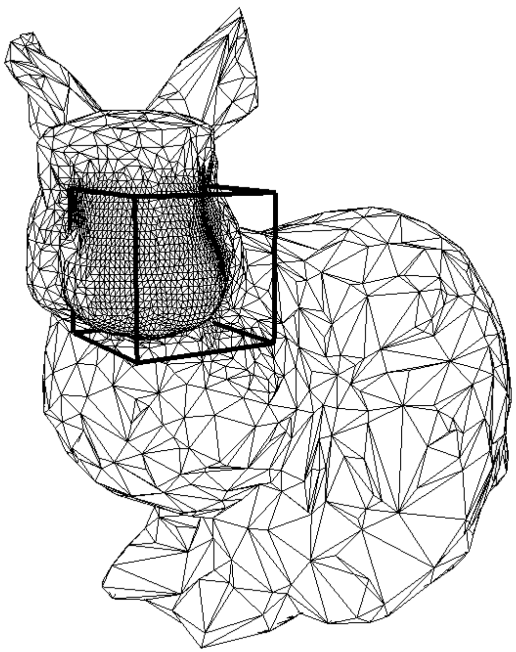
Spatial Queries on an MT

Special cases of a *general extraction query* specified by:

- an *accuracy condition*:
  - specification of the LOD at which the mesh is queried
  - threshold function bounding the distance between the original surface and the mesh extracted from the MT
- a *focus condition*:
  - specification of the type of geometric operation defined by the query
  - focus set defining the area of interest of the query

Example:

maximum resolution inside a box



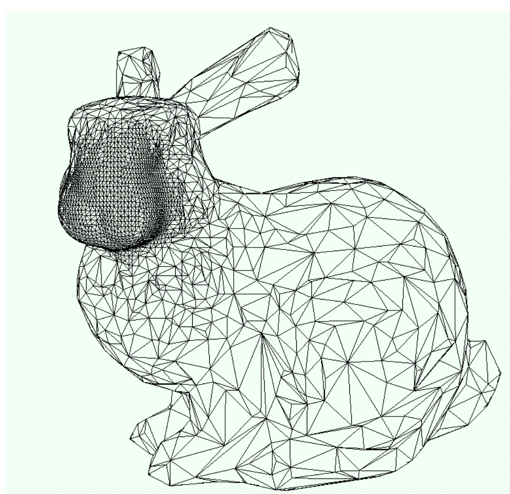
...Spatial Queries on an MT...

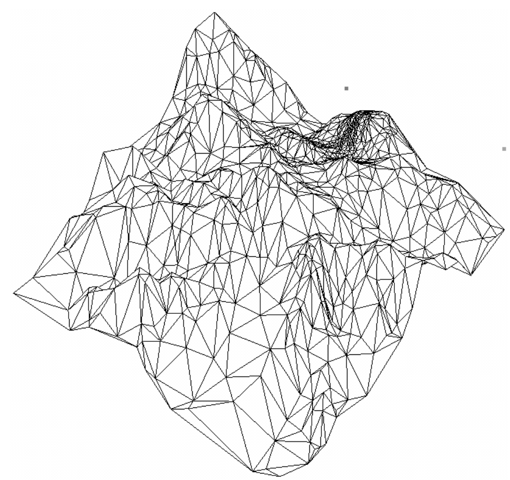
Accuracy Condition

- *Threshold function* $\tau : R^3 \rightarrow R$
- A triangle ***t*** is called *valid* iff its approximation error is lower than the minimum value of the threshold over ***t***
- A triangle mesh satisfies τ if all its triangles are valid

Examples of threshold functions:

- for *arbitrary surfaces*: increasing with the distance from the viewpoint, measured in 3D space
- for *terrains*: increasing with the distance from the viewpoint, measured on the x-y plane





...Spatial Queries on an MT...

Focus Condition

- *Focus set **F in R**^{3}*
- A triangle ***t*** is called *active* iff

  $t \cap F$ is not empty

- A focus set describes the region of interest of the query

  Examples of focus sets:
  - *Point* : point location query
  - *Line/polyline* : segment/line interference query
  - *Region* : window query, region interference query
  - *Volume*: view frustum

...Spatial Queries on an MT...

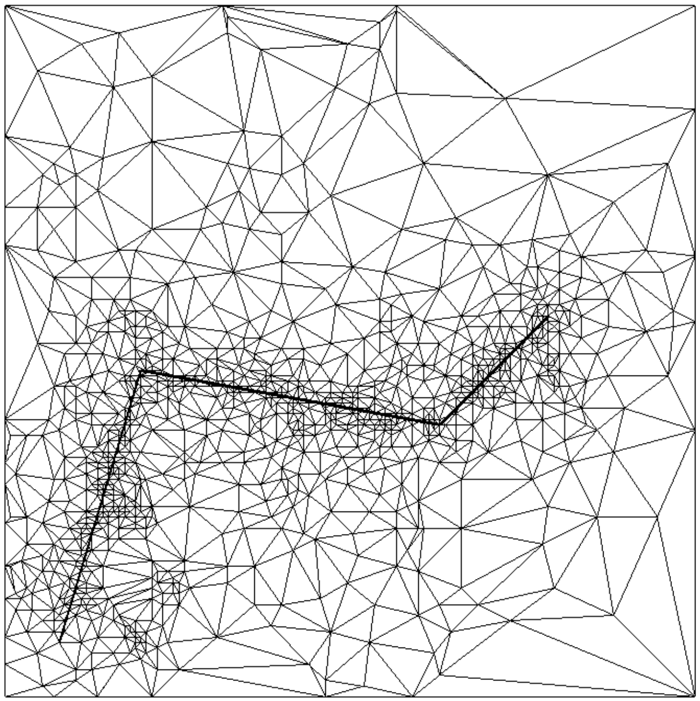
General Extraction Query (called **Selective Refinement**)

