Mesh Simplification for Building Typification

Dirk Burghardt and Alessandro Cecconi

Outline

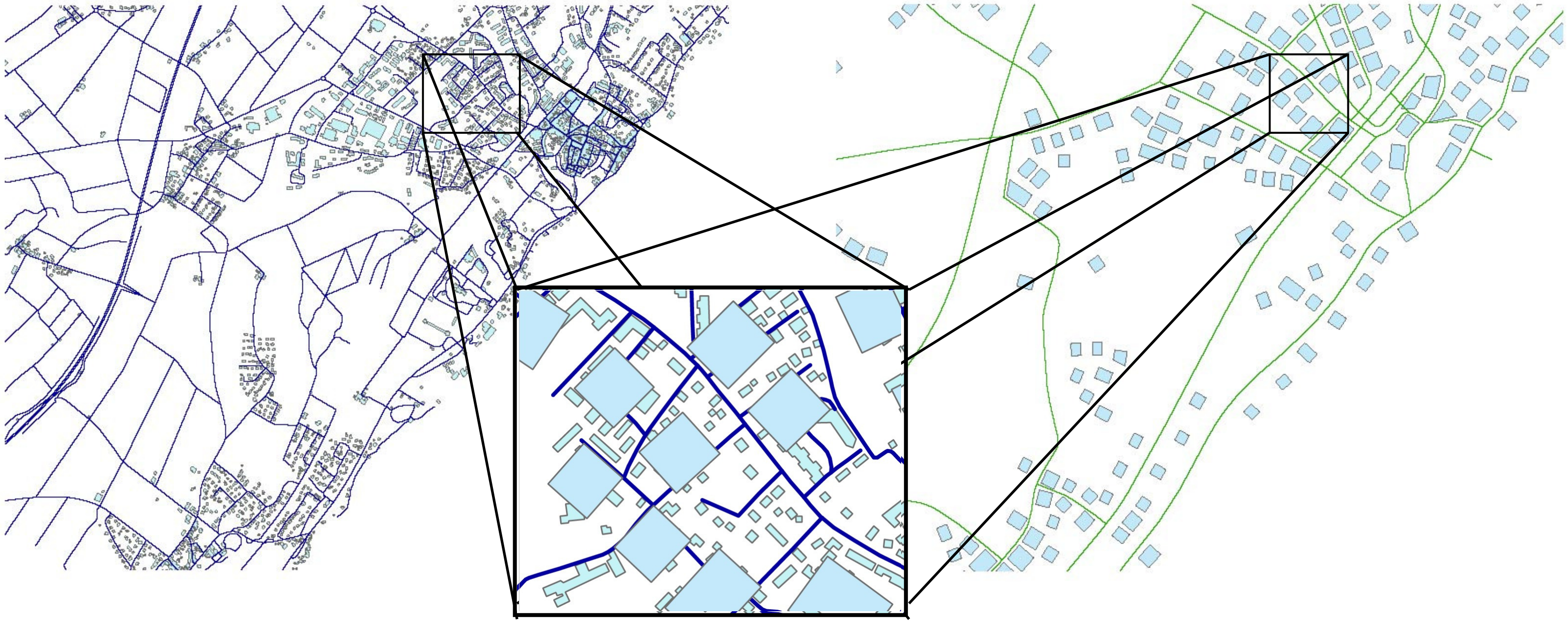
1. Background and context
2. Mesh simplification
    a) Theory of mesh optimization
    b) Mesh simplification adapted for typification
    c) Shape construction
3. Control parameter
    a) Control of building density
    b) Semantic control
4. Limitations and possible improvements

Background and context

Task: generate map of intermediate scale with help of multiscale database

Given: two layer / level of detail (scale 1:25'000 and 1:200'000)

Fokus: typification of buildings

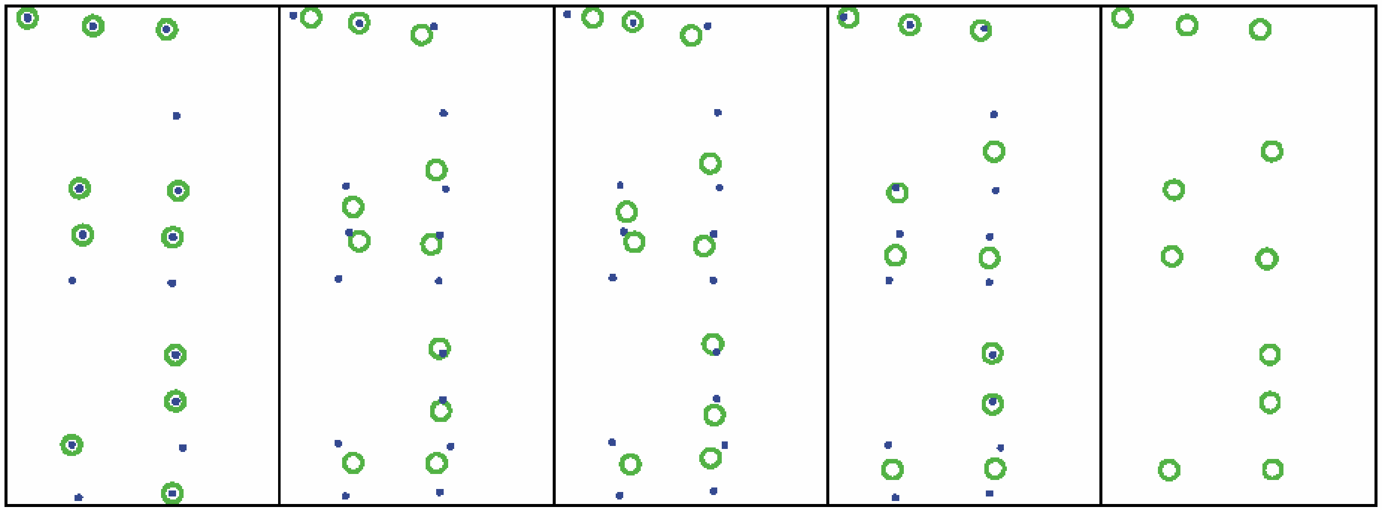


Background and context

Publications:

Regnauld, N. (1996): Recognition of Building Clusters for Generalization.

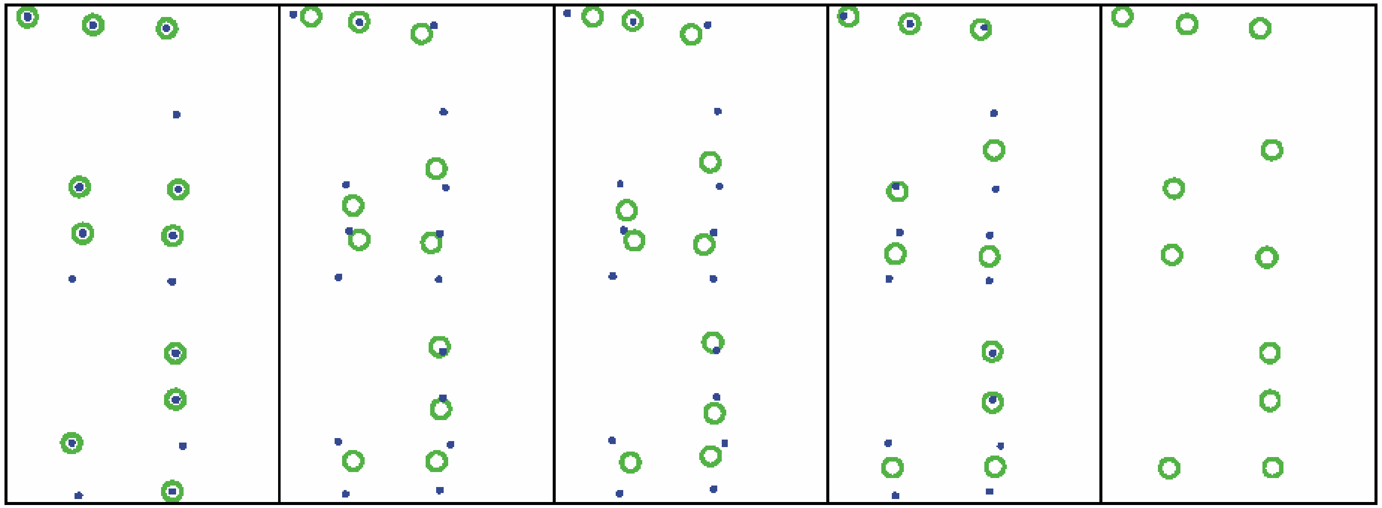
Sester, M. and Brenner (2000): Typification Based on Kohonen Feature Nets.



Sester, M. and Brenner (2000): Typification Based on Kohonen Feature Nets.

Kohonen Feature Maps

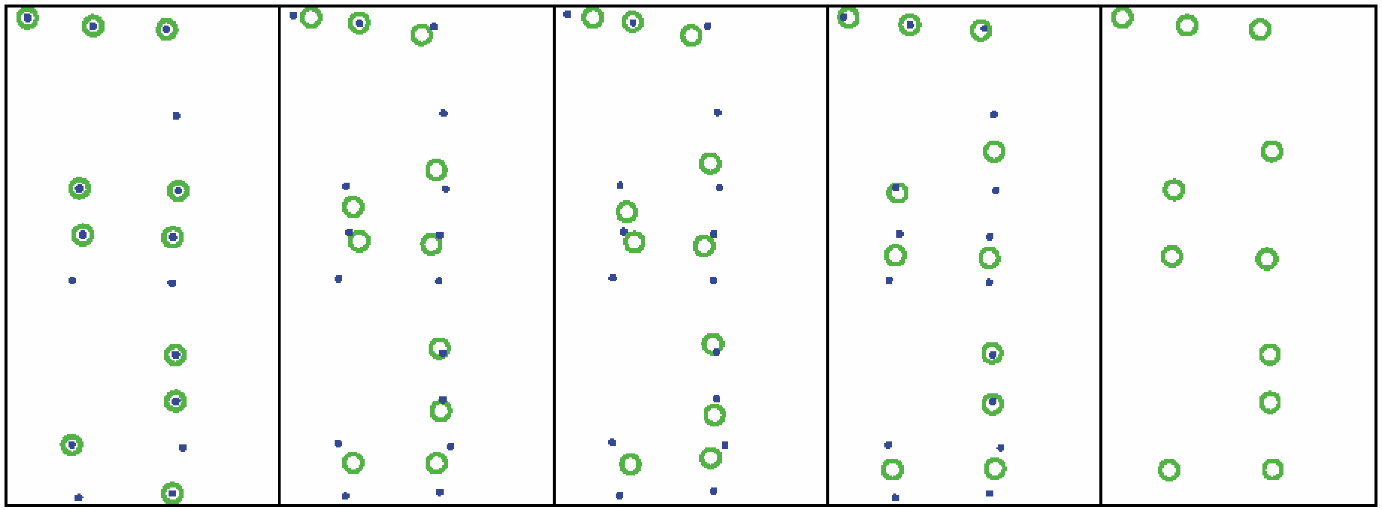
- based on learning techniques with neuronal nets
- neurons are adapted to new situtation, while keeping their spatial ordering
- good results



Sester, M. and Brenner (2000): Typification Based on Kohonen Feature Nets.

