Segmentation for 3D building generalisation

Frank Thiemann

Monika Sester

Institute of cartography and geo-informatics

University of Hanover

Content

Algorithm for Segmentation

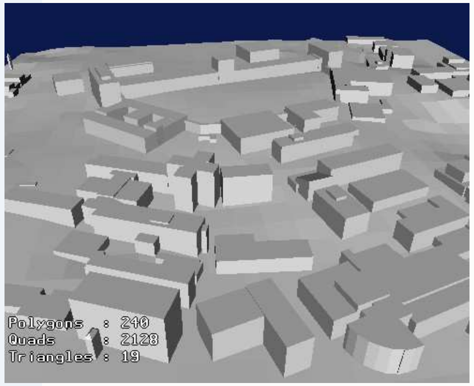
Analysis and Generalisation

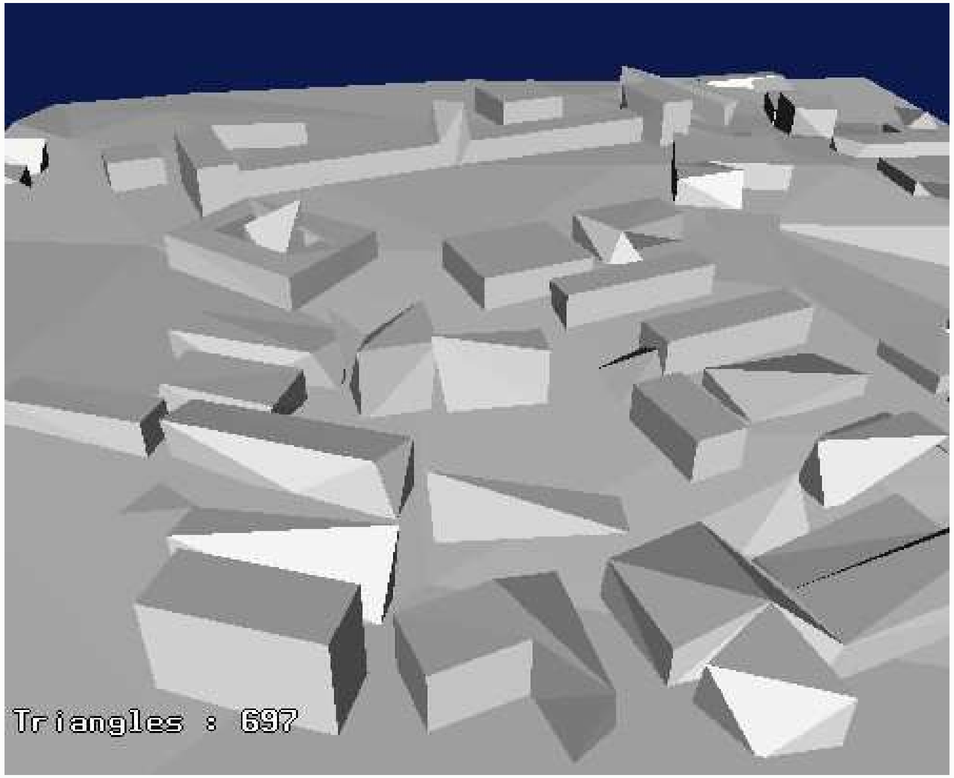
- Geometric classification of features
- Additional common sense knowledge

Conclusions

Introduction

- elimination of single vertices is not suited for 3D building generalisation





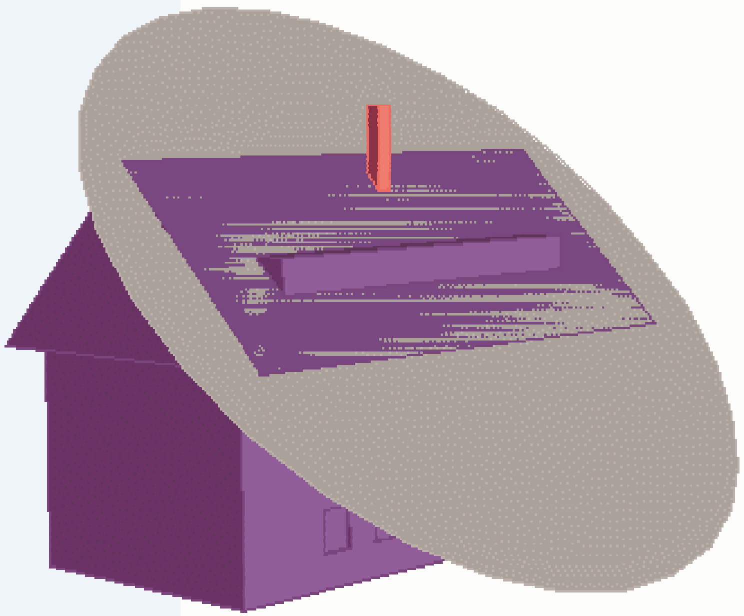
- examine parts
  – size
  – semantic / significance

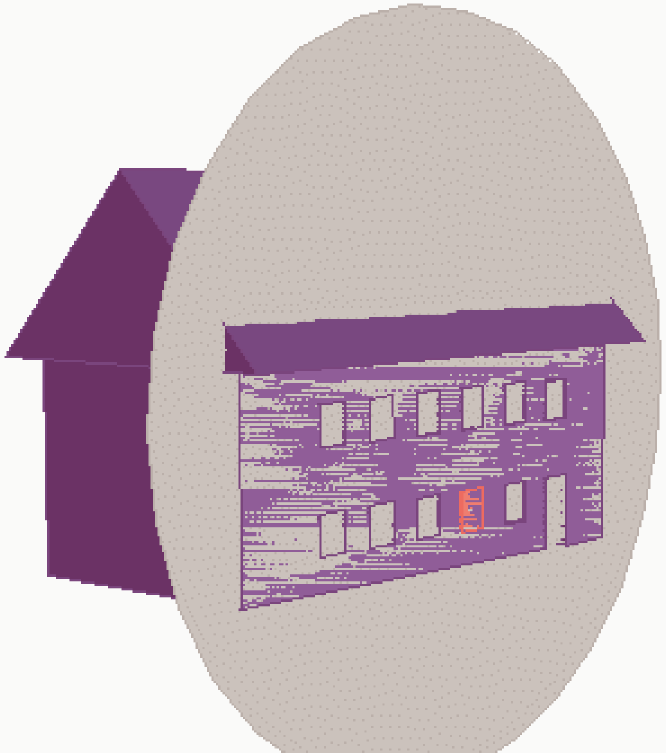
→ Decompose the complex object in smaller “meaningful” parts to get small objects which can be separately handled.