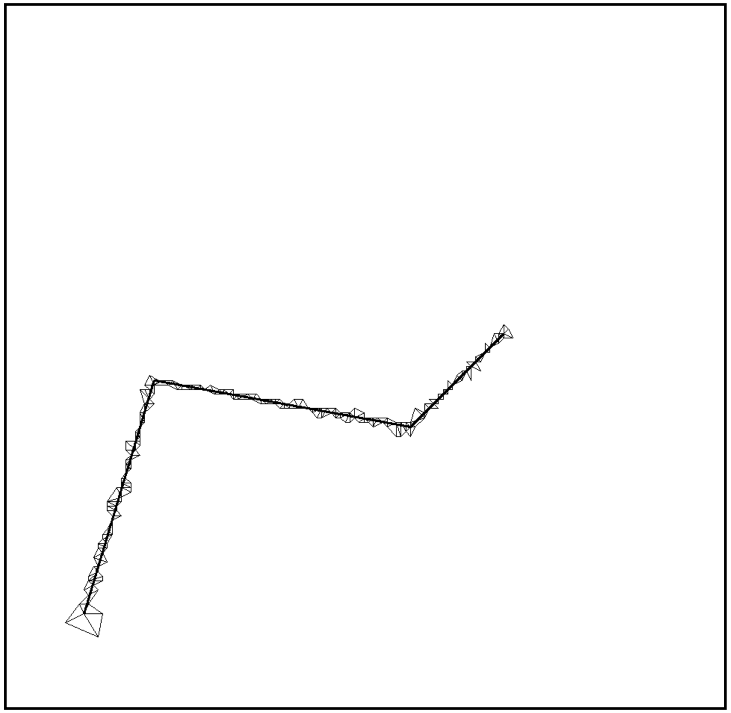
- Triangle mesh ***T***, among all meshes described by the MT, such that
  - ***T*** has minimal size (minimal number of triangles)
  - all active triangles of ***T*** are valid

...Spatial Queries on an MT...

Two instances of the General Extraction Query

- Resulting mesh *globally defined* :
  - defined on the whole surface
- Resulting mesh *locally defined* :
  - defined only on the area of interest





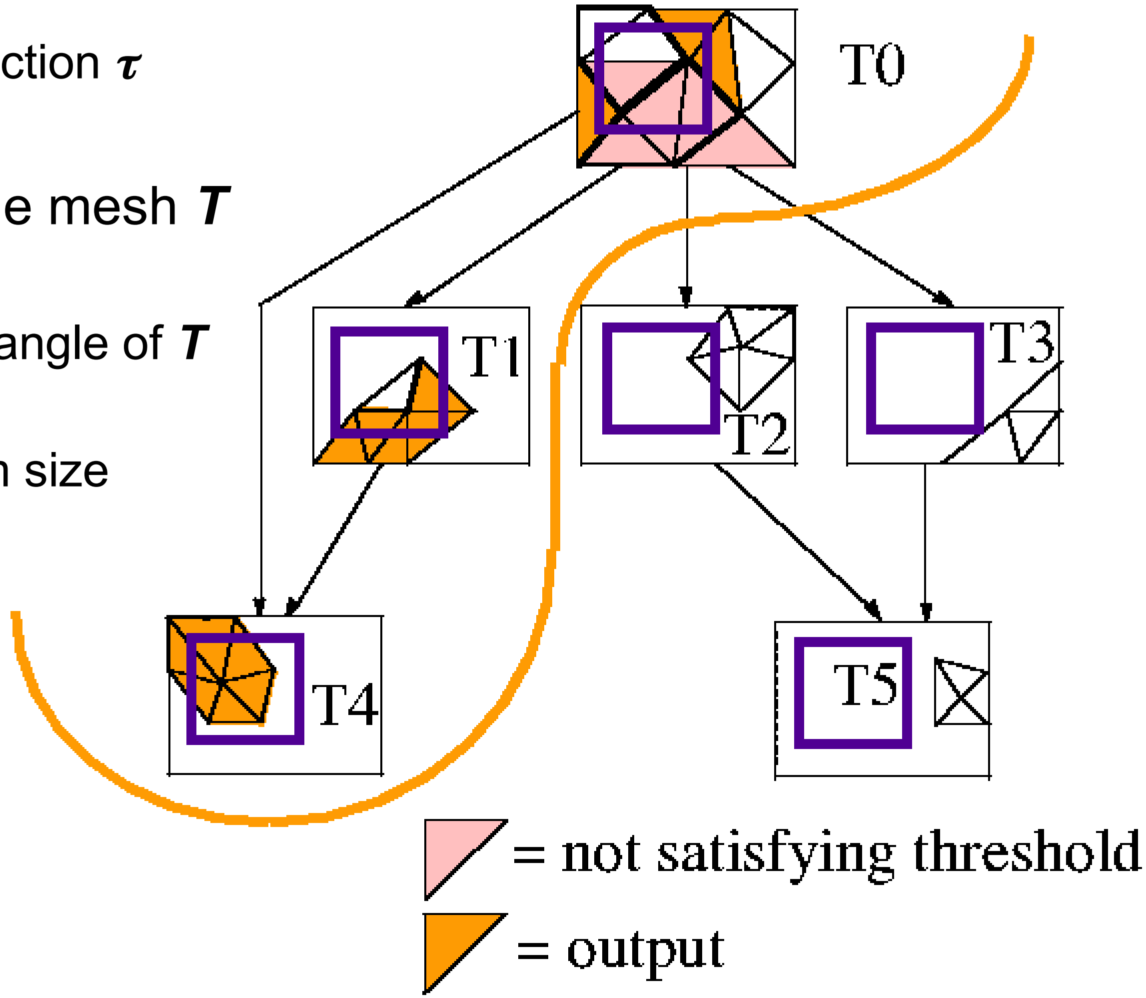
...Spatial Queries on an MT...

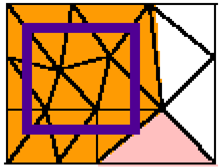
Extraction Queries

- Extraction of a mesh *from scratch* :
  - *static extraction query*
- Extraction of a mesh by *updating* a previously extracted one :
  - *dynamic extraction query*

Globally Defined Static Extraction Query

- Given
  - a threshold function τ
  - a focus set ***F***
- Retrieve a triangle mesh ***T*** such that
  - every active triangle of ***T*** is valid
  - ***T*** has minimum size