Kohonen Feature Maps

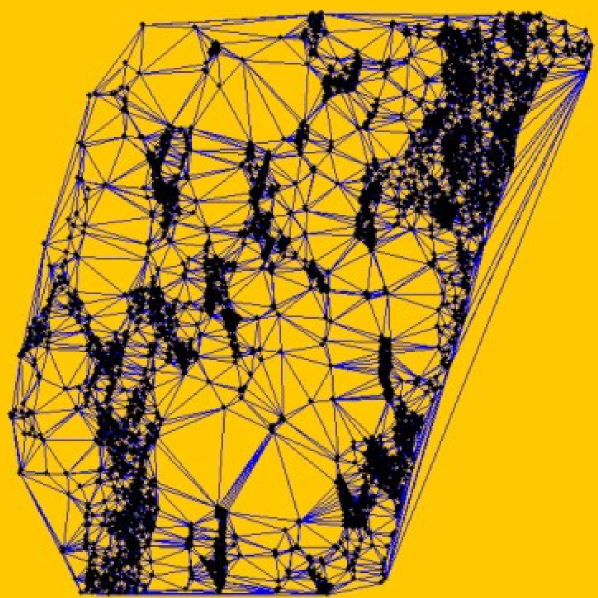
- approach is non deterministic -
  as a result of random selection of neurons at the beginning → after rerun different results will be achieved
- **goal**: looking for an algorithm, which creates reproducible results



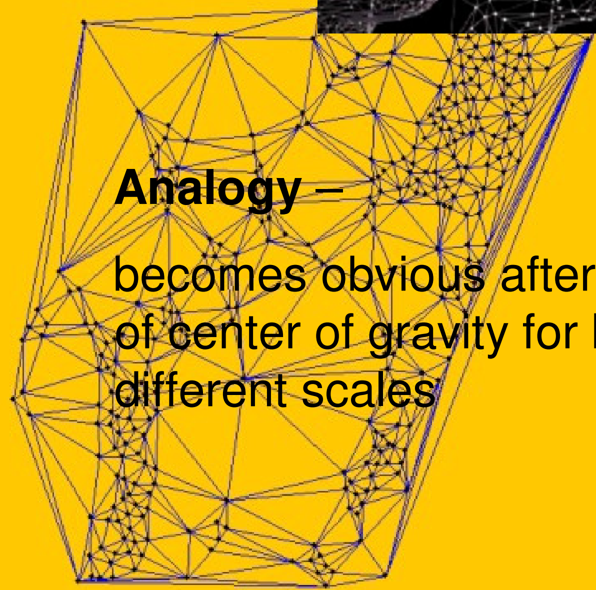
Sester, M. and Brenner (2000): Typification Based on Kohonen Feature Nets.

Why mesh optimization?

application of mesh optimization
in computer graphics



Scale: 1:25‘000



Scale: 1:200‘000

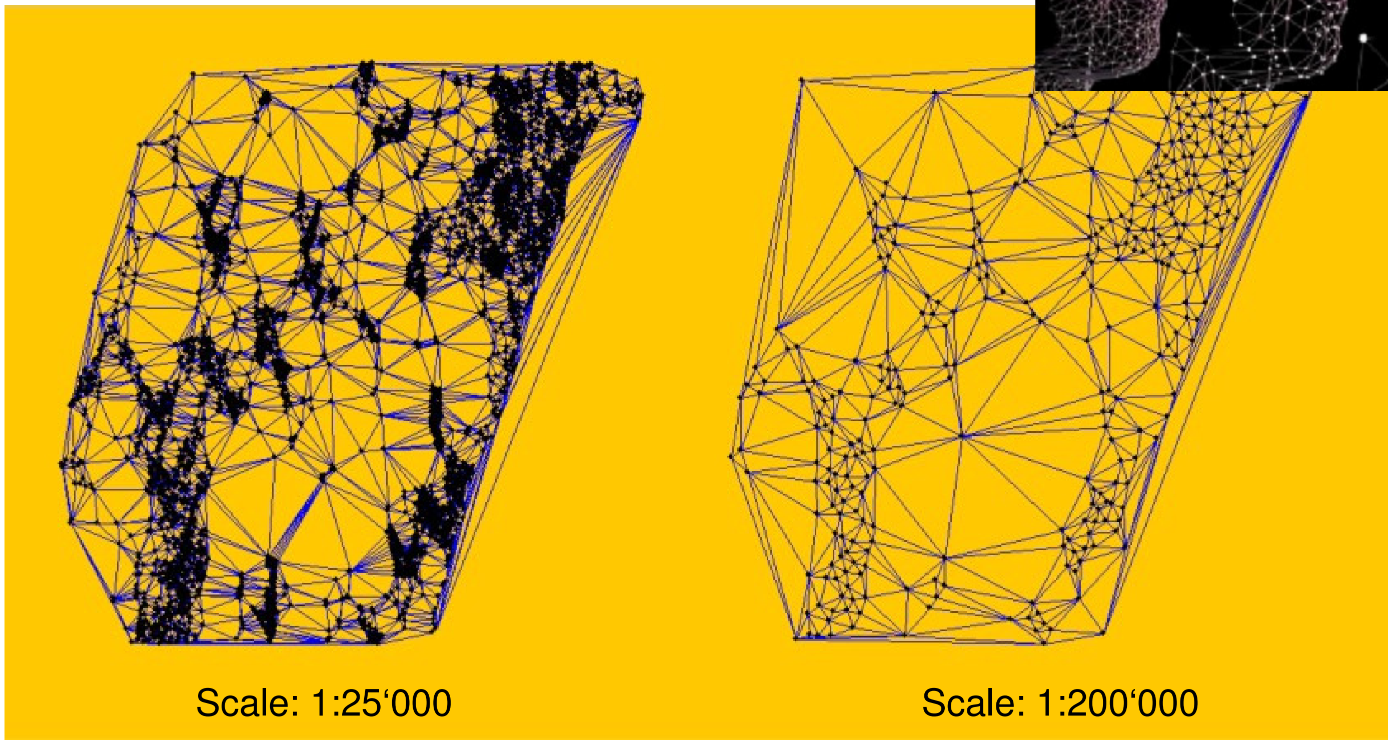
Analogy –

becomes obvious after triangulation of center of gravity for buildings in different scales

Why mesh optimization?

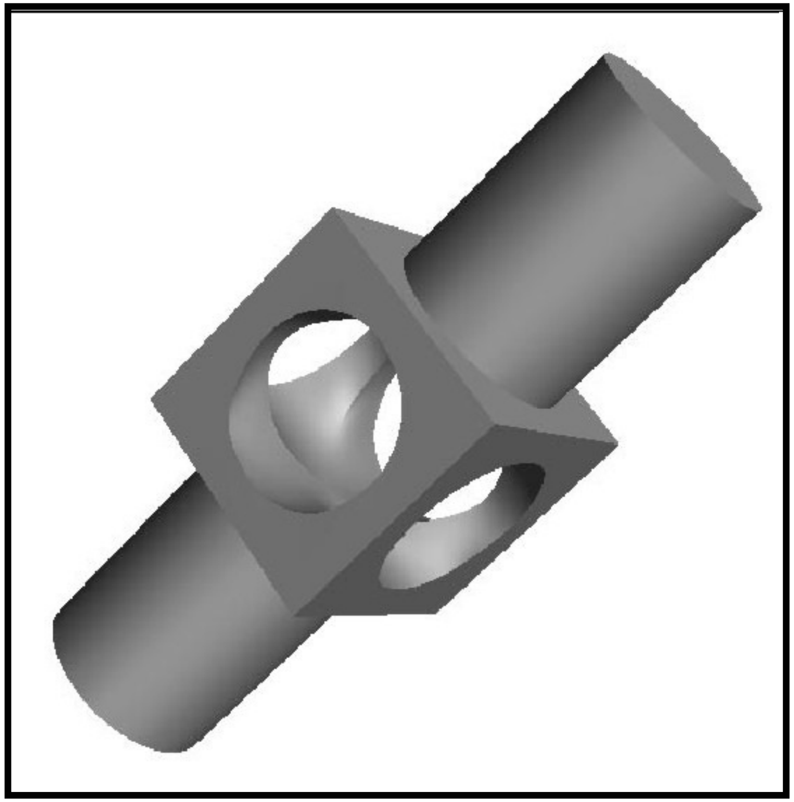
- because the results are comparable -

application of mesh optimization in computer graphics



Mesh Optimization

Method for automated reconstruction of 3D objects with optimal triangulation nets



- surface reconstruction
- mesh optimization
- smoothing

Mesh Optimization

Objective: Simplification of triangulation, through selection with use of energy function and not with random delete of edges and nodes

$$E = E_{dist} + E_{rep} + E_{spring}$$

$$E_{dist}(K,V) = \sum_i d^2(\mathbf{x}_i, \phi_V(|K|))$$

mit $d^2(\mathbf{x}_i, \phi_V(|K|)) = \min_{\mathbf{b}_i \in |K|} \left\| \mathbf{x}_i - \pi_V(\mathbf{b}_i) \right\|^2$

describes the distance of the meshes from the scaned 3D original points

Mesh Optimization ⟶ Typification

Objective: Simplification of triangulation, through selection with use of energy function and not with random delete of edges and nodes

$$E = E_{dist} + E_{rep} + E_{spring}$$

$$E_{dist}(K,V) = \sum_i d^2(\mathbf{x}_i, \phi_V(|K|)) \longrightarrow E_{node}(V) = \sum_i d^2(\mathbf{x}_i^{orig}, \mathbf{x}^{rep})$$

mit $d^2(\mathbf{x}_i, \phi_V(|K|)) = \min_{\mathbf{b}_i \in |K|} \left\| \mathbf{x}_i - \pi_V(\mathbf{b}_i) \right\|^2$

mit $d^2 = \sum_i \min_j \left\| \mathbf{x}_i^{orig} - \mathbf{x}_j^{rep} \right\|^2$

describes distance of representatives from the original nodes

Mesh Optimization ⟶ Typification

Objective: Simplification of triangulation, through selection with use of energy function and not with random delete of edges and nodes

$$E = E_{dist} + E_{rep} + E_{spring}$$

$$E_{dist}(K,V) = \sum_i d^2(\mathbf{x}_i, \phi_V(|K|)) \longrightarrow E_{node}(V) = \sum_i d^2(\mathbf{x}_i^{orig}, \mathbf{x}^{rep})$$

mit $d^2(\mathbf{x}_i, \phi_V(|K|)) = \min_{\mathbf{b}_i \in |K|} \left\| \mathbf{x}_i - \pi_V(\mathbf{b}_i) \right\|^2$

mit $d^2 = \sum_i \min_j \left\| \mathbf{x}_i^{orig} - \mathbf{x}_j^{rep} \right\|^2$

$$E_{rep}(K) = c_{rep} m$$

reduction of point number ☺

Mesh Optimization → Typification

Objective: Simplification of triangulation, through selection with use of energy function and not with random delete of edges and nodes

$$E = E_{dist} + E_{rep} + E_{spring}$$

$$E_{dist}(K,V) = \sum_i d^2(\mathbf{x}_i, \phi_V(|K|))$$
mit $d^2(\mathbf{x}_i, \phi_V(|K|)) = \min_{\mathbf{b}_i \in |K|} \left\| \mathbf{x}_i - \pi_V(\mathbf{b}_i) \right\|^2$