Segmentation

- using the algorithm of Ribelles, Heckbert ... to decompose a polyhedron in meaningful parts (so called features)

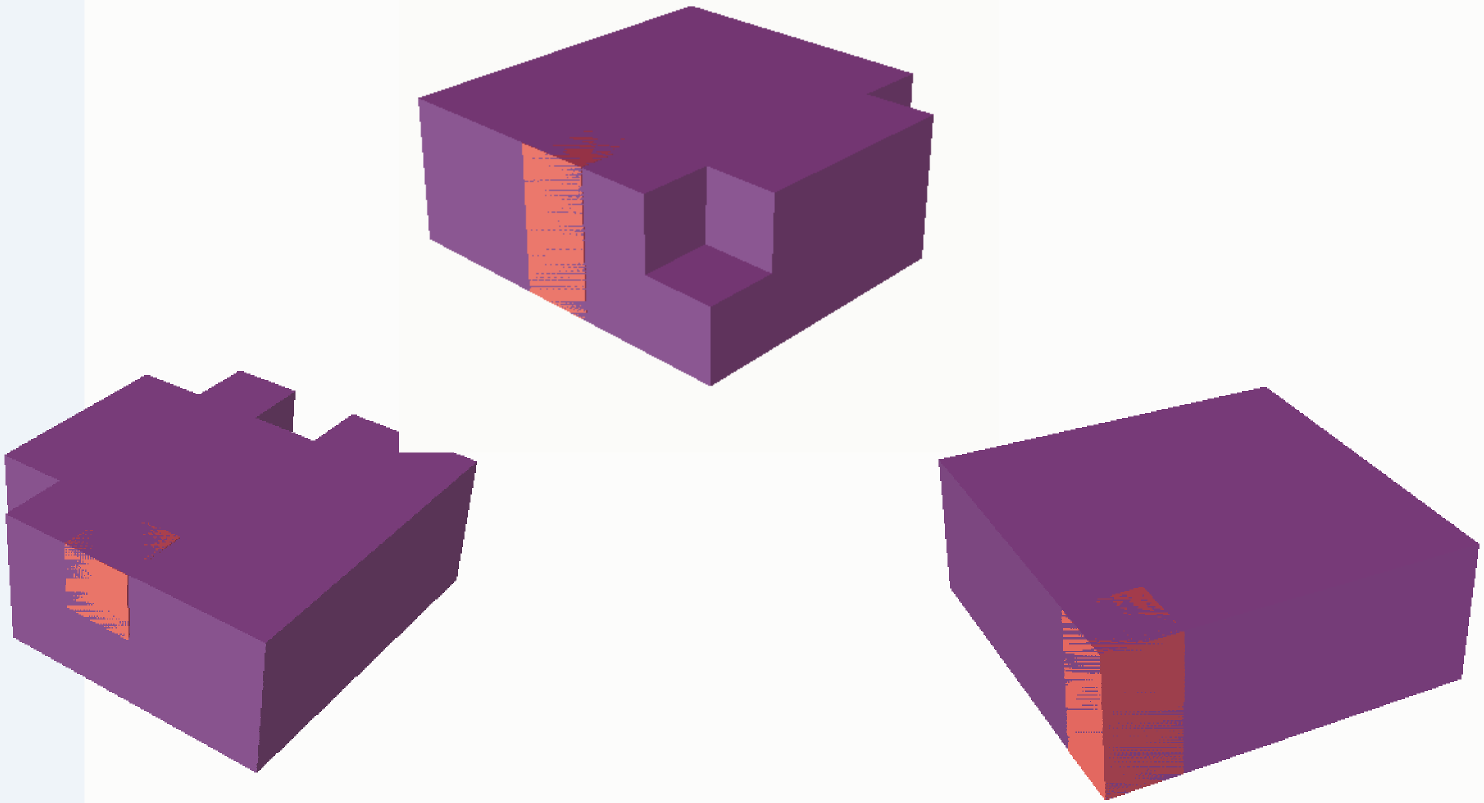
- intersect the polyhedron with one or more plans of its boundary to cut protrusions or fill holes.



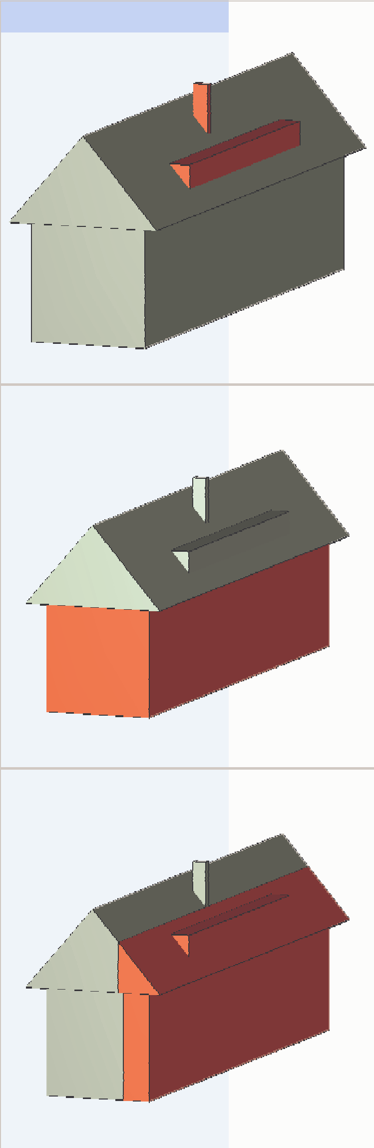


Complex Holes

- to fill complex holes more than one split-plane is needed



Quality of a split



- quality after Ribelles et. al

$$q = \frac{\text{area of the new inserted face}}{\text{area of the origin coplanar with the split plane}}$$