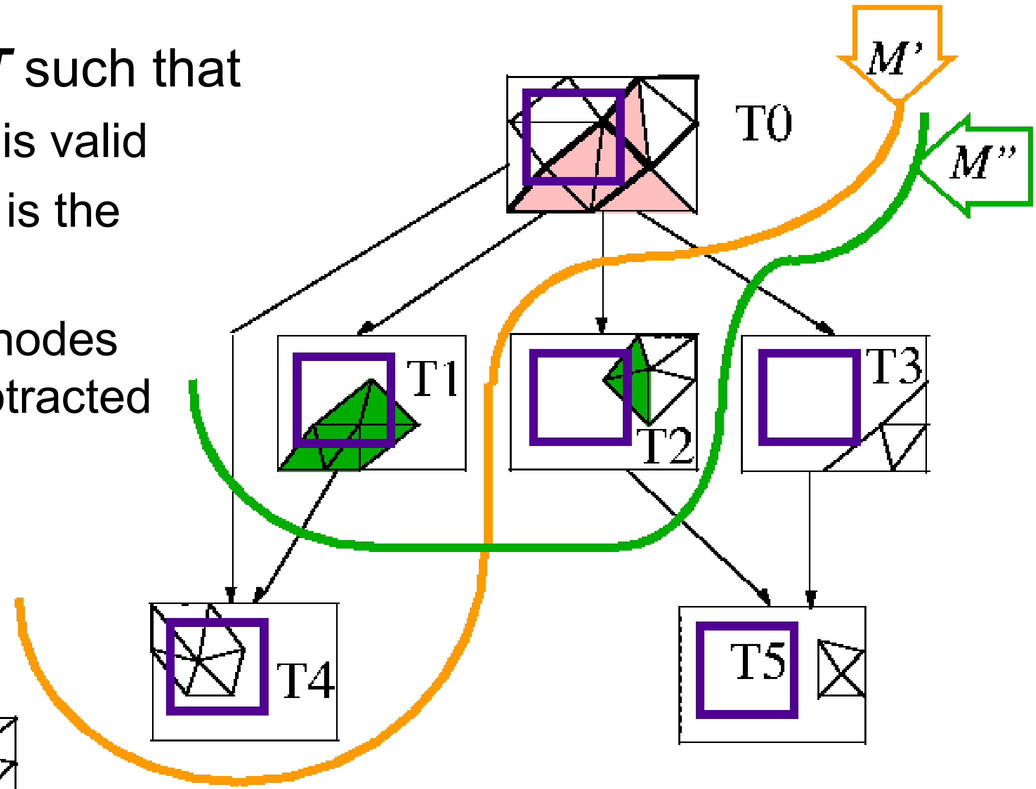


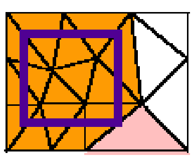
L O D M O D E L S

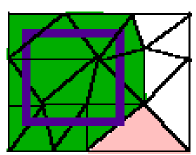
Globally Defined Dynamic Extraction Query

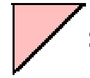
- Given
  - a threshold function τ , a focus set ***F***
  - a subMT ***M'***
- Retrieve a triangle mesh ***T*** such that
  - every active triangle of ***T*** is valid
  - the subMT ***M"*** defining ***T*** is the closest to ***M'***

where distance = number of nodes which must be added to / subtracted from ***M'*** to obtain ***M"***




previous mesh


current mesh


= not satisfying threshold

Algorithms for Extracting Meshes at Variable Resolution

- Algorithms for globally defined queries:
  - an algorithm for answering the static mesh extraction query
  - an algorithm for answering the dynamic mesh extraction query
- For an algorithm for the locally defined query in the static case, see (De Floriani et al., IEEE Visualization'98)

Static Extraction Algorithm (De Floriani, Magillo, Puppo, 1997)

L O D M O D E L S

- Breadth-first traversal of the MT
  - A *current subMT* is maintained during traversal
  - The *current mesh* is the boundary mesh of the current subMT
- Initially, the current subMT contains just the root
- If some active triangle ***t*** of the current mesh is not valid, then
  - get the MT node T_i refining ***t***
  - recursively add to the current subMT all parents of T_i
  - add T_i to the current subMT
- Repeat until either all active triangles are valid (the desired accuracy is achieved) or time is expired

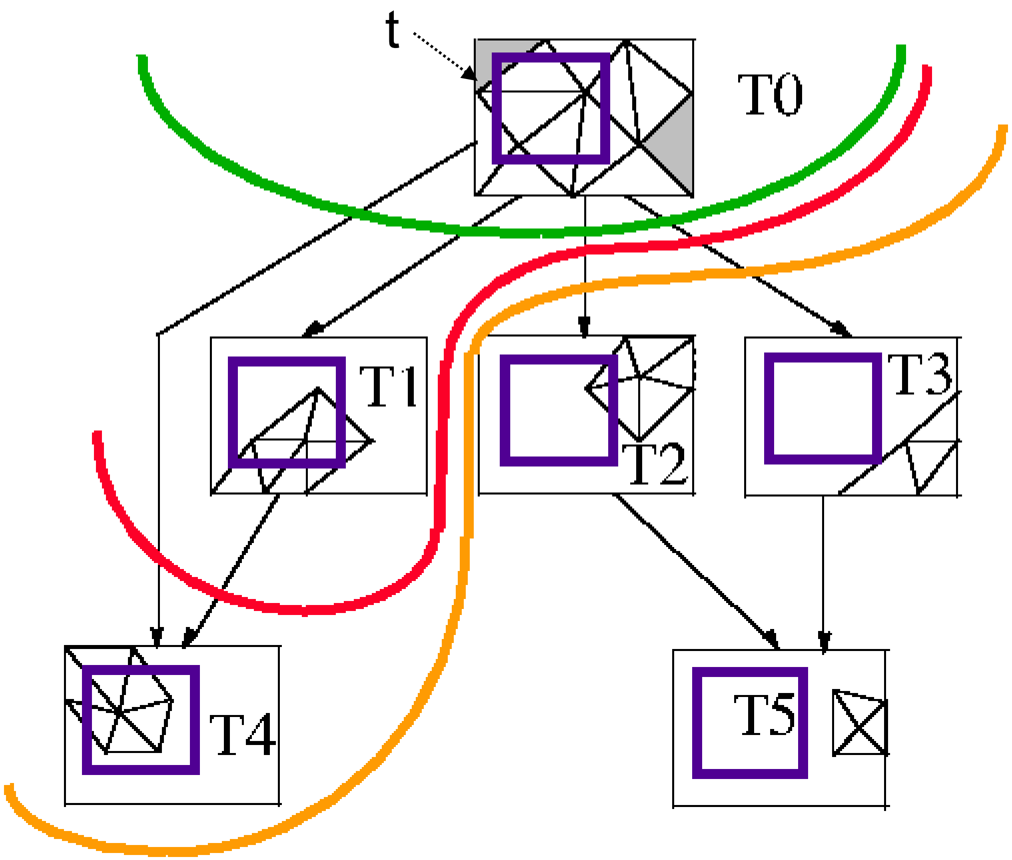
...Static Extraction Algorithm...

