Constrained tetrahedral models and update algorithms for 3D topography

Friso Penninga
Delft University of Technology
OTB, section GIS Technology
the Netherlands
F.Penninga@otb.tudelft.nl

Seminair Geo-Information and Computational Geometry

TU Delft

Delft University of Technology

Presentation outline

- Introduction
- 3D modelling in GIS
  - Selection of 3D primitive
  - Modelling concept
  - Implementation in a spatial DBMS
- 3D Topogaphy and Computational Geometry
  - Amount of data
  - Updating the model: requirements from a CG perspective
- Conclusions and further research
- Discussion

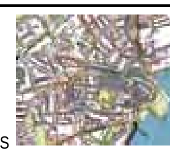
TU Delft

Introduction:
Backgrounds

- PhD-research, started spring 2004 (supervision Peter van Oosterom)
- Part of bigger research project BSIK-RGI '3D Topography', covering entire geo-information proces:
  3D data collection, storage, analysis, visualization
- 3D Topography consortium partners:
  Delft University of Technology, ITC Enschede, Dutch Topographic Survey, Rijkswaterstaat, NedGraphics, Oracle

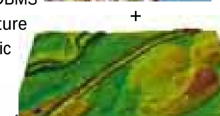
TU Delft

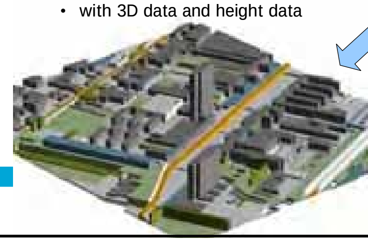
Introduction:
Research goal



Create a 3D topographic data model in a DBMS
- using a newly developed data structure
- by combining current 2D topographic data sets
- with 3D data and height data

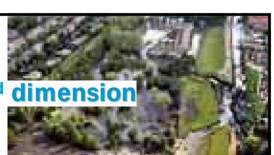
+





TU Delft

Introduction:
The need for the 3rd dimension

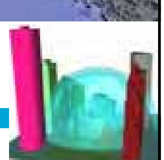


Real world consists of 3D objects
- Objects become more complex: multiple land use

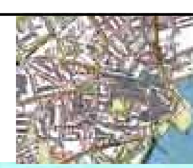
Rising awareness of importance sustainable urban environments
- Requires 3D planning / modelling

Need for better data for emergency services and (natural and non-natural) disaster response

Requires 3D Topography: not only visualization!

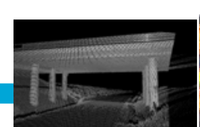


Introduction:
Available data sources



Topography (Top10NL):
- 2D (some height information in symbology)
- Conversion to object-oriented format

Heigth data sources:
- AHN (laserscan data) 1 point/ 16 m^2
- DTB's: topography along rivers + highways
- Terrestrial laserscanning



TU Delft

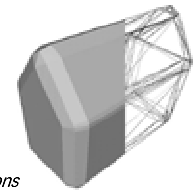
Presentation outline

- Introduction
- 3D modelling in GIS
  - Selection of 3D primitive
  - Modelling concept
  - Implementation in a spatial DBMS
- 3D Topography and Computational Geometry
  - Amount of data
  - Updating the model: requirements from a CG perspective
- Conclusions and further research
- Discussion

3D modelling in GIS:
Selection of a 3D primitive

So focus is on 3D analyses
These are best supported by models with 3D primitives (volumes)

Different approaches:
- simplexes (Carlson)
- 3D FDS (Molenaar)
- CSG+B-rep (de Cambray)
- 2.5D TIN + 3D FDS (Pilouk)

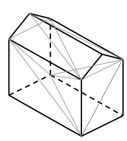


Selected in this research: 0D-3D simplexes
TEN: nodes, edges, triangles, tetrahedrons

3D modelling in GIS:
Selection of a 3D primitive

Advantages simplexes:
- Well defined: a kD-simplex is bounded by k+1 (k-1)D-simplexes
  a 2D-simplex (triangle) is bounded by 3 1D-simplexes (edges)
- Flatness of faces: every face can be described by three points
- A kD-simplex is convex (simplifies point-in-polygon tests)

Disadvantage simplexes
- Increasing complexity: a 1:n relationship between topographic features and their representations (set of simplexes)



3D modelling in GIS:
Modelling concept

Two fundamental observations:

- ISO19101: a feature is an 'abstraction of real world phenomena'. These real world phenomena have by definition a volume
- Real world can be considered to be a volume partition (analogous to a planar partition: a set of non-overlapping volumes that form a closed modelled space)

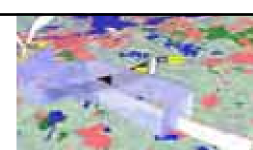
3D modelling in GIS:
Modelling concept

'Abstraction of real world phenomena' (ISO19101):
- until now:
  abstraction (simplification) ⟶ less dimensional representation
- when using meshes:
  simplification is in subdivision into easy-to-handle parts (analogously to Finite Element Method for solving Partial Differential Equations)

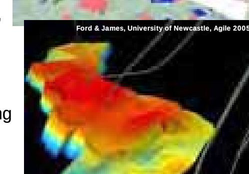
So: model only volume features in a volume partition

But allow polygon features:
faces mark boundary between two volumes (e.g. walls),
so polygon features are derived from volumes (associaton class)