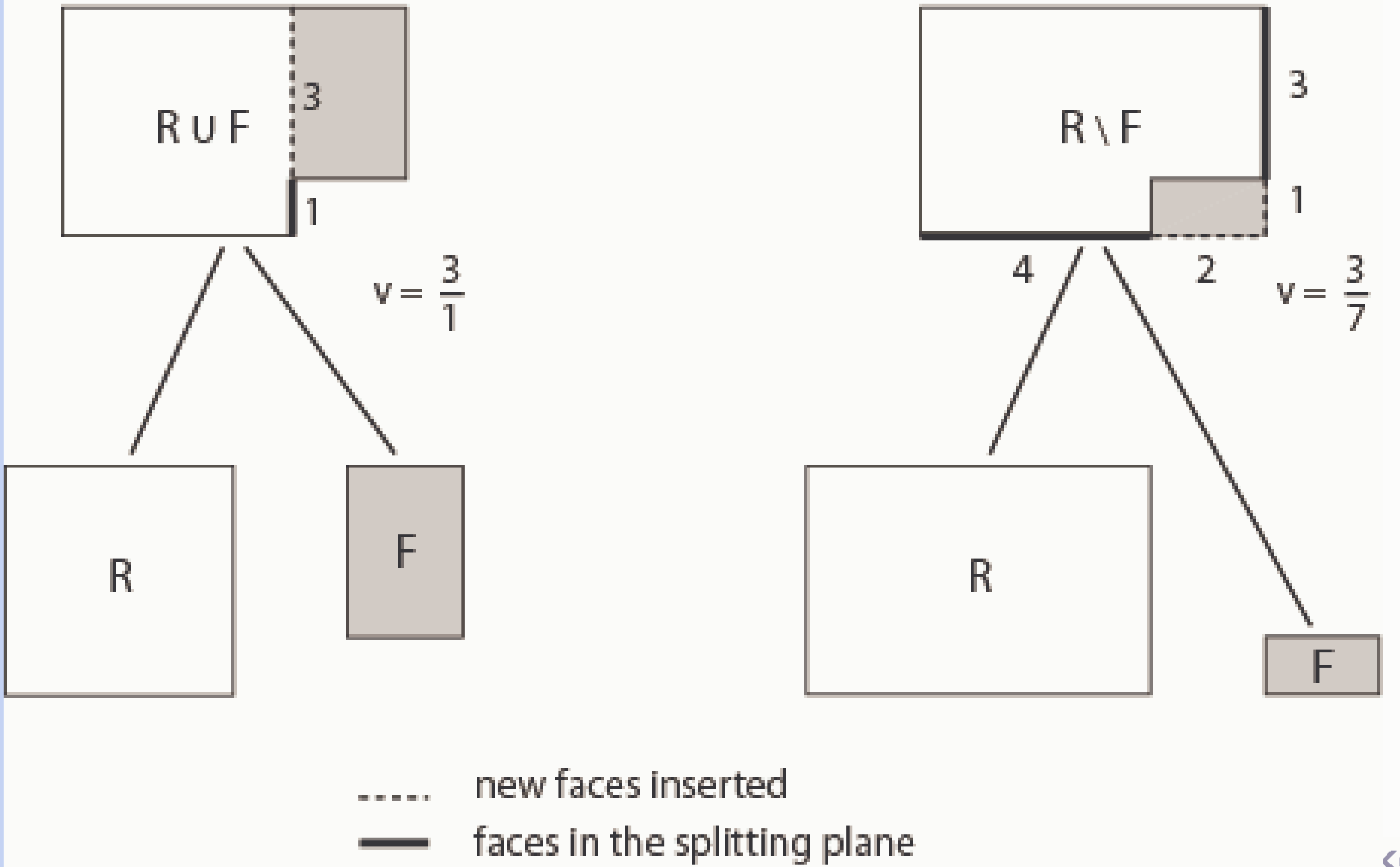
(q lower -> better)

- more than one feature → evaluate them separatly

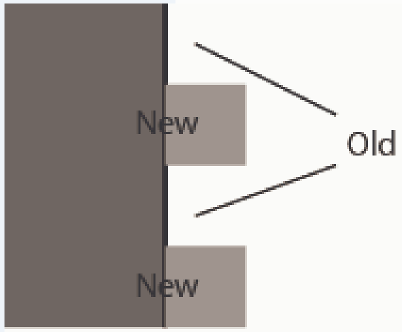
- validation of the split
  q < 1 → new area smaller than existing area