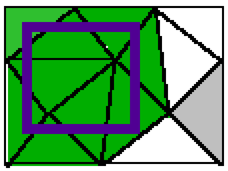
LOD MODELS

Grey triangles are not valid

Focus set is a box

- Initial situation: green
- Triangle ***t*** is active and not valid
  - ***t*** is refined by ***T4***
  - ⇒ must add first ***T1***, then ***T4***

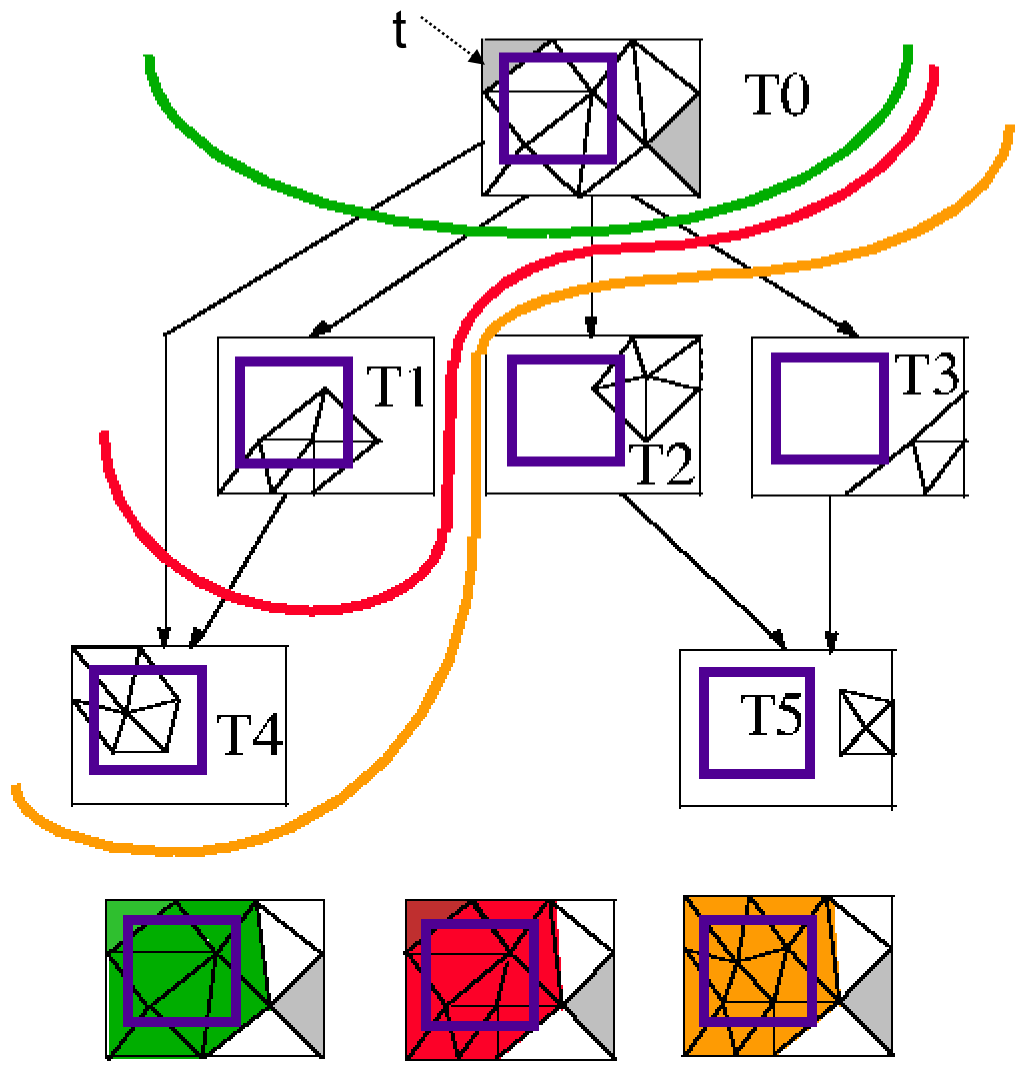




...Static Extraction Algorithm...

L O D   M O D E L S

- Add ***T1*** to subMT
  - ⇒ red
- Add ***T4*** to subMT
  - ⇒ orange
- All active triangles are valid
  - ⇒ stop



...Static Extraction Algorithm...

- Variable resolution with arbitrary threshold supported
- Interruptibility:
  - it converges to the exact solution by producing better and better approximations
- Correctness:
  - set of output triangles forms a boundary triangulation of a subMT
  - any boundary triangulation of smaller size does not satisfy the threshold
- Time complexity:
  - linear in the number of visited triangles
  - linear in the output size, if the MT has a linear growth

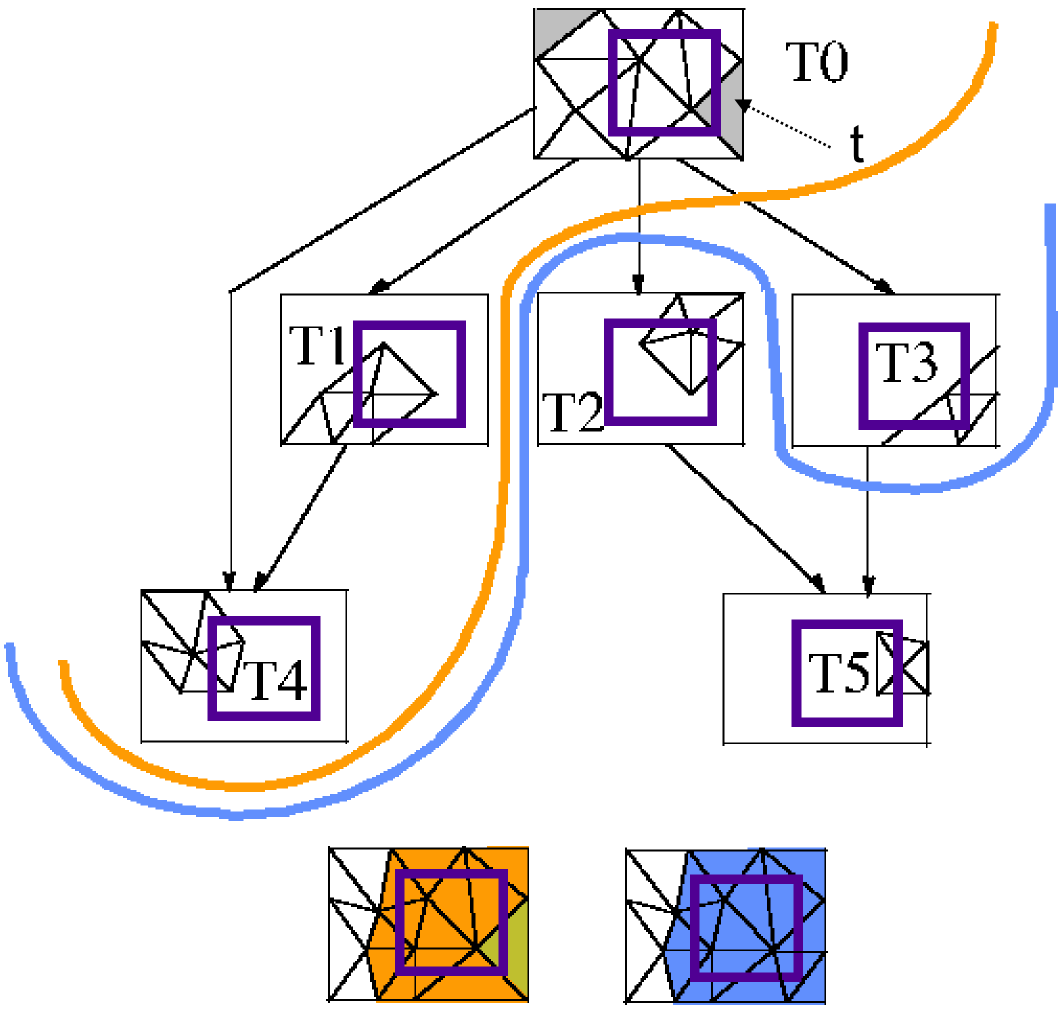
Dynamic Extraction Algorithm (De Floriani, Magillo, Puppo, 1998)