→

$$E_{node}(V) = \sum_i d^2(\mathbf{x}_i^{orig}, \mathbf{x}^{rep})$$
mit $d^2 = \sum_i \min_j \left\| \mathbf{x}_i^{orig} - \mathbf{x}_j^{rep} \right\|^2$

$$E_{rep}(K) = c_{rep} m$$

reduction of point number ☺

$$E_{spring}(K,V) = \sum_{\{j,k\} \in K} k \left\| \mathbf{v}_j - \mathbf{v}_k \right\|^2$$

→

$$E_{edge} = \frac{1}{2} \alpha \sum_{edge(DT)} \left(\Delta l^{akt} \right)^2$$

keeps the length of delaunay edges

Mesh Optimization ⟶ Typification

Objective: Simplification of triangulation, through selection with use of energy function and not with random delete of edges and nodes

$$E = E_{dist} + E_{rep} + E_{spring} \longrightarrow E = E_{node} + E_{rep}$$

$$E_{dist}(K,V) = \sum_i d^2(\mathbf{x}_i, \phi_V(|K|)) \longrightarrow E_{node}(V) = \sum_i d^2(\mathbf{x}_i^{orig}, \mathbf{x}^{rep})$$

mit $d^2(\mathbf{x}_i, \phi_V(|K|)) = \min_{\mathbf{b}_i \in |K|} \left\| \mathbf{x}_i - \pi_V(\mathbf{b}_i) \right\|^2$ mit $d^2 = \sum_i \min_j \left\| \mathbf{x}_i^{orig} - \mathbf{x}_j^{rep} \right\|^2$

$$E_{rep}(K) = c_{rep} m$$

reduction of point number ☺

$$E_{spring}(K,V) = \sum_{\{j,k\} \in K} k \left\| \mathbf{v}_j - \mathbf{v}_k \right\|^2 \longrightarrow E_{edge} = \frac{1}{2}\alpha \sum_{edge(DT)} \left(\Delta l^{akt} \right)^2$$

Typification

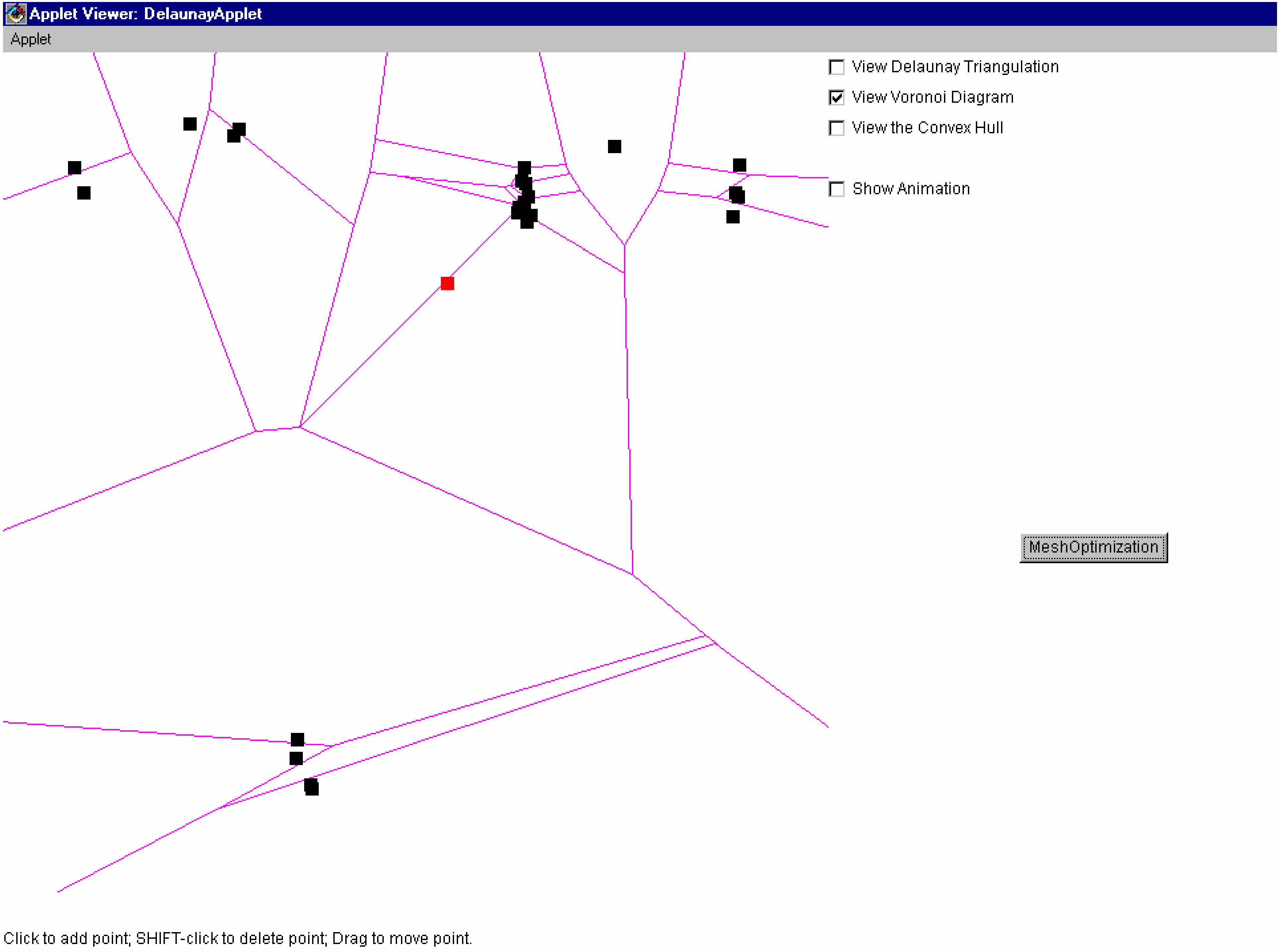
- External loop looked for the representatives, which could be removed with the smallest energy increase
- Internal loop optimize the position of remaining representatives for the given energy function with help of QR decomposition

$$\left\| \underset{(n+e,m)}{\mathbf{M}} \cdot \underset{(m,1)}{\mathbf{x}^1} - \underset{(n+e,1)}{\mathbf{d}^1} \right\|^2$$

M – design matrix, derived from energy terms

x – vector of representatives

d – vector of original objects (center of gravity)


Applet Viewer: DelaunayApplet
Applet
View Delaunay Triangulation
View Voronoi Diagram
View the Convex Hull
Show Animation
MeshOptimization
Click to add point; SHIFT-click to delete point; Drag to move point.

Conclusion of pre-project

Advantages

- mesh optimization technique could be used for typification
- open to include additional constraints

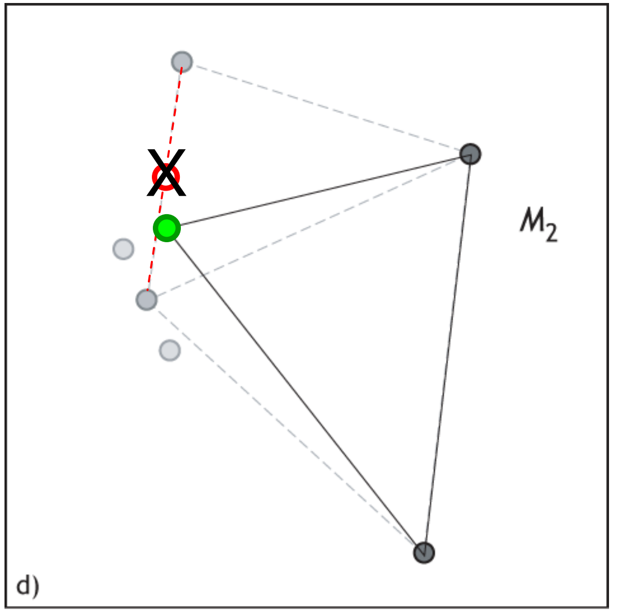
Disadvantages

- approach depends n^2 from the number of objects

→ slow if number of objects more then 20-30 !