- using only the best split
- split the two parts recursively

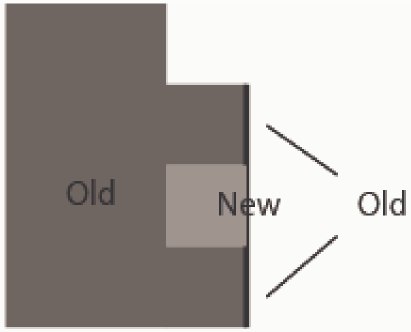
Quality value



Quality Value



Old =2, 2x New =1

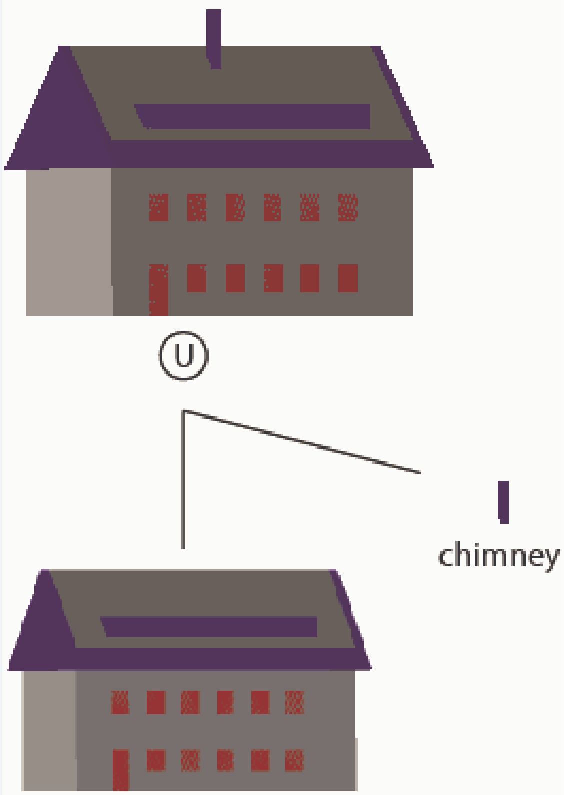


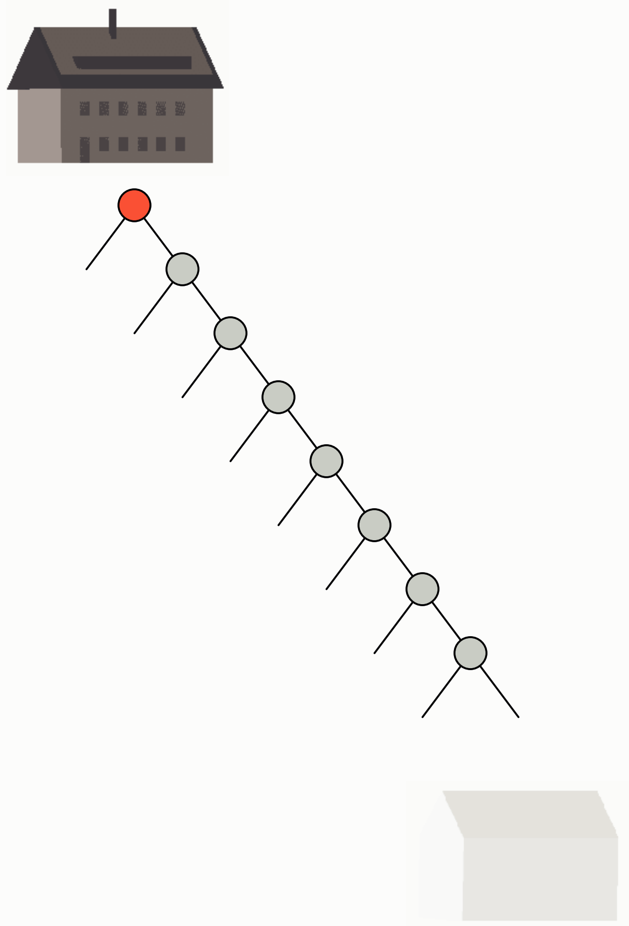
Old =2, New =1

Old = area of all facets laying in the split plane

New = area of the faces inserted to seperate the parts