Two basic steps:

- *Expansion step*:
  - refine the current mesh until all active triangles are valid
- *Contraction step*:
  - coarsen the current mesh until it cannot be further coarsened without getting some active triangle which is not valid

- Expansion *adds nodes* to the current subMT
  - proceed as in the static case
- Contraction *removes nodes* from the current subMT
  - check all the nodes which are leaves of the current subMT
  - if a leaf node T_i can be removed without getting some invalid active triangle, then remove T_i and update the current mesh

...Dynamic Extraction Algorithm...



Expansion

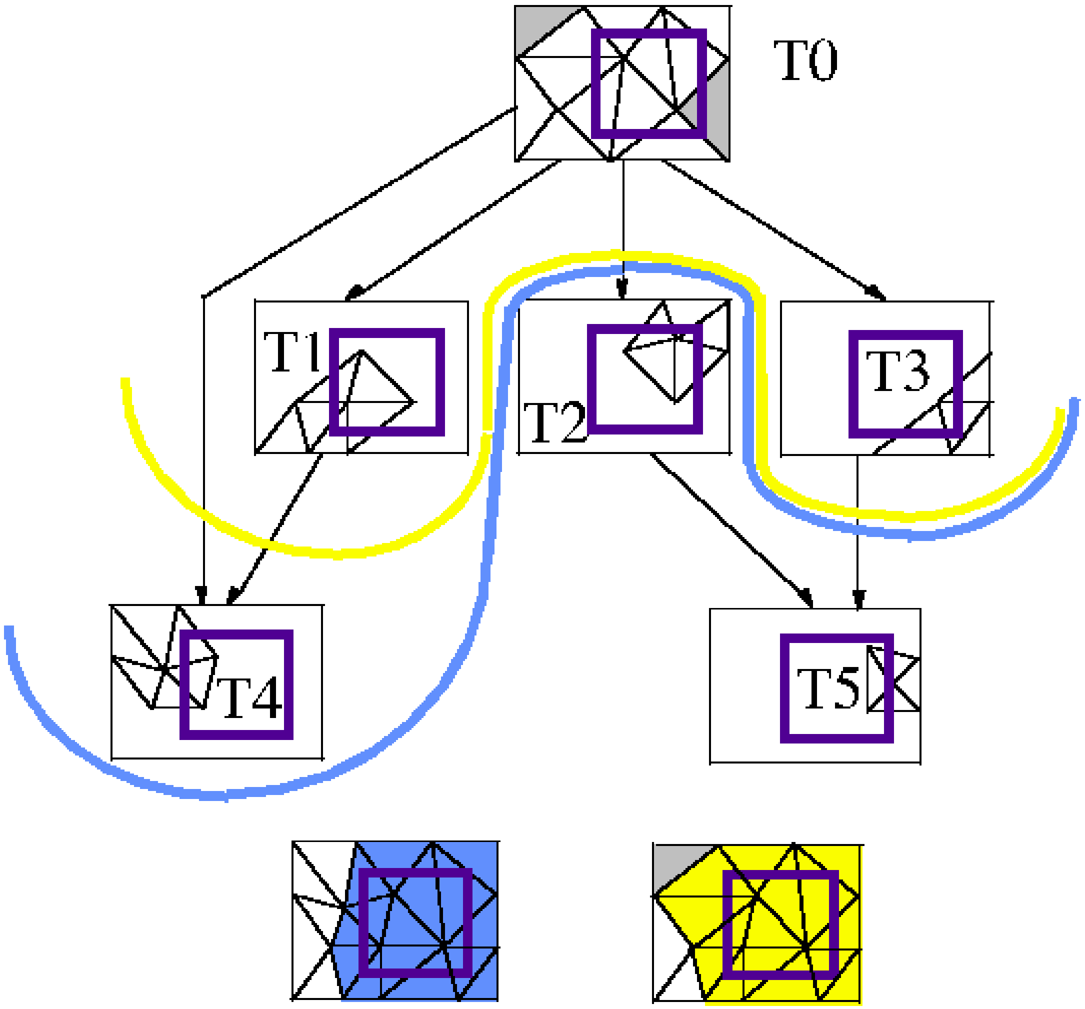
- Initial situation: orange
- Triangle t is active and not valid
  - ==> must add T3
- Add T3 to subMT
  - ==> blue
- all active triangles are valid
  - ==> stop expansion

...Dynamic Extraction Algorithm...