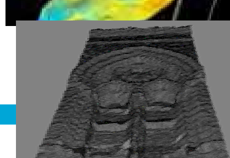
3D modelling in GIS:
Modelling concept



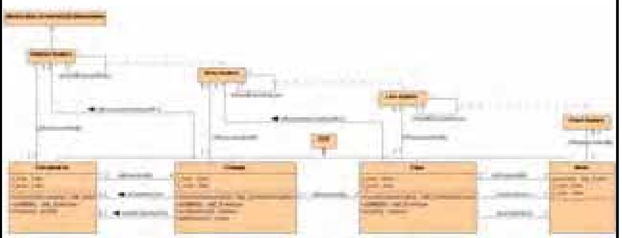
Rigid approach ⟶ full volume partition, including earth and air tetrahedrons



Advantages:
- Air often subject of analysis: modelling noise, odour, air polution
- Model extensible: e.g. air traffic corridors, geology, petroleum reservoirs
- Indoor topography / high level of detail in buildings (terrestrial laser scanning)



3D modelling in GIS:
Conceptual model

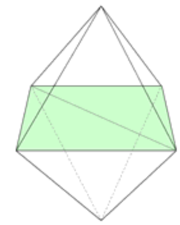


TUDelft

3D modelling in GIS:
Implementation in spatial DBMS

Four step approach:

1. Start with four initial tetrahedrons: two air, two earth
2. Refine earth surface by inserting height information from a DEM
3. Refine air and earth tetrahedrons in case of ill-shaped tetrahedrons
4. Add real topographic features (in tetrahedronized form)



TUDelft

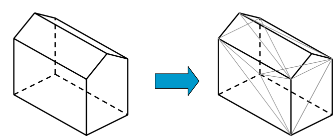
Presentation outline

- Introduction
- 3D modelling in GIS
  - Selection of 3D primitive
  - Modelling concept
  - Implementation in a spatial DBMS
- 3D Topogaphy and Computational Geometry
  - Amount of data
  - Updating the model: requirements from a CG perspective
- Conclusions and futher research
- Discussion

TUDelft

3D Topography and
Computational Geometry

Amount of data:



Building as polyhedron	Building as TEN
(1 volume)	8 tetrahedrons
7 faces	24 triangles
(15 edges)	25 edges
(10 points)	10 nodes

TUDelft

3D Topo & Computational Geometry
Updating: CG requirements

Case: a model is stored in the DBMS

```
create table node(
   nid integer,
   geom sdo_geometry);

create table edge(
   eid integer,
   startnode integer,
   endnode integer,
   isconstraint integer);

create table triangle(
   trid integer,
   edge1 integer,
   edge2 integer,
   edge3 integer,
   isconstraint integer,
   afid integer);

create table tetrahedron(
        tetid integer,
        triangle1 integer,
        triangle2 integer,
        triangle3 integer,
        triangle4 integer,
        vfid integer);

create table areafeature(
        afid integer,
        type varchar2(30));

create table volumefeature(
        vfid integer,
        type varchar2(30));
```

TUDelft

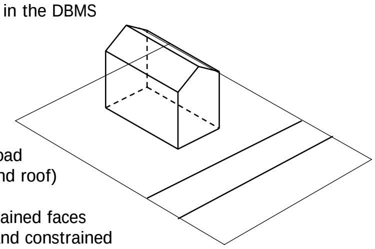
3D Topo & Computational Geometry
Updating: CG requirements

Case: a model is stored in the DBMS

Tetrahedrons represent the house, the air and the earth

Triangles represent the earth surface and the road (eventually also walls and roof)

In order to do so constrained faces bound some volumes, and constrained edges bound some faces



However: triangulation / tetrahedronization algorithms only work with constrained egdes

TUDelft

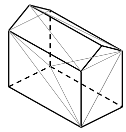
3D Topo & Computational Geometry
Tetrahedronization algorithms

Tetrahedronization algorithms:
two-step approach

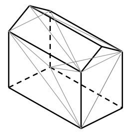
First step:
Create surface triangulation

Second step:
Create tetrahedrons in interior

All edges from surface triangulation are handled as constrained edges, the surface triangles are labelled as constraint



1



2

3D Topo & Computational Geometry
Case: updating procedure

Case: remove the building from the data set

On feature level
- Remove record from table volume feature

On TEN level
- Find relevant constrained egdes and remove constraints
- Find relevant constrained faces and remove constraints
- Simplify TEN by creating larger tetrahedrons if possible
- Optimize triangle/tetrahedron shape if necessary

On feature level
- Re-classify new tetrahedrons

3D Topo & Computational Geometry
CG algorithm requirements

- Creating / maintaining the TEN: incremental tetrahedronization
- Due to amount of data algorithm should work within database
- In database: preferably updating impacts TEN structure as local as possible
- Efficient TEN storage is required, but working with implicit triangles or edges is not an option
  (as tetrahedrons represenent features,
  triangles contain important topological information,
  edges contain constraints and nodes contain geometry)
- Numerically stable: detection and repair of ill-shaped triangles and tetrahedrons (Shewchuck: mesh refinement)

Presentation outline

- Introduction
- 3D modelling in GIS
  - Selection of 3D primitive
  - Modelling concept
  - Implementation in a spatial DBMS
- 3D Topogaphy and Computational Geometry
  - Amount of data
  - Updating the model: requirements from a CG perspective
- Conclusions and further research
- Discussion

Conclusions

- Tetrahedronizations offer good computational / analytical capabilities
- TEN approach will lead to very large data sets
- Fast and reliable algorithms are required
- Constrained TEN incremental algorithm required
- Mesh refinement to ensure numerically stable operations

Despite conceptual advantages of 3D TEN approach, the succes of the TEN approach completely depends on the degree in which algorithms are capable of querying, analyzing and altering the structure with acceptable performance

Further research

- Implement required algorithms for constrained triangulation and tetrahedronization
- Implement within the DBMS an incremental algorithm for tetrahedronization
- Perform real tests with real world features
- Compare results to other approaches

Discussion