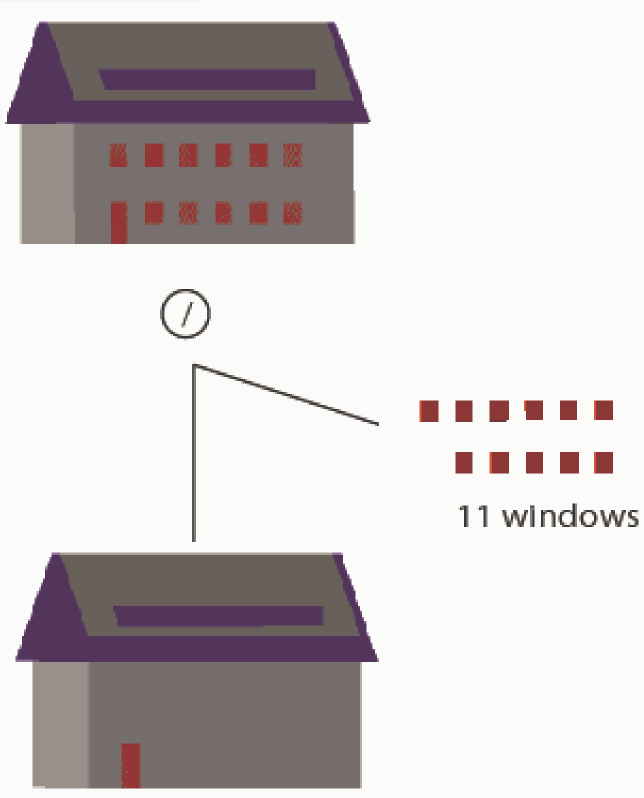
Sample building - Step 1

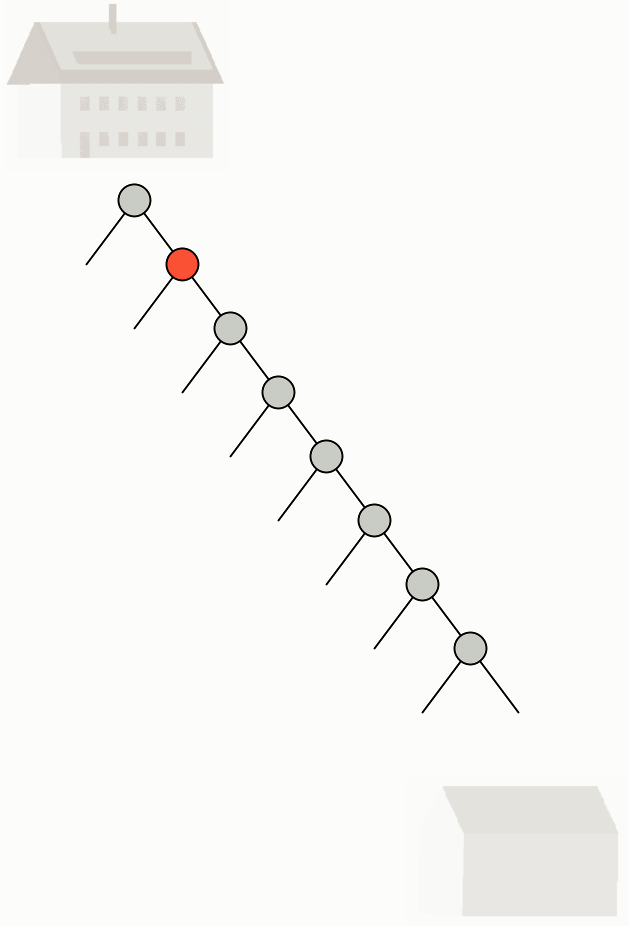




q= 0.003

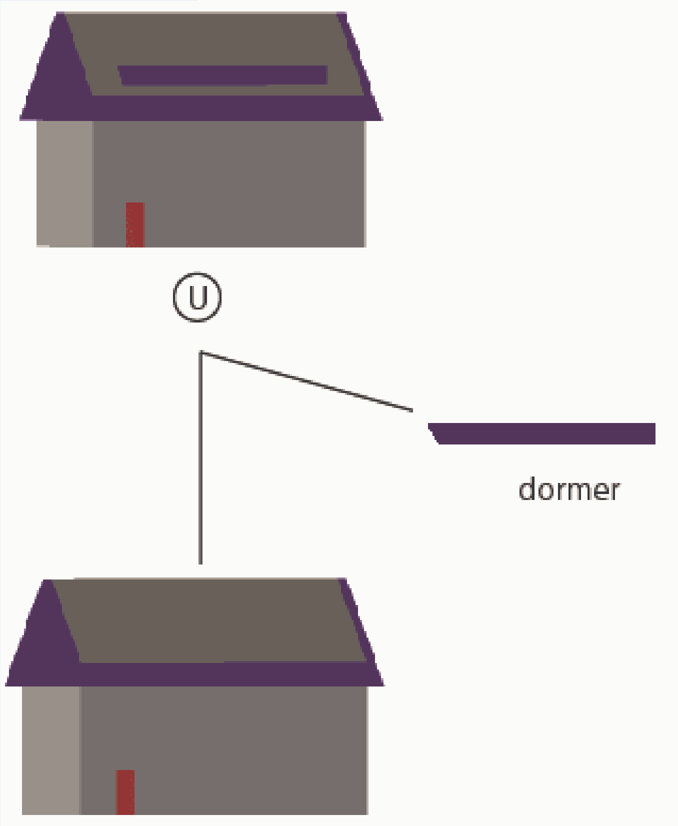
Sample building - Step 2

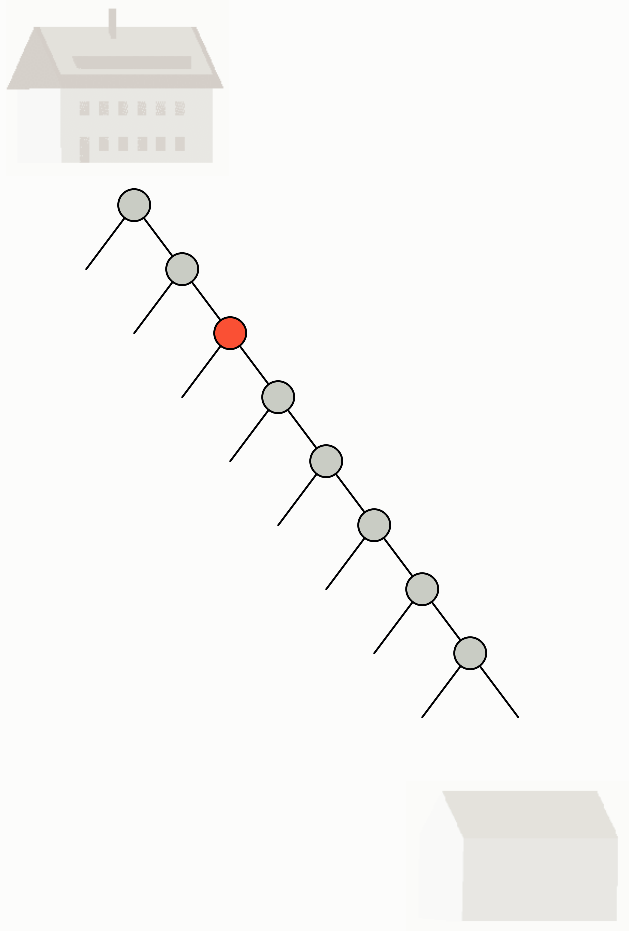




q= 0.014

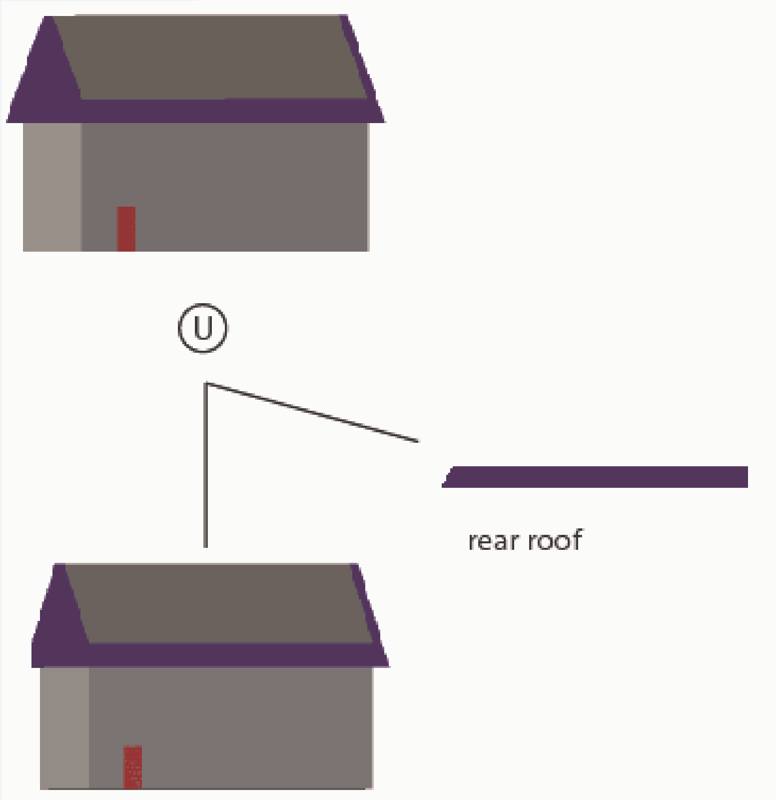
Sample building - Step 3

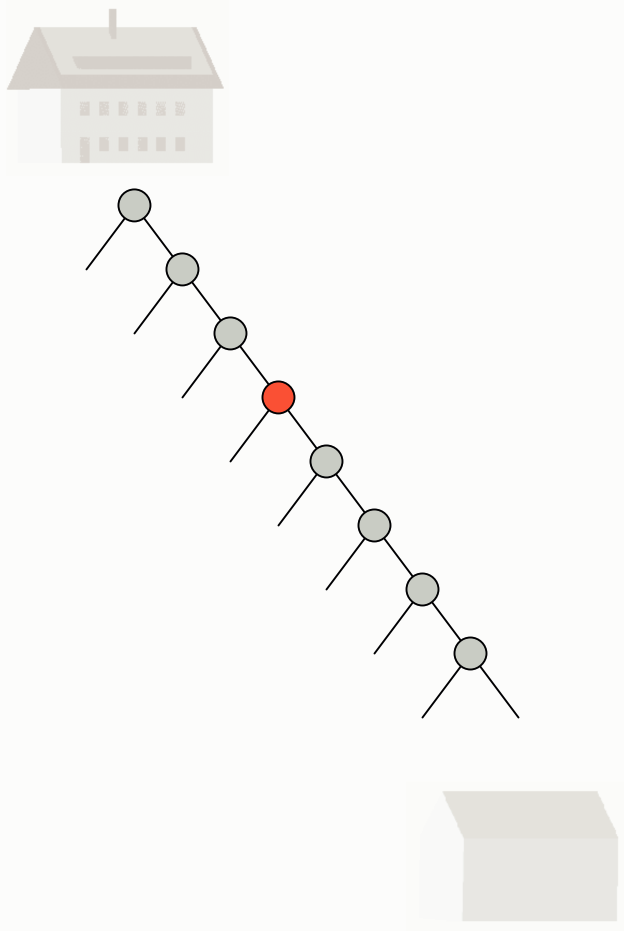