L O D M O D E L S

Contraction

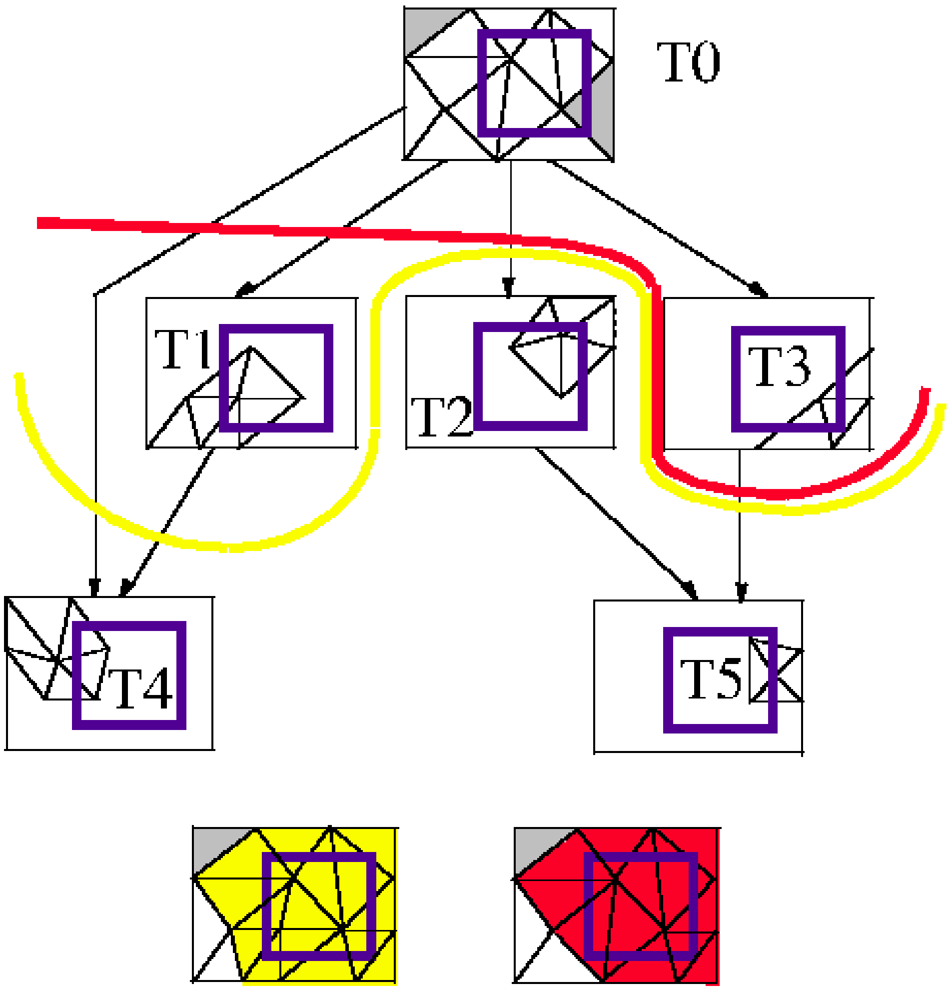
- initial situation: blue
  - leaves: ***T4***, ***T3***
- examine leaf ***T4***
  - ==> remove ***T4***
  - ==> yellow
  - ==> ***T1*** becomes a leaf



...Dynamic Extraction Algorithm...

Contraction continues

- Examine leaf ***T3***
  - do not remove ***T3***
  - ==> still yellow
- examine leaf T1
  - remove T1
  - ==> red
- no more leaves
  - ==> stop



...Dynamic Extraction Algorithm...

- Correctness:
  - set of output triangles forms a boundary mesh of a subMT
  - removing a node from the final subMT makes the boundary triangulation violate the threshold
- Time complexity:
  - linear in the number of visited triangles
  - linear in the sum of the input size and of the output size, if the MT has a linear growth

Experimental Comparison of Static and Dynamic Approaches

- Terrain dataset (maximum resolution: 32,250 triangles)
- Threshold increases with distance from a moving viewpoint
- Focus set is a view frustum

LOD MODELS

iter.	output triangles	swept triangles		time (msecs)	
		static	**dyna.**	**static**	**dyna.**
5	2681	11663	326	53.77	36.07
15	2734	11900	38	56.51	36.03
25	2736	11902	260	55.24	35.68
35	2758	11996	44	55.31	35.48
45	2792	12114	134	54.47	35.38
55	2640	11450	14	58.17	36.25
65	2688	11688	242	56.28	36.37
75	2768	12014	266	55.69	35.43
85	2710	11770	207	53.68	34.85
95	2682	11644	90	54.04	34.69

...Experimental Comparison...

- Bunny dataset (maximum resolution: 69,451 triangles)
- Threshold is zero, focus set is a moving box

iter.	output triangles	swept triangles		time (msecs)	
		static	**dyna.**	**static**	**dyna.**
5	2230	4166	62	39.31	25.76
15	2426	5094	158	42.52	27.77
25	2572	5776	0	43.12	28.29
35	2690	6354	50	44.58	28.90
45	2749	6713	268	45.33	30.25
55	2745	6667	134	44.85	29.24
65	2602	5858	624	41.36	29.20
75	2688	6312	56	43.34	28.20
85	2690	6372	20	43.50	28.20
95	2584	5836	270	40.94	27.94

How to Construct a "good" MT

L
O
D

M
O
D
E
L
S

- Shape of the MT versus construction strategy
- An MT can be built from a sequence of local modifications on an initial (coarse or fine) mesh : *construction sequence*
- A construction sequence is generated through a mesh refinement or mesh simplification process
- See (De Floriani, Magillo and Puppo, IEEE Visualization'97) for algorithms to build an MT from a construction sequence

...How to Construct a “good” MT...

Requirements for a Construction Algorithm

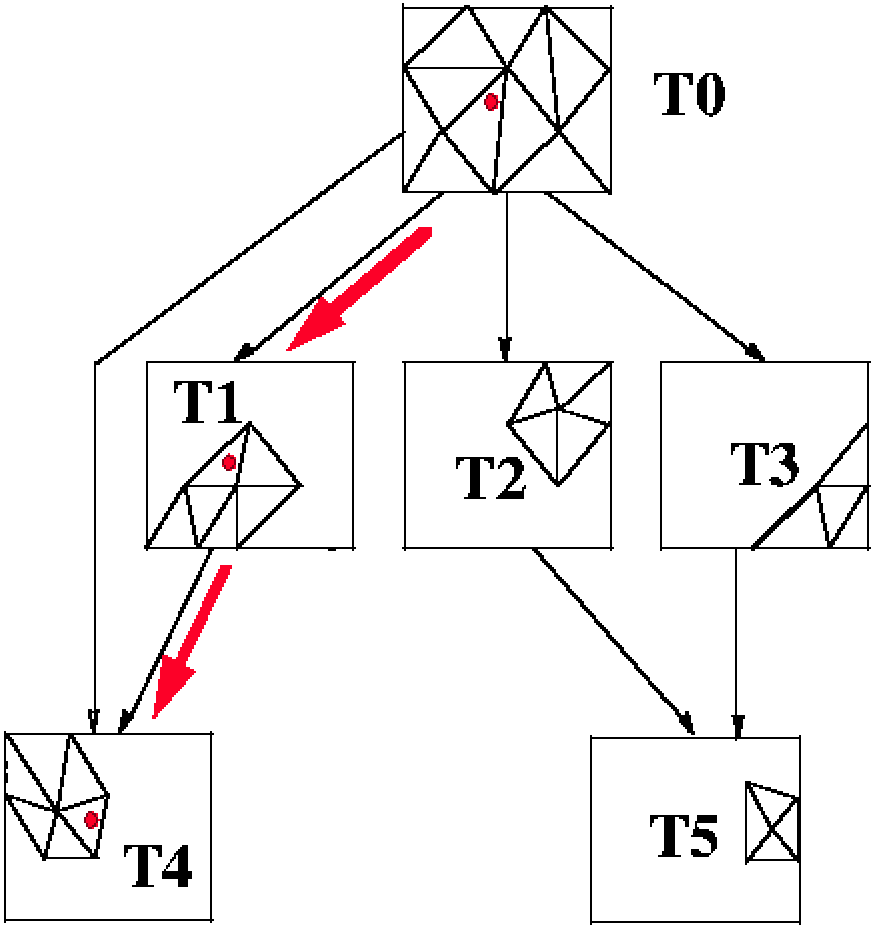
- *Good compression ratio*: reduced size of any extracted mesh
- *Linear growth*: small overhead factor
- *Bounded width*
- *Logarithmic height*