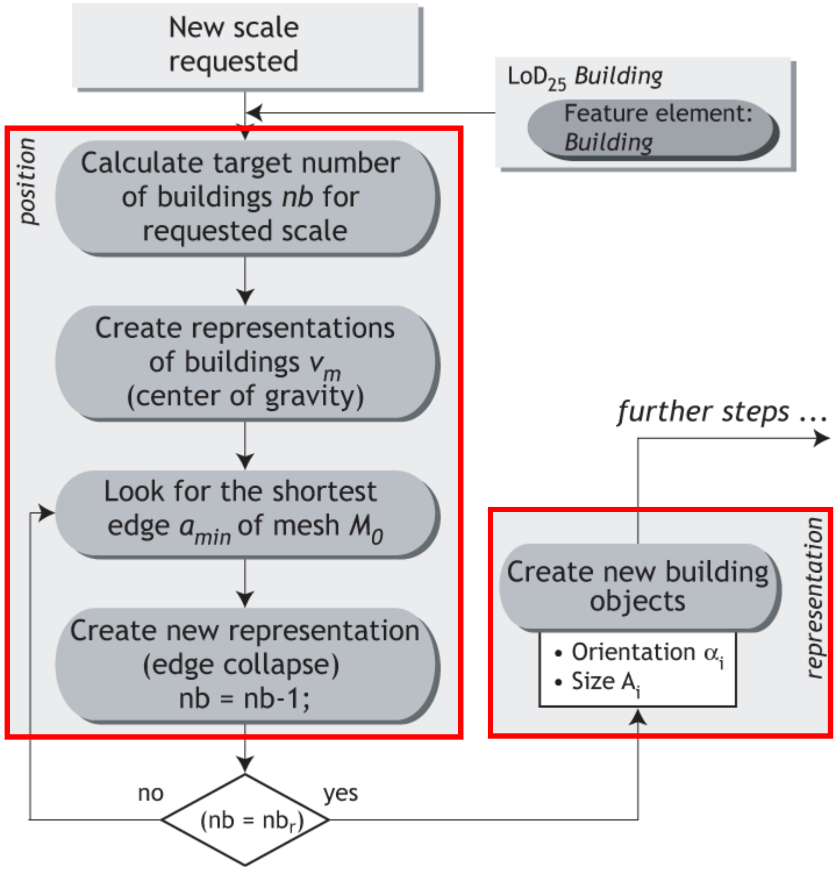
Alternative – mesh simplification

- faster geometric solution – mesh simplification
- Iteration:
  look for shortest edge between representatives and delete edge (replace two representatives with one)



- determine center of gravity from all objects where the representatives stands for

Building typification process



two step process -

1. **Positioning**:
   Determining the *number* and the *position* of the new objects with respect to the requested scale

2. **Representation**:
   Creation of a new building objects with calculation of *size* and *orientation*

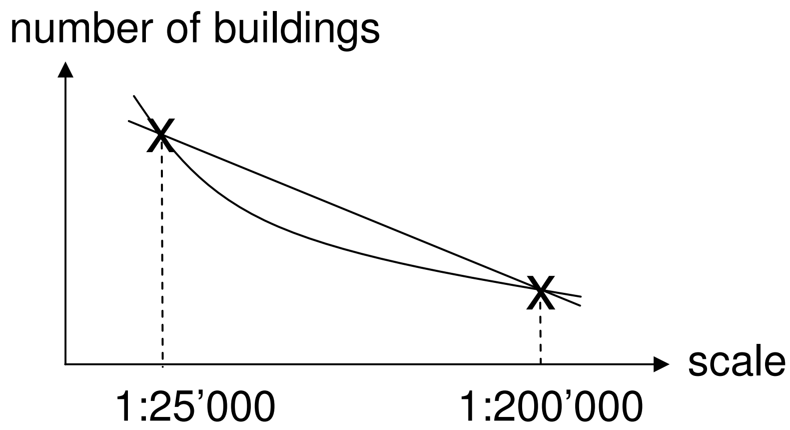
1. Positioning

a) Determine number of objects

- use Töpfer's radical law
  (Töpfer and Pillewizer 1966)

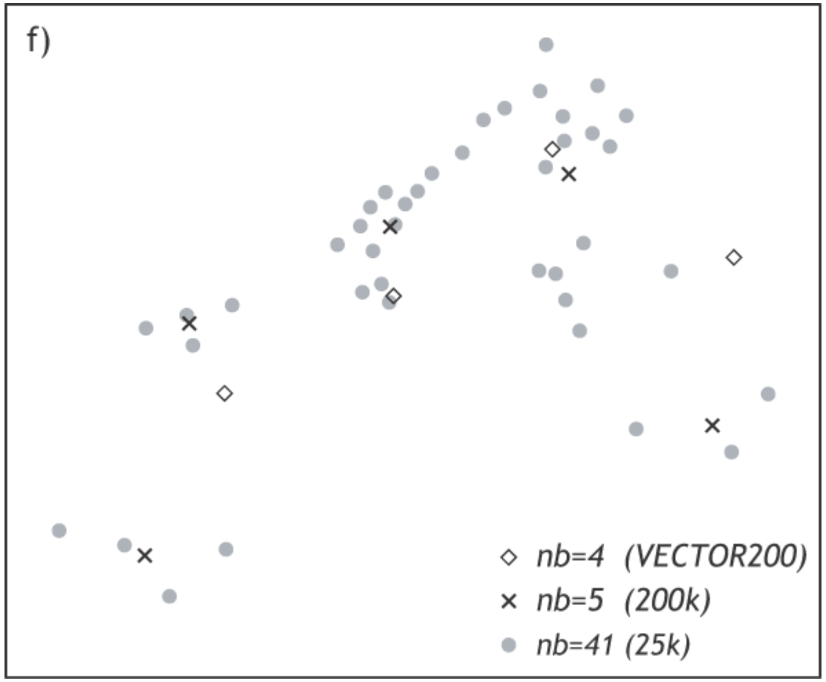
$$n_{build}^{dest} = n_{build}^{25}\sqrt{\frac{m_{25}}{m_{dest}}}$$

- keep the „black-white" ratio between buildings and background constant for all scales
- calculate interpolation function (linear, root) from number of buildings at the two scales (1:25'000 and 1:200'000)



1. Positioning

b) Determine position with mesh simplification



2. Representation

a) Calculation of area

- simple shape presentation with rectangles
- size of areas is computed out of the average of the pertaining buildings at scale 1:25'000
- a scaling factor f_{area} is used to consider additionally the size of buildings at scale 1:200'000

$$\overline{A_n} = \frac{\sum_k A_k}{n_{dest}} \cdot f_{area} = \frac{\sum_k A_k}{n_{dest}} \cdot \underbrace{\frac{m_{dest}^2}{m_{25}^2} \cdot \left(1.0 - \Psi \cdot \frac{m_{dest} - m_{25}}{m_{200} - m_{25}} \right)}_{f_{area}}$$

$$\Psi = 1.0 - \frac{m_{25}^2}{m_{200}^2} \cdot \frac{\overline{A_{LOD200}}}{\overline{A_n}}$$

2. Representation

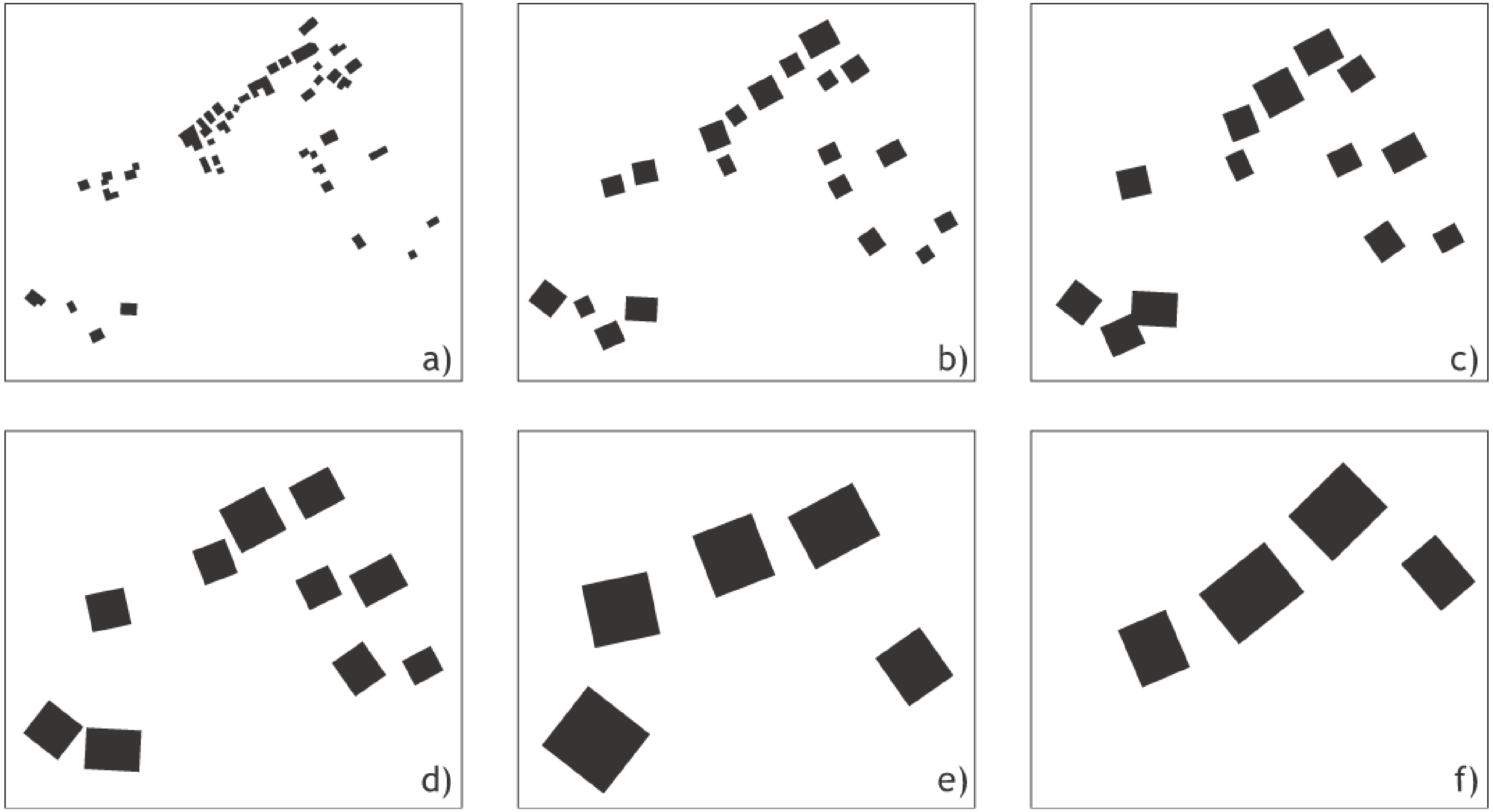
a) Calculation of area

- simple shape presentation with rectangles
- size of areas is computed out of the average of the pertaining buildings at scale 1:25’000
- a scaling factor f_{area} is used to consider additionally the size of buildings at scale 1:200’000