q= 0.162

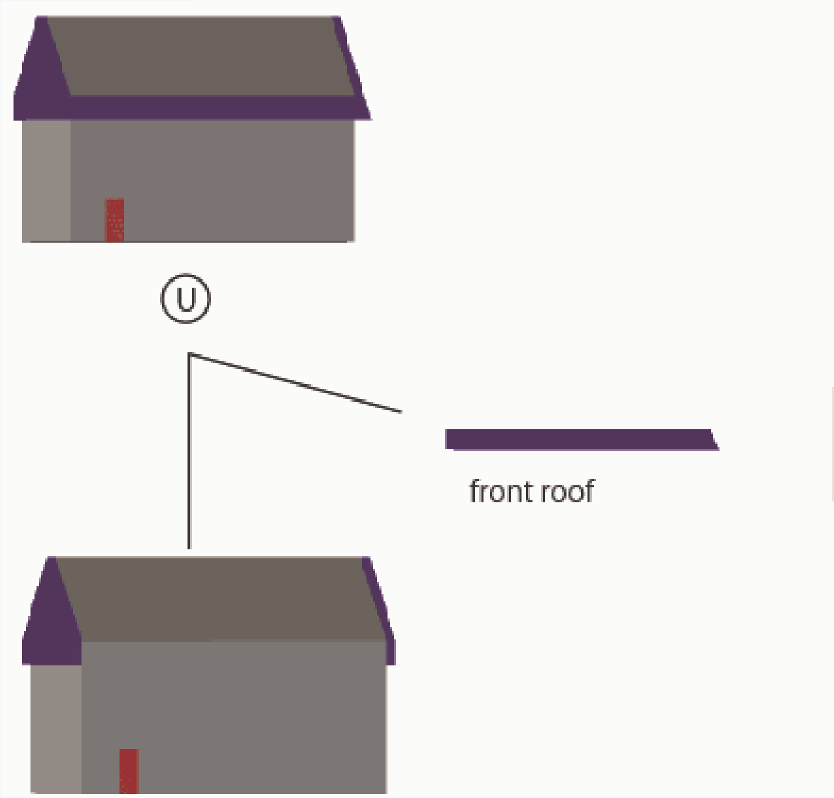
Sample building - Step 4

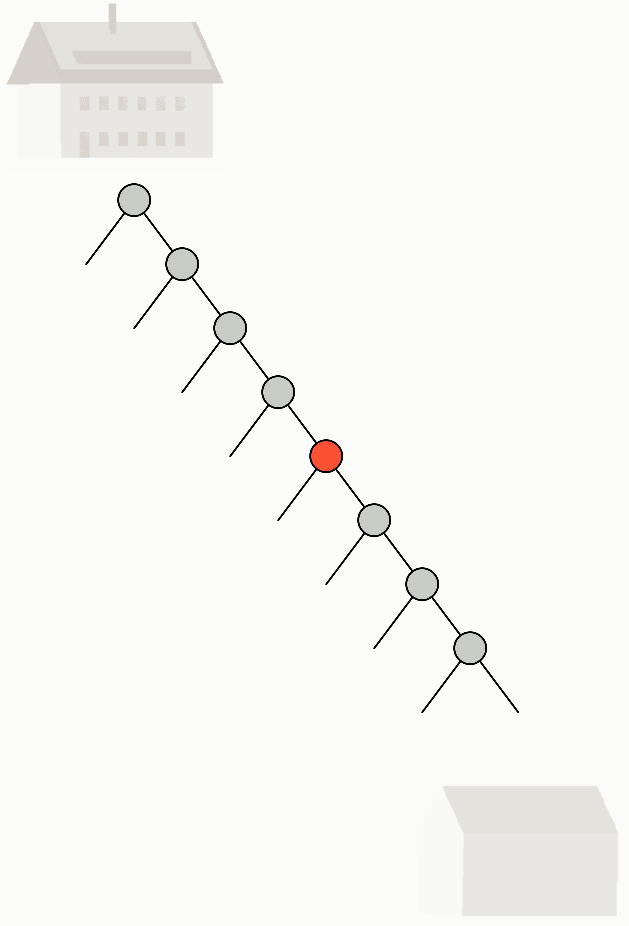




q= 0.205

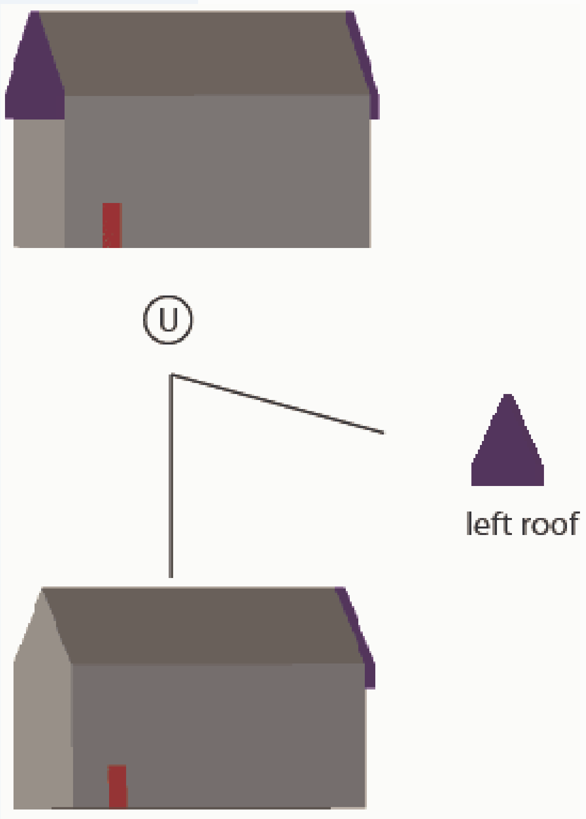
Sample building - Step 5

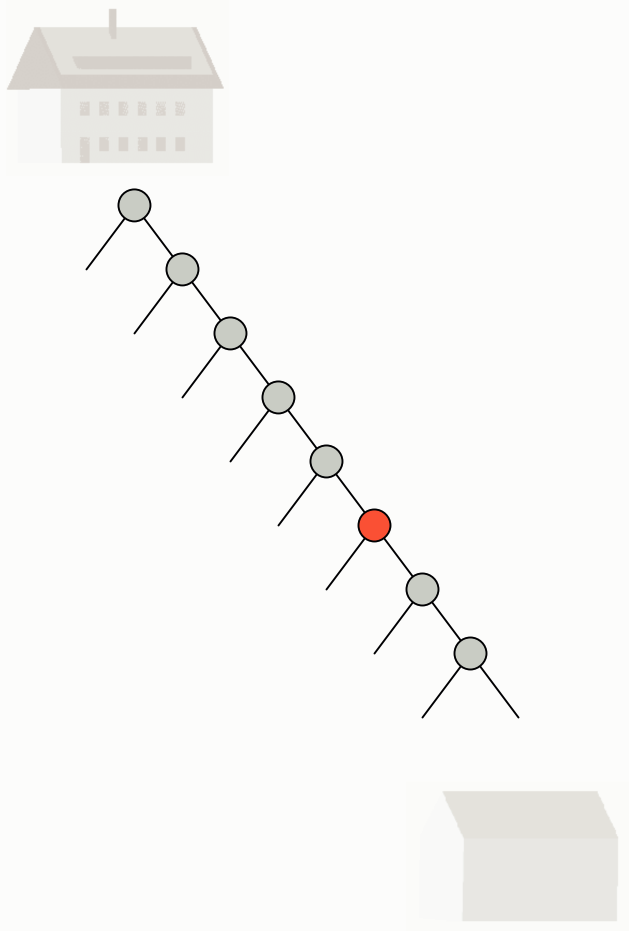




q= 0.210

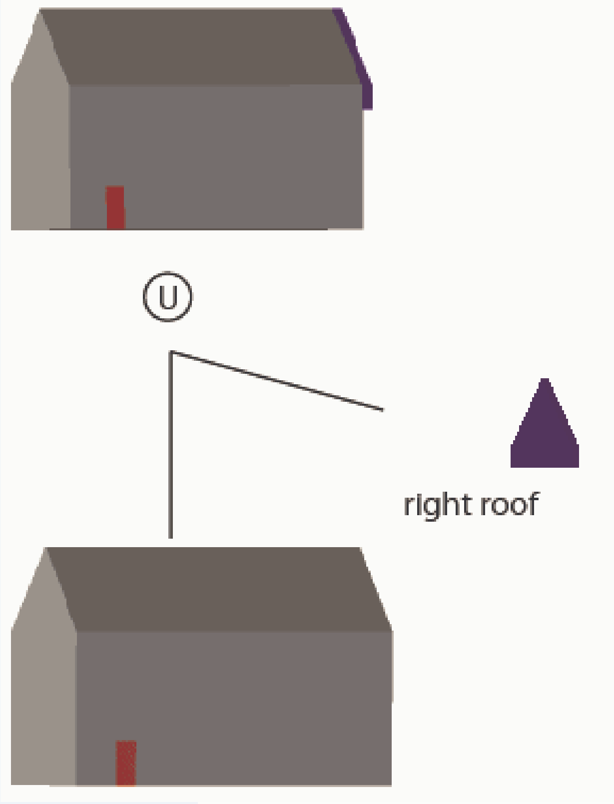
Sample building - Step 6

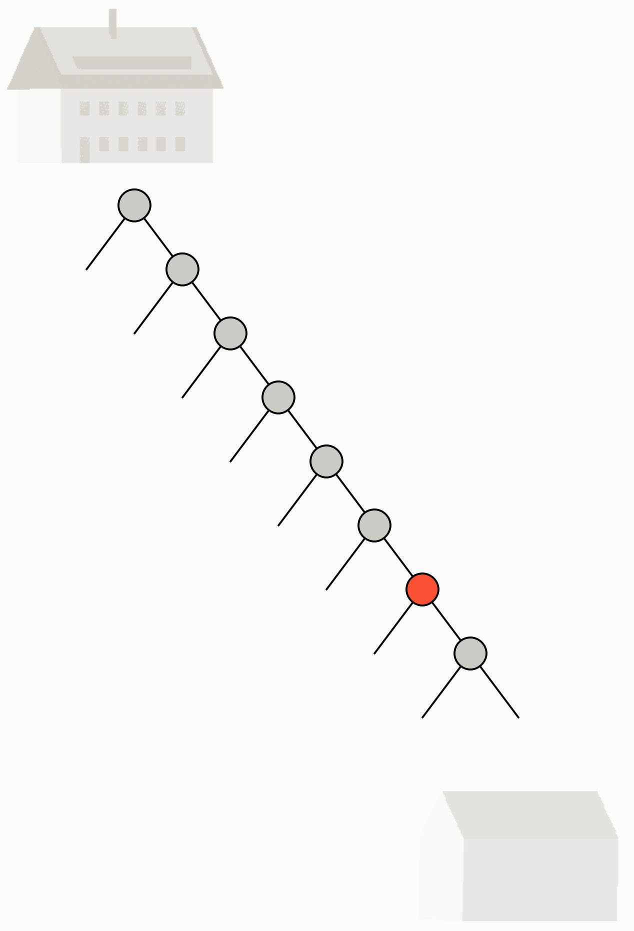