...How to Construct a “good” MT...

Why Such Requirements?

An example:
point location on an MT at variable resolution

Cost depends on:
- width
- height
- size of the visited subMT



L O D M O D E L S

...How to Construct a “good” MT...

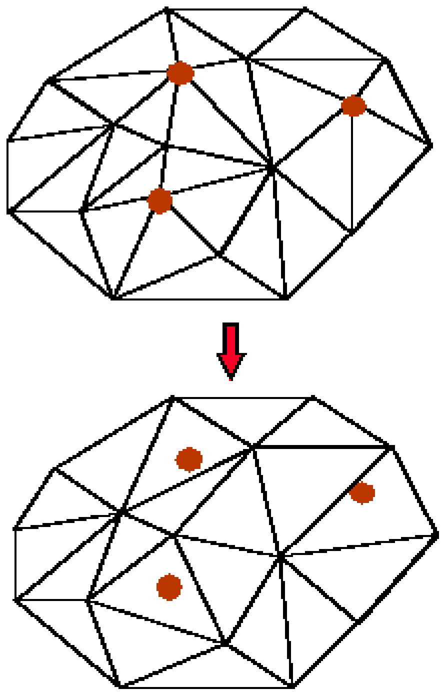
Evaluation of Construction Strategies

- Theoretical and experimental evaluation of the shape of the MT based on the algorithm used for generating the construction sequence
- We have performed such evaluation in the case of terrains considering four different variants of the vertex removal strategy

...How to Construct a "good" MT...

- **Method 1**:
  - remove an arbitrary maximal set of independent vertices of bounded degree
- **Method 2**:
  - as method 1, but always starting from the vertex causing the smallest error increase

- Methods 1 and 2 guarantee:
  - linear growth
  - bounded width
  - logarithmic heigth

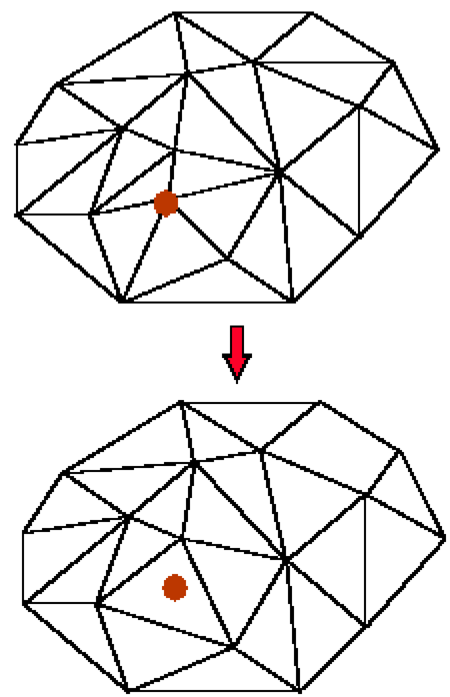


Error-driven selection of method 2 maintains "important" vertices close to the root

...How to Construct a “good” MT...

- **Method 3**:
  - remove the vertex with bounded degree causing the smallest error increase
- **Method 4**:
  - remove the vertex causing the smallest error increase

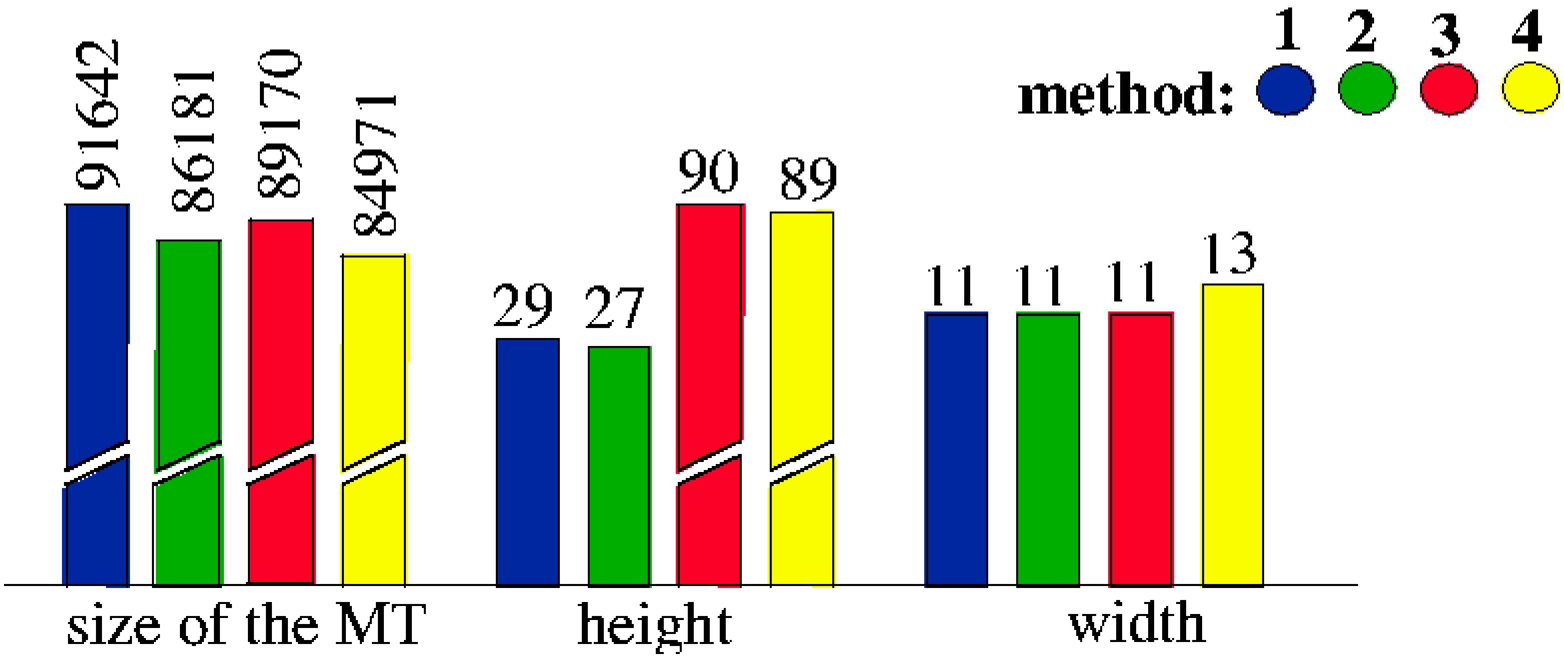
- Method 3 guarantees:
  - linear growth
  - bounded width
  - no logarithmic height
- Method 4 guarantees:
  - none of the three properties