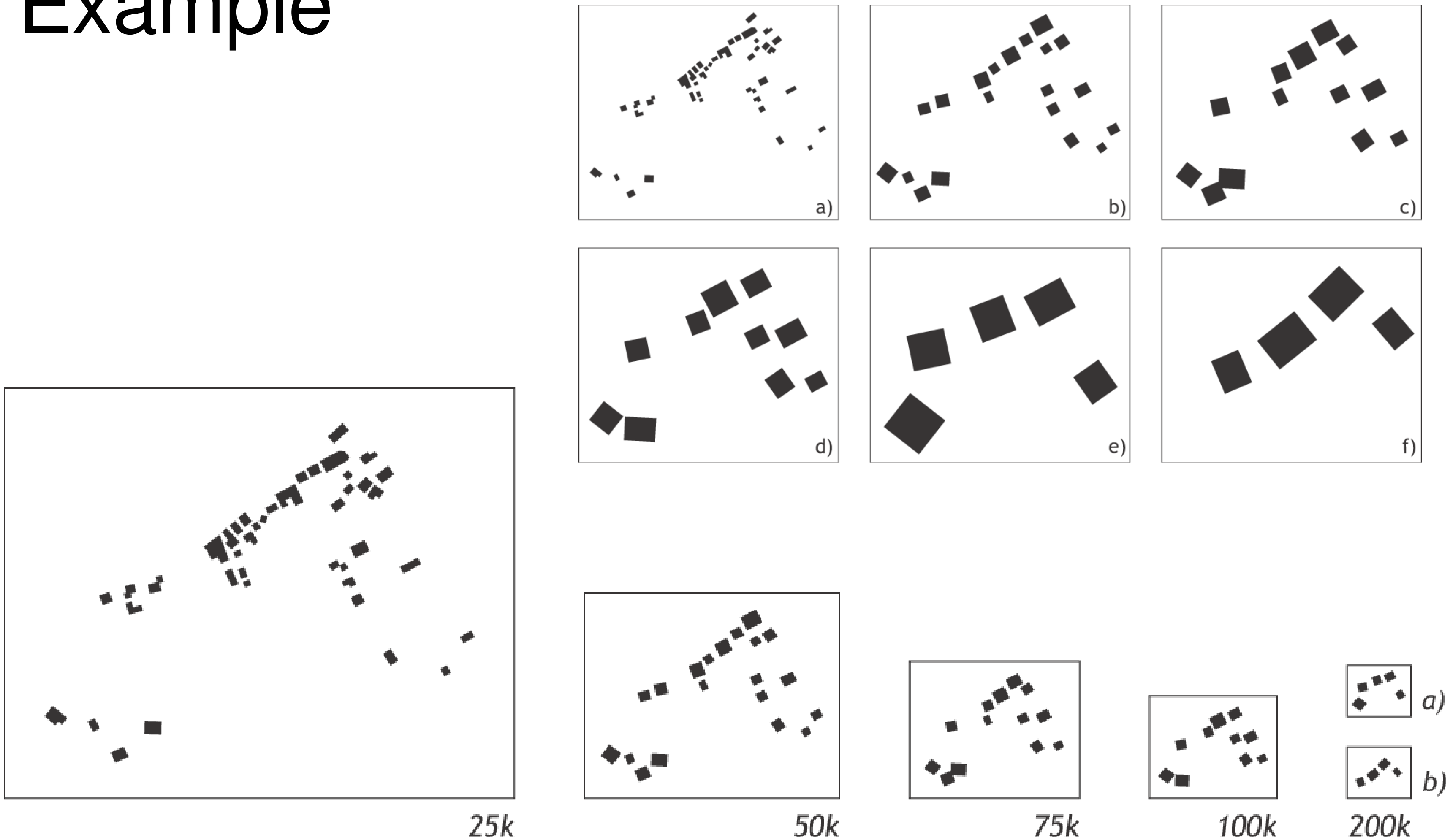
b) Orientation α

- orientation α of the new building is taken from the largest object of the group

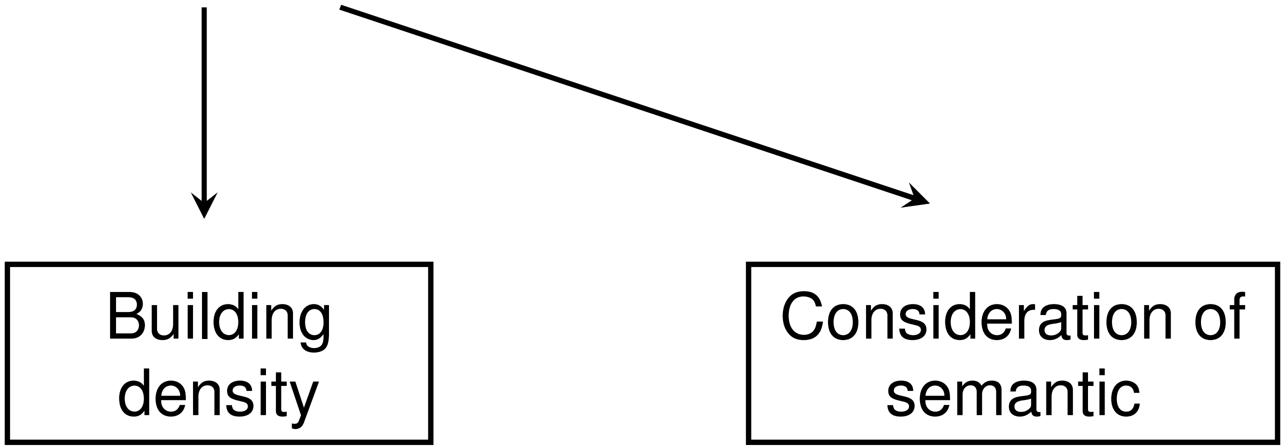
Example



Example



Control parameter



Control parameter – building density

- areas with high building density would be thinned out more strongly than sparsely populated regions – because approach is based on looking for the shortest distance

- correction factor allows to elongate the real edge length *a* between the vertices –
decrease the thinning process in densely populated areas

Control parameter

$$a_{min} = \min a, \quad \text{where} \quad a = f_a \cdot \sqrt{(x_s - x_t)^2 + (y_s - y_t)^2}$$

$$f_a = s_u \cdot (r - 1) + 1$$

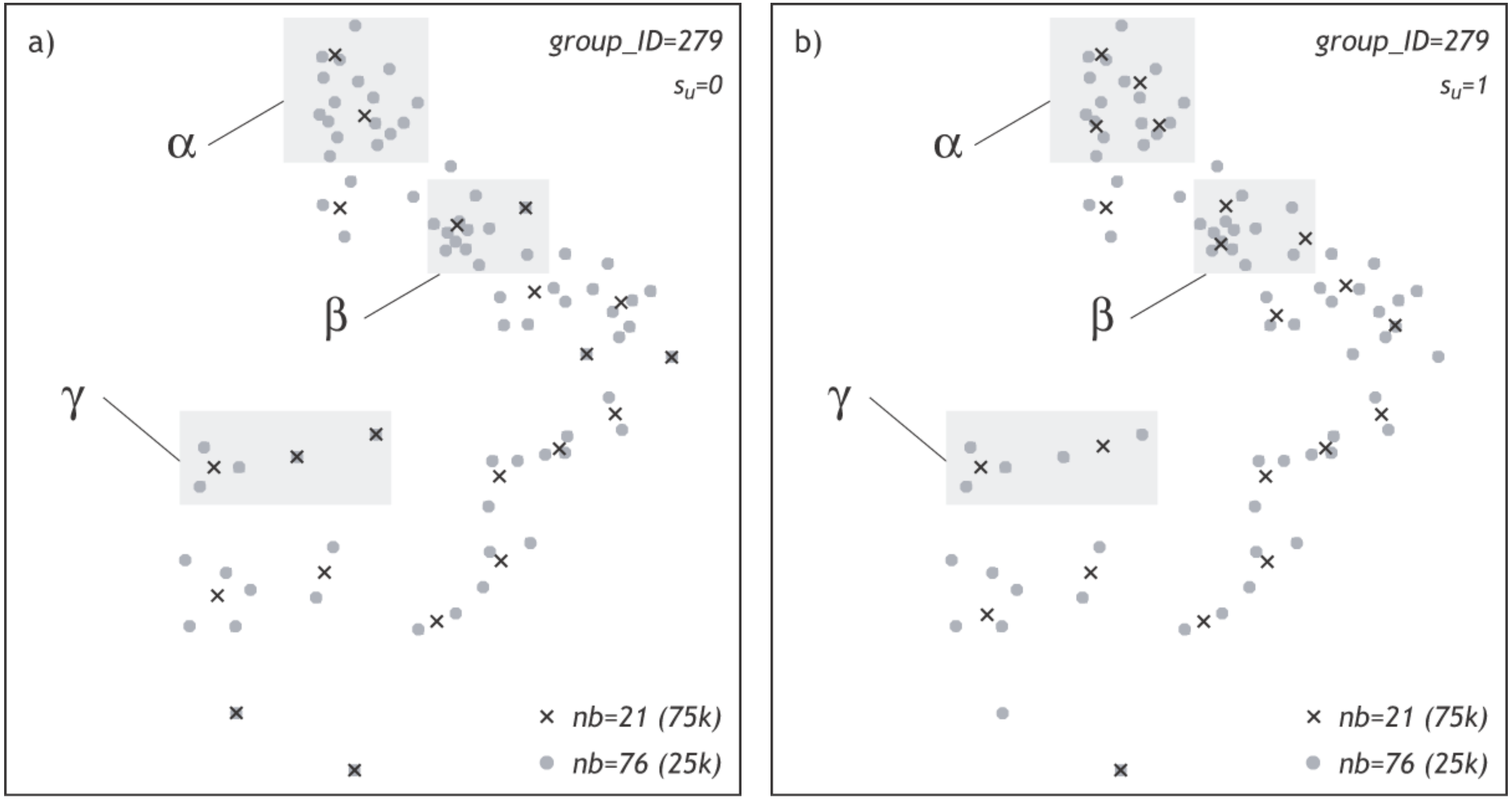
r - number of objects that a placeholder stands for

$0 \le s_u \le 1$ set by user

$s_u = 0.0$ no correction is done

$s_u = 1.0$ correction by means of the original number of represented objects

Control parameter



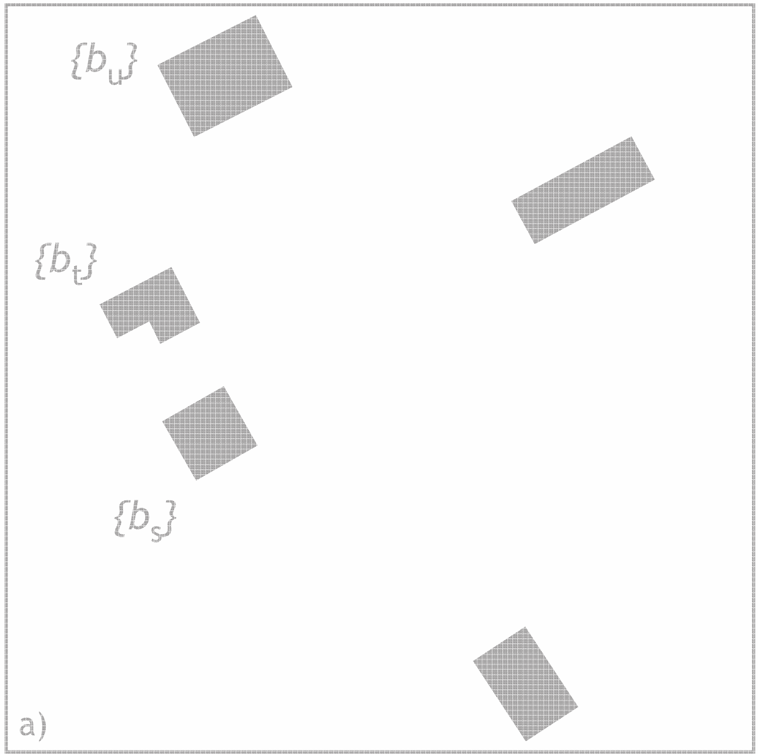
Control parameter – semantic S, size A

during positioning of the placeholder the size of areas (A) and the semantic of objects (S) not considered