q= 0.517

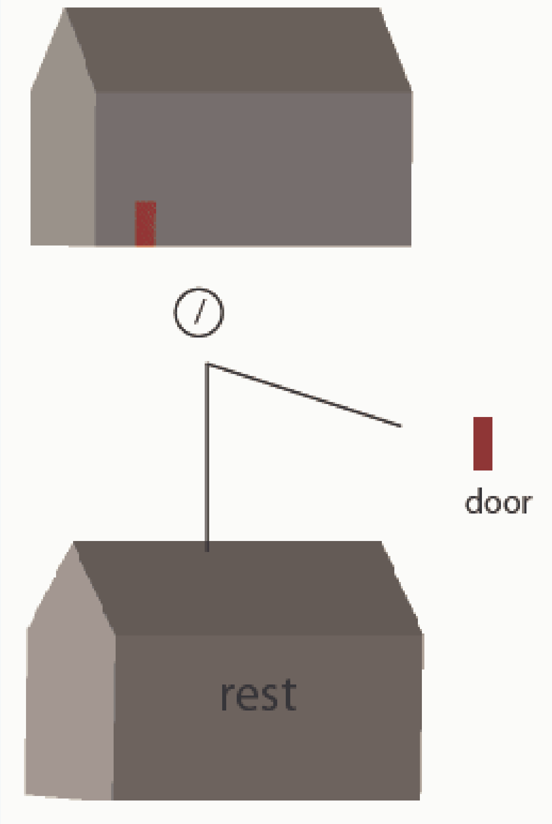
Sample building - Step 7

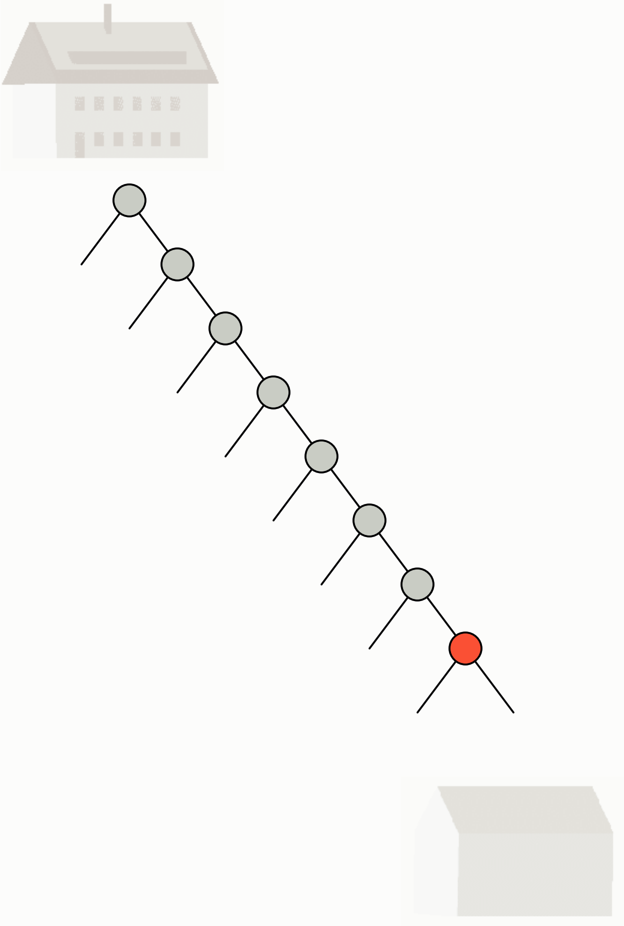




q= 0.517

Sample building - Step 8



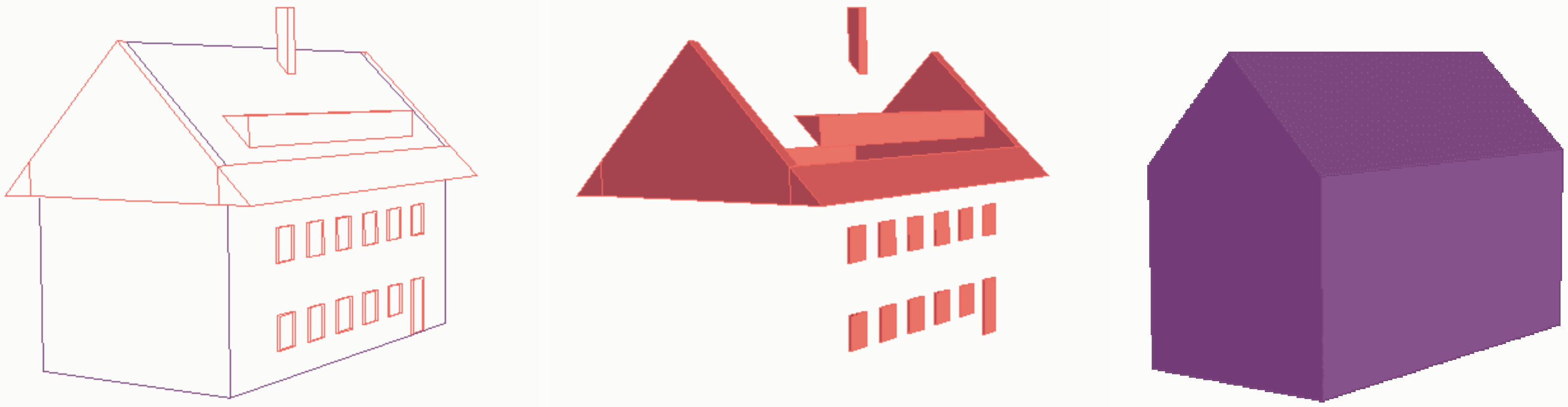


q= 0.009

Result of segmentation

- convex parts (protrusions and holes)
- CSG-tree (hierarchy of splits)

- not a generalisation!