...How to Construct a "good" MT...

Experimental evaluation of the four methods

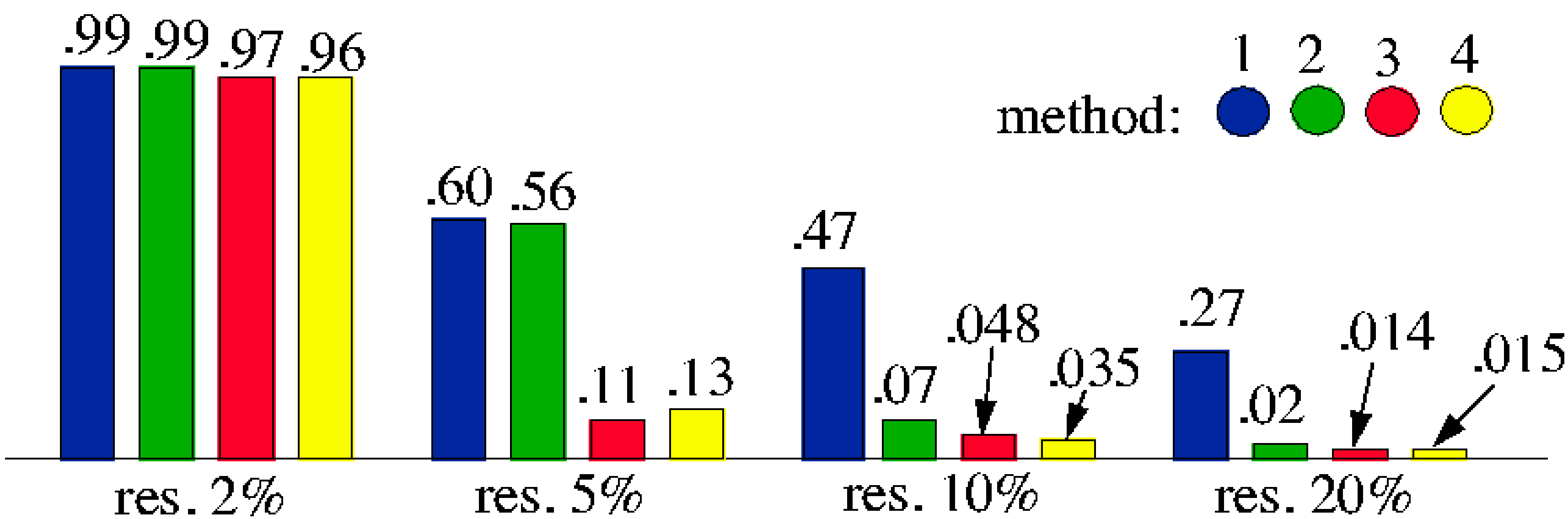
- Data set: 128 x 128 grid of elevation data from US Geological Survey
- Size of the triangulation at maximum resolution: 32,258 triangles



...How to Construct a “good” MT...

Compression Factor

- Evaluation based on extraction of meshes at different LODs
- Ratio between the size of the output and the size of the mesh at maximum resolution



MT for Modeling Free-Form Surfaces

- An MT can be built:
  - *from an initial refined triangle mesh* through a simplification approach (we use the vertex decimation algorithm by Cignoni, Montani and Scopigno, 1997)
  - *from scattered data* through a sculpturing approach (Boissonnat, 1994; Veltkamp, 1993; Bajaj et al., 1996, De Floriani et at., ICPR'98)
  - *from contours*: initial mesh built by connecting contours lying on adjacent planes (Fuch et al., 1977; Boissonnat, 1988; Geiger, 1993)
  - *from a parametric surface description*: data are a collection of adjacent patches; refinement approach applied to the boundary curves and then to the interior of each patch (De Floriani, Magillo, Puppo, ICIAP'97)

More on the MT

Implementation of the MT as a Library

- Independent of how the construction sequence is generated
- Library written in C++, tested on both SGI and PC platforms
- It implements:
  - several extraction algorithms as well as several internal encoding structures
  - a collection of threshold functions and focus sets
  - algorithms for building an MT from a given construction sequence

Extension to Volume Data Representation

- MT definition is independent of the dimension of the space
- The definition of MT for tetrahedral meshes directly extends the one for triangle meshes
- We are currently developing a library for 3D MT (in collaboration with Cignoni, Montani, and Scopigno)

References

L O D M O D E L S

- *General definitions and properties*: Puppo, CCCG'96 / Computational Geometry, 1998; Magillo, PhD Thesis, 1999
- *Multidimensional extension and other models in the MT framework*: De Floriani, Magillo, Puppo, DGCI'99
- *Construction of an MT from triangle meshes*: De Floriani, Magillo, Puppo, IEEE VIS'97
- *Construction of an MT from parametric surfaces*: De Floriani, Magillo, Puppo, ICIAP'98
- *Construction of an MT from scattered points:* De Floriani, Magillo, Puppo, ICPR'98
- *VARIANT* (a terrain modeling system): De Floriani, Magillo, Puppo, ACM GIS 1997
- *Data structures and extraction algorithms*: De Floriani, Magillo, Puppo, IEEE VIS'98

Our web page:

http://www.disi.unige.it/research/Geometric_modeling/