Idea: use weighted points

$$x = \frac{\sum p_i x_i}{\sum p_i} \qquad y = \frac{\sum p_i y_i}{\sum p_i}$$

$$p_i = \lambda A_i + \kappa S_i$$



Control parameter – semantic S, size A

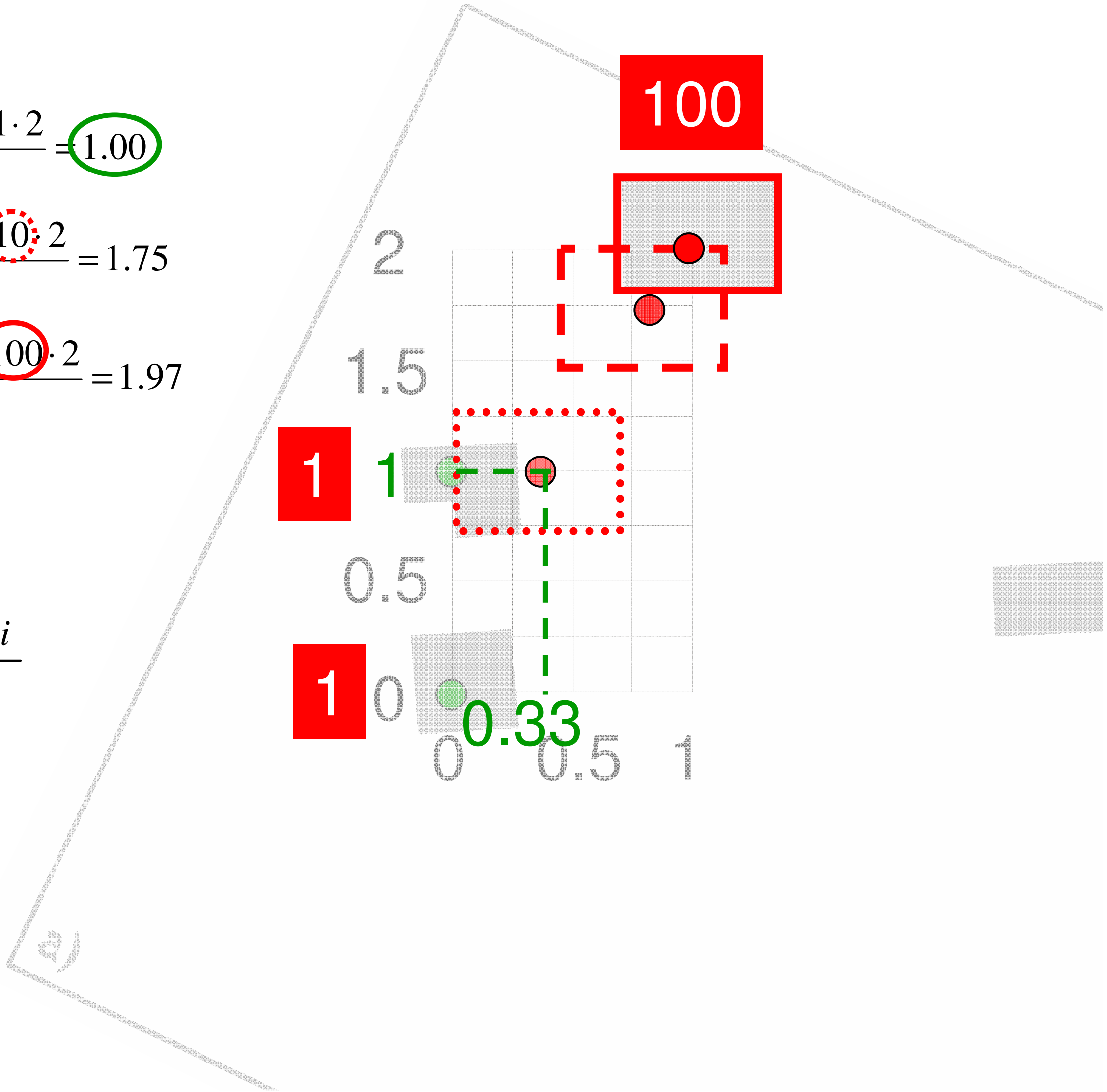
$$x_1 = \frac{1\cdot 0 + 1\cdot 0 + 1\cdot 1}{3} = 0.33 \qquad y_1 = \frac{1\cdot 0 + 1\cdot 1 + 1\cdot 2}{3} = 1.00$$

$$x_2 = \frac{1\cdot 0 + 1\cdot 0 + 10\cdot 1}{12} = 0.83 \qquad y_2 = \frac{1\cdot 0 + 1\cdot 1 + 10\cdot 2}{12} = 1.75$$

$$x_3 = \frac{1\cdot 0 + 1\cdot 0 + 100\cdot 1}{102} = 0.98 \qquad y_3 = \frac{1\cdot 0 + 1\cdot 1 + 100\cdot 2}{102} = 1.97$$

$$x = \frac{\sum p_i x_i}{\sum p_i} \qquad y = \frac{\sum p_i y_i}{\sum p_i}$$

$$p_i = \lambda A_i + \kappa S_i$$



Limitations and possible improvements

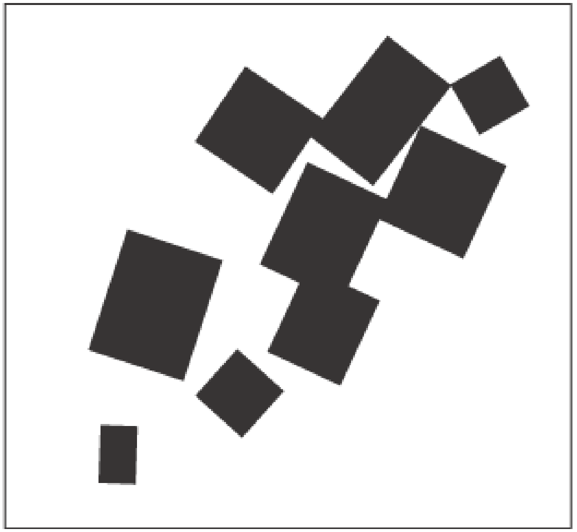
- mesh simplification technique fails to maintain alignments of buildings
- during positioning of the placeholder the size of areas and their semantic not considered
- the minimum distances are taken into account only indirectly with considering two scales

- such patterns could be preserved by introducing additional energy term to the method
- with weighted points a concept for a solution is suggested
- include additional constraint or solve the problem with following operation (shrinking, displacement)

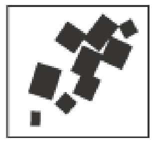
Limitations and possible improvements

Figures show an example of overlaps created by excessive building sizes

a)



b)



c)

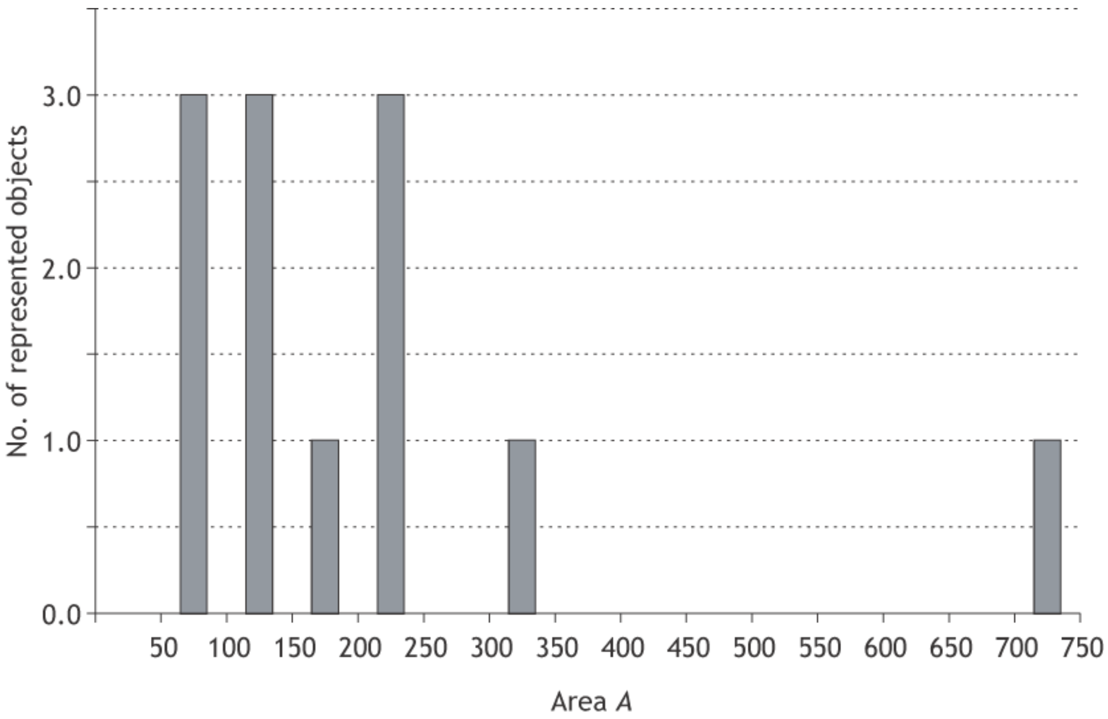
Limitations and possible improvements

- mesh simplification technique fails to maintain alignments of buildings
- during positioning of the placeholder the size of areas and their semantic not considered
- the minimum distances are taken into account only indirectly with considering two scales
- if a placeholder represents buildings with extremely different sizes the average size is displayed

- such patterns could be preserved by introducing additional energy term to the method
- with weighted points a concept for a solution is suggested
- include additional constraint or solve the problem with following operation (shrinking, displacement)
- a statistical evaluation of the represented building objects could be accomplished