Complexity of segmentation

- brute force: “try all combinations of planes for all parts”
  - $O(m \cdot n^{2k})$
    - n ... count of planes
    - m ... count of parts
    - k ... count of planes used at the same time

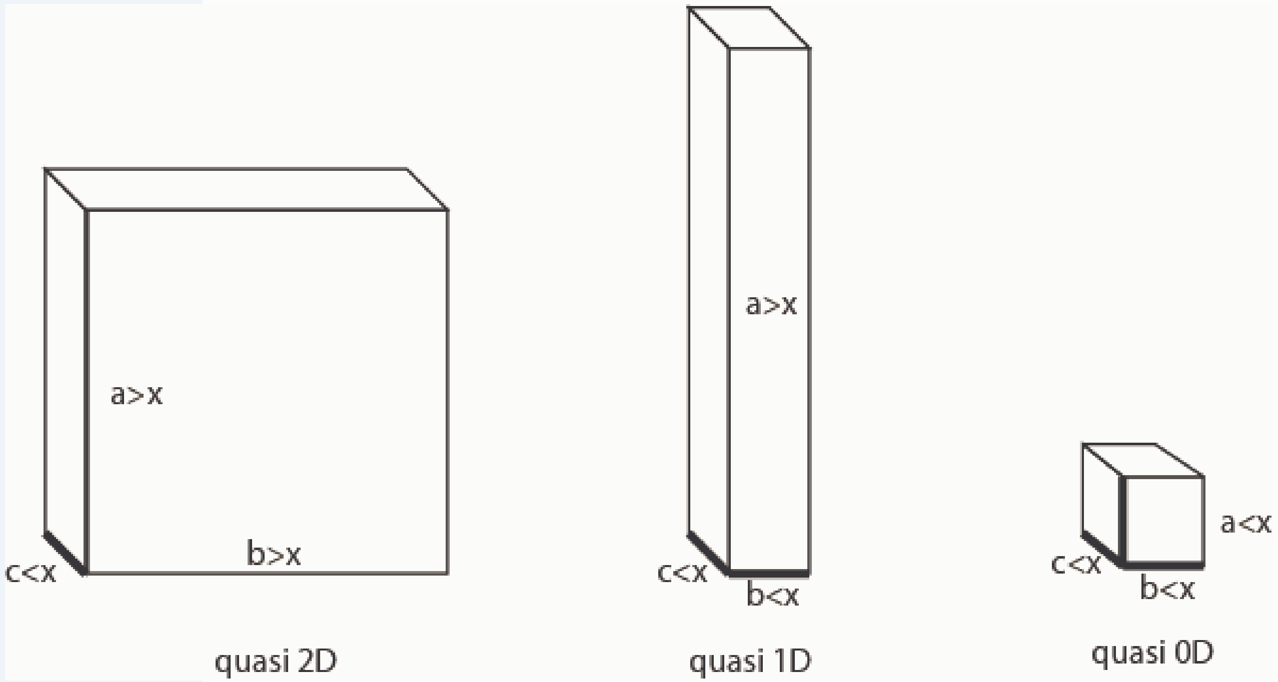
- optimization:
  - only planes with different normals at once
  - only anti parallel planes with positive distance

1. cut and fill with one plane at once (→ only simple holes)
2. fill complex holes with 2, 3 and 4 planes

Generalisation (generic determination)

- decide only on geometric parameters
  - size – but which?
    - extents
    - area
    - volume
  - is it observable?
    - visible area

Quasi k-D

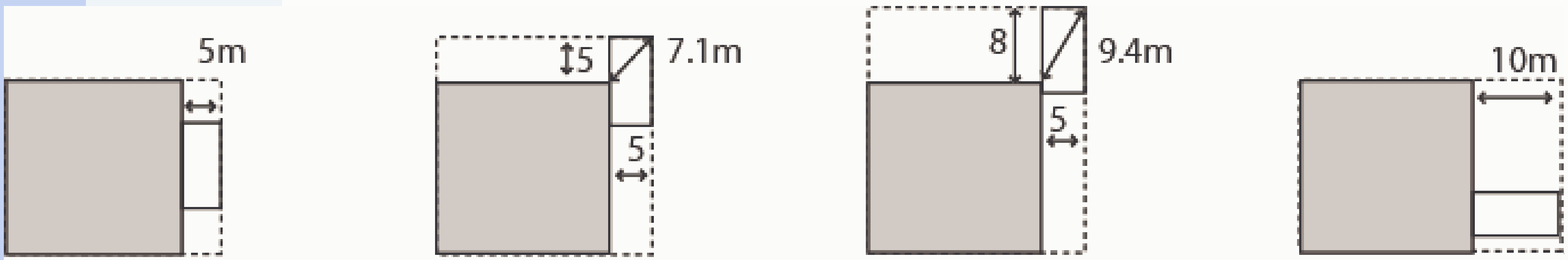


- quasi 2D – laminar - small in 1 dimension – a large area
- quasi 1D – linear – small in 2 dimensions – a large length
- quasi 0D – punctiform – small in all 3 dimensions

Quick-test for visibility using a bounding box

- sorting sides in descending order (a>=b>=c)
  - if $c > x_c$ → no generalisation necessary
  - $a < x_a$ → quasi 0D → omit, enlarge, ...
  - $b < x_b$ → quasi 1D → omit, enlarge, ...
  - $A = a\,b < x_A$ → omit, enlarge, ...
  - only quasi 2Ds are left

- Depending on the position of quasi-2Ds there is a smaller or a bigger difference in visibility.



Object depending Generalisation

- recognition of the “function” (window, door, roof, balcony, ...)
- attributes;
  - orientation
  - height
  - context
  - similarity (size) in given neighbourhood (distance)
- knowledge
  - typical size / orientation / positions
  - typical height of stories

Conclusions

- algorithm of Ribelles et al.
  - is suited to separate buildings in meaningful parts
  - needs optimization to reduce runtime
- based on the derived features
  - generalisation with generic determination gets possible
  - some optimizations of the CSG tree could become necessary
- object depending generalisation
  - symbolisation, aggregation, typification need semantic information
  - recognition of the semantic type (roof, window ...)

End

- Thank you for listening.