Limitations and possible improvements

Figure shows a histogram for a placeholder



University of Zurich

Department of Geography

Dr. Dirk Burghardt

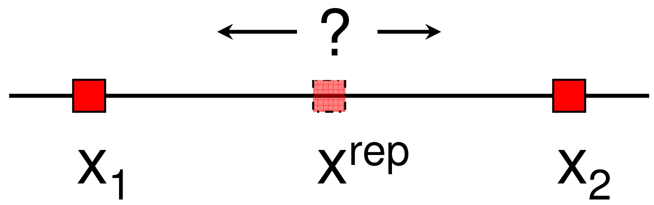
Tel.: +41 1 63 56534

burg@geo.unizh.ch

Typification

Derivation of equation

$$\left\| \underset{(n+e,m)}{\mathbf{M}} \cdot \underset{(m,1)}{\mathbf{x}^1} - \underset{(n+e,1)}{\mathbf{d}^1} \right\|^2$$



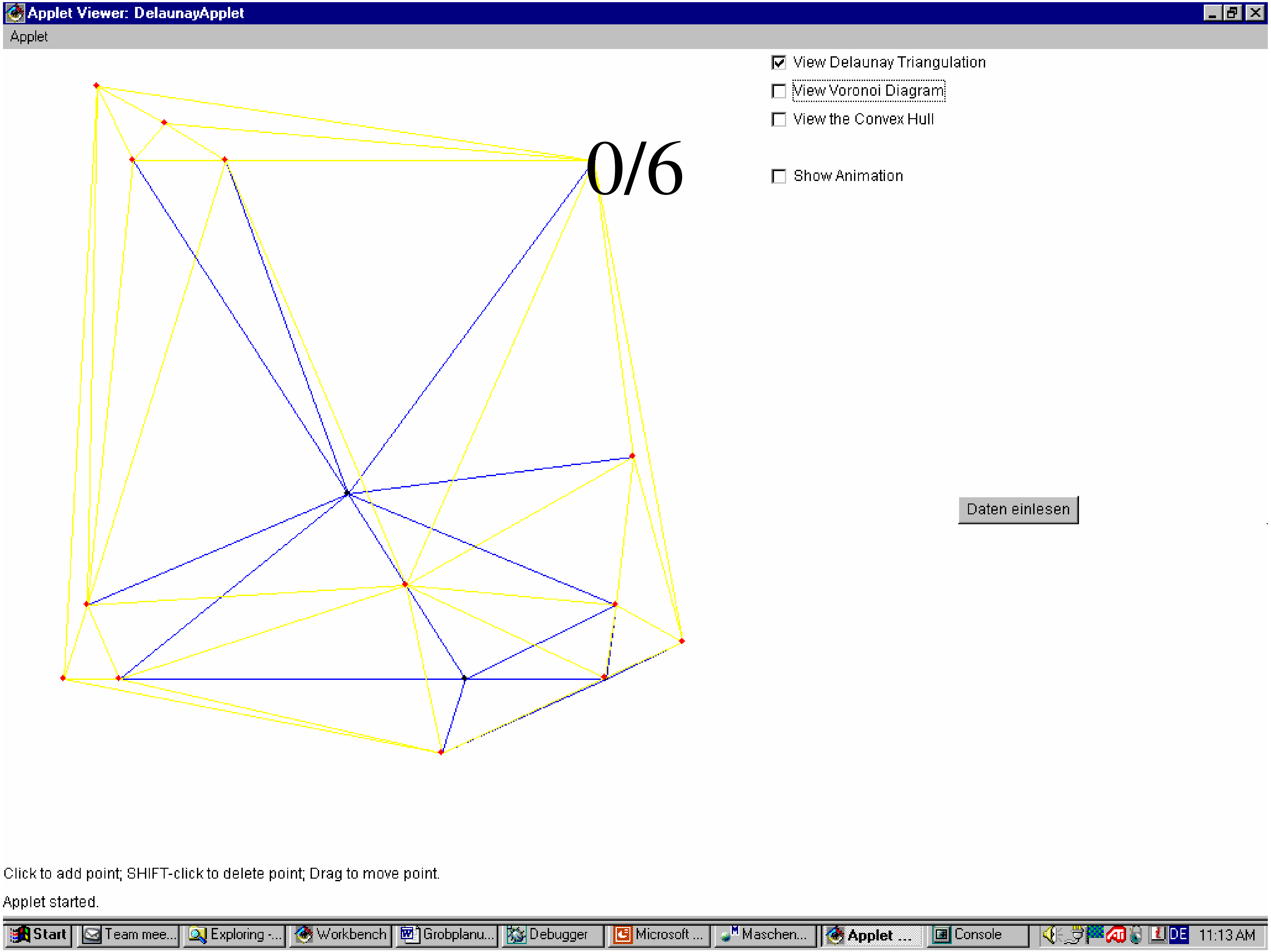
$$E_{node} = \frac{1}{2}\alpha \left[(x^{rep} - x_1)^2 + (x^{rep} - x_2)^2 \right] \longrightarrow \text{Min} \longrightarrow \delta E_{node} = 0$$

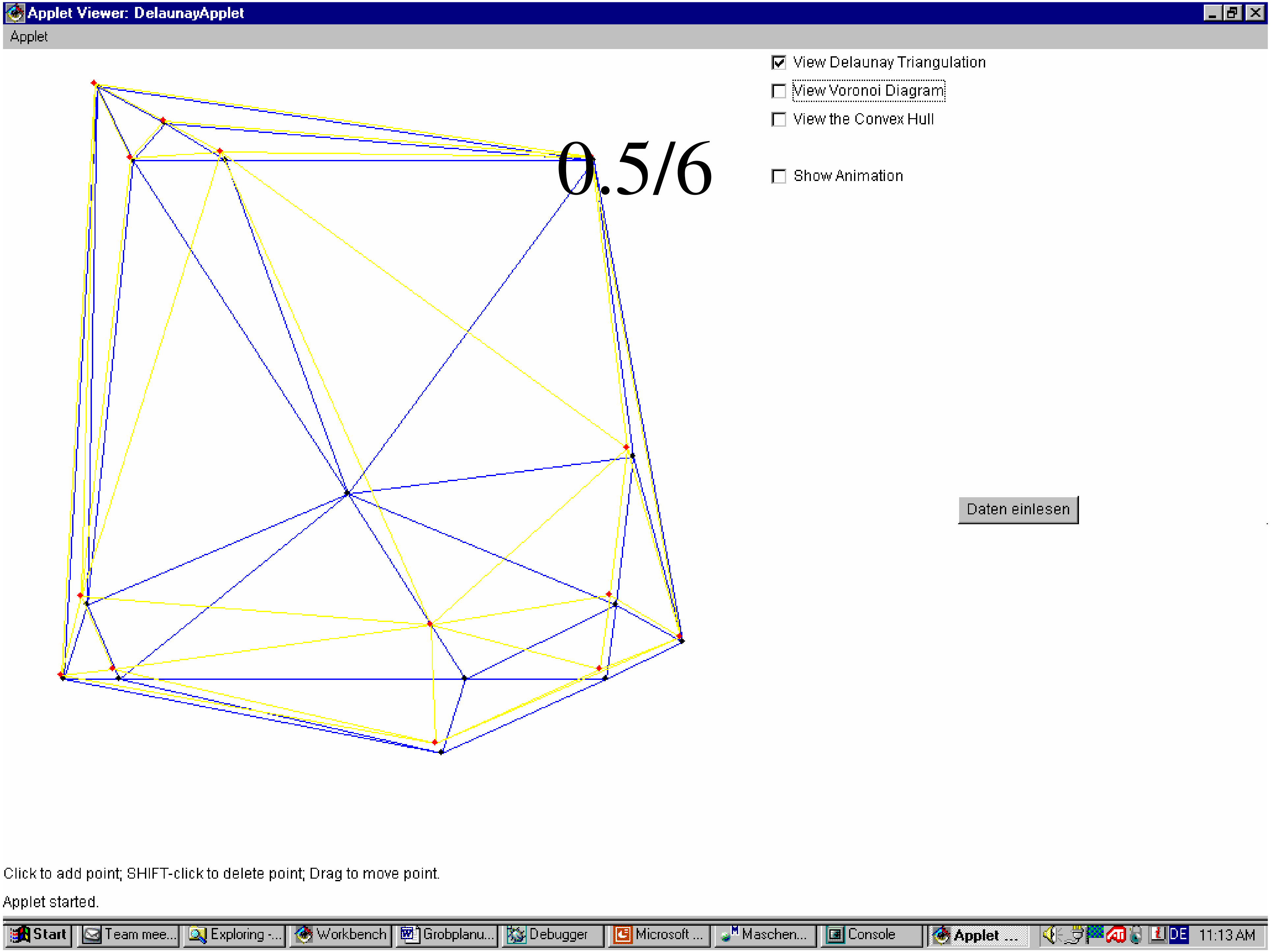
Variation of energy:

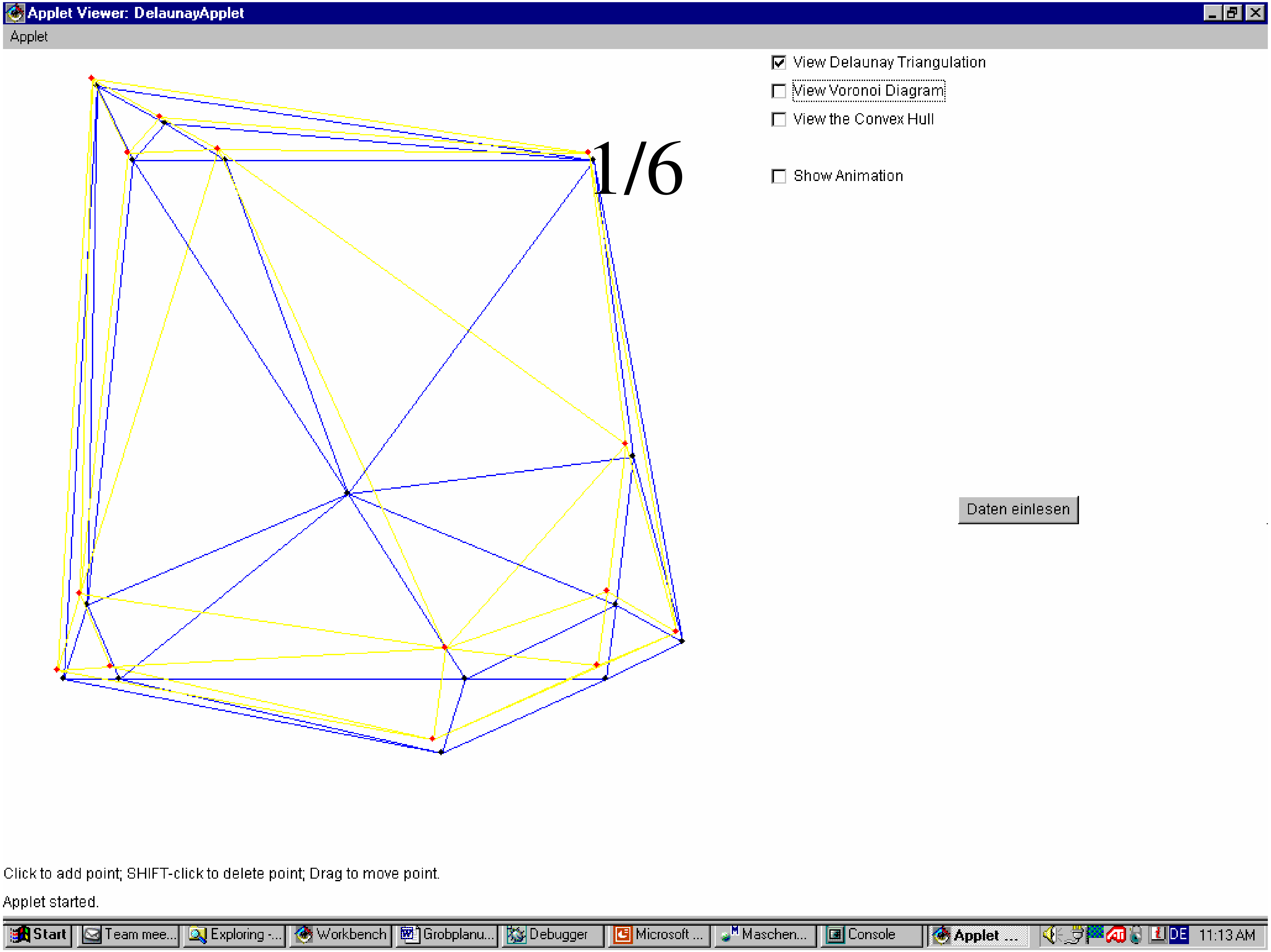
$$\delta E_{node} = \frac{1}{2}\alpha \frac{\partial}{\partial x^{rep}} \left[(x^{rep} - x_1)^2 + (x^{rep} - x_2)^2 \right] \delta x^{rep} = 0$$

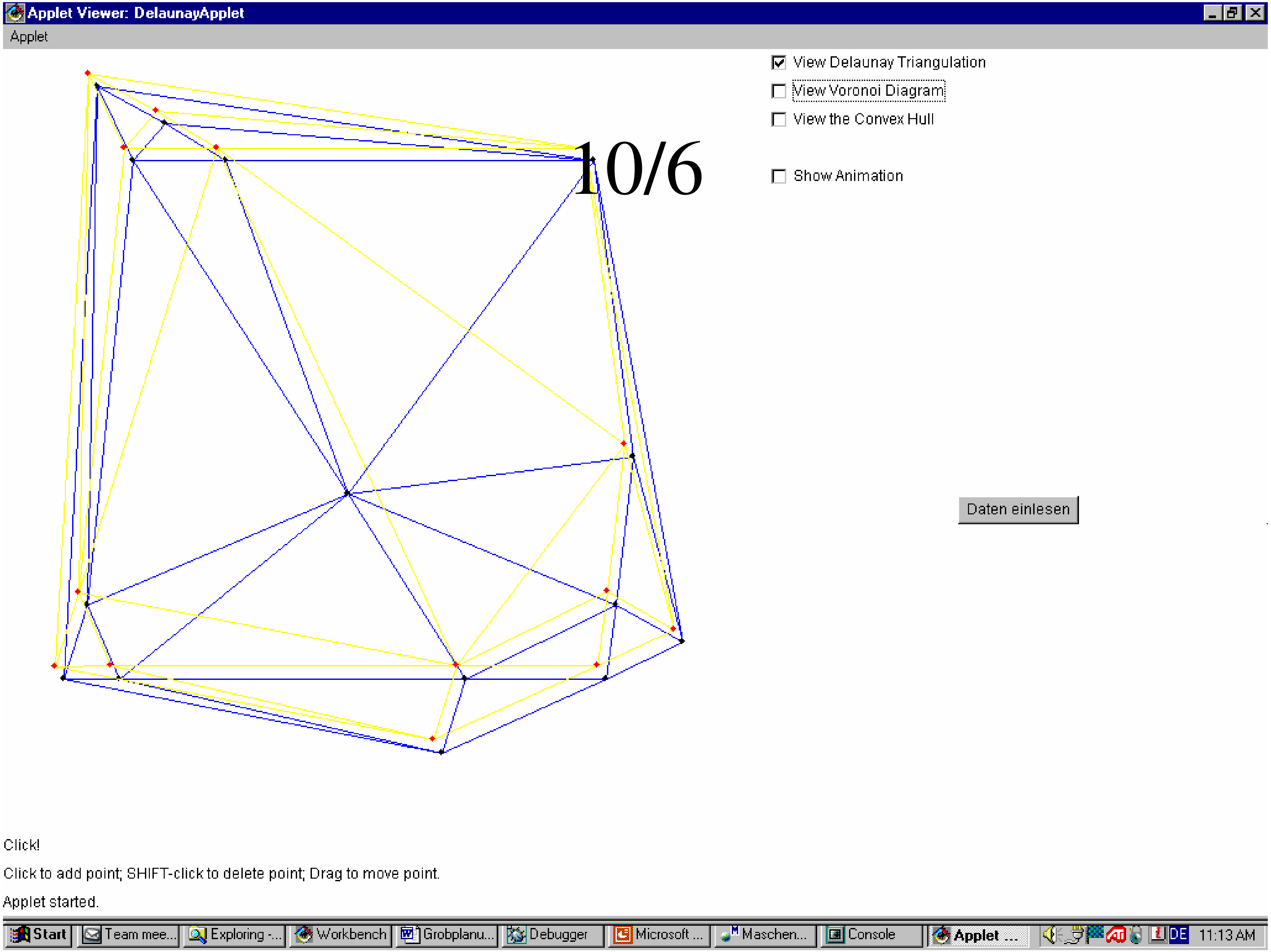
$$= \frac{1}{\not{2}} \cdot \not{2}\alpha \left[\underbrace{(x^{rep} - x_1) + (x^{rep} - x_2)}_{\begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x^{rep} \\ 0 \end{pmatrix} - \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}} \right] \delta x^{rep} = 0$$

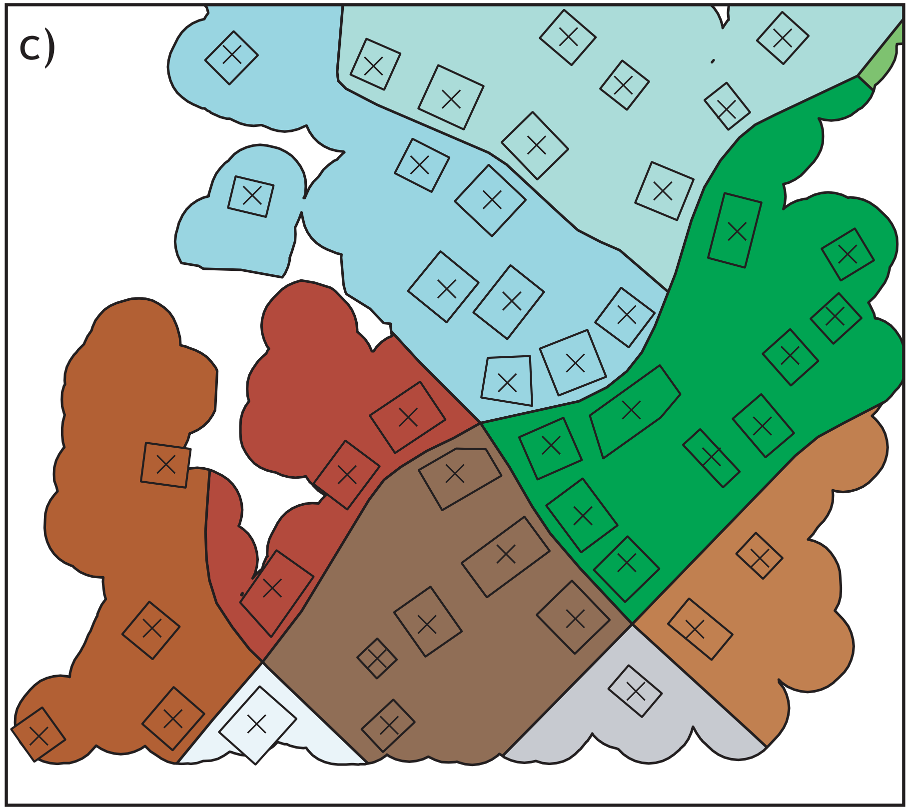
M x - d


Applet Viewer: DelaunayApplet
Applet
0/6
View Delaunay Triangulation
View Voronoi Diagram
View the Convex Hull
Show Animation
Daten einlesen
Click to add point; SHIFT-click to delete point; Drag to move point.
Applet started.
Start
Team mee...
Exploring -...
Workbench
Grobplanu...
Debugger
Microsoft ...
Maschen...
Applet ...
Console
DE
11:13 AM


Applet Viewer: DelaunayApplet
Applet
0.5/6
View Delaunay Triangulation
View Voronoi Diagram
View the Convex Hull
Show Animation
Daten einlesen
Click to add point; SHIFT-click to delete point; Drag to move point.
Applet started.
Start
Team mee...
Exploring -...
Workbench
Grobplanu...
Debugger
Microsoft ...
Maschen...
Console
Applet ...
DE
11:13 AM


Applet Viewer: DelaunayApplet
Applet
1/6
View Delaunay Triangulation
View Voronoi Diagram
View the Convex Hull
Show Animation
Daten einlesen
Click to add point; SHIFT-click to delete point; Drag to move point.
Applet started.
Start
Team mee...
Exploring -...
Workbench
Grobplanu...
Debugger
Microsoft ...
Maschen...
Console
Applet ...
DE
11:13 AM


Applet Viewer: DelaunayApplet
Applet
10/6
View Delaunay Triangulation
View Voronoi Diagram
View the Convex Hull
Show Animation
Daten einlesen
Click!
Click to add point; SHIFT-click to delete point; Drag to move point.
Applet started.
Start
Team mee...
Exploring -...
Workbench
Grobplanu...
Debugger
Microsoft ...
Maschen...
Console
Applet ...
DE
11:13 AM


c)