Height in the Cadastre
Integrating Point Heights and Parcel Boundaries

Jantien STOTER and Ben GORTE, the Netherlands

Key words: 3D cadastre, 3D data models, geo-DBMS, DTM.

SUMMARY

In the research concerning 3D cadastre (Stoter et al., 2002) a prototype has been built that improves insight in the vertical dimension of rights registered in the cadastral registration system. This is important when different properties are located above each other (tunnels, pipelines, building complexes). In the prototype, parcel boundaries need to be located in 3D space as they have to be combined with 3D objects such as tunnels and pipelines. To extend the spatial model of parcel boundaries into 3D, simply assigning one z-coordinate to each parcel is not sufficient. Providing a z-coordinate to the vertices describing parcel boundaries does also not meet the requirements for a 3D cadastre. In this paper we elaborate on how parcel boundaries can be integrated with point heights to meet the requirements for a 3D cadastre, i.e. obtaining a height surface for parcels. A height surface of parcels is the digital terrain model (DTM) on the locations of parcels. Therefore a parcel surface changes when the terrain surface changes. We will describe and evaluate several possibilities to obtain a height surface for parcels based on a DBMS approach.

Height in the Cadastre
Integrating Point Heights and Parcel Boundaries

Jantien STOTER and Ben GORTE, the Netherlands

1. INTRODUCTION

According to FIG (1995) a cadastre is "a parcel-based, and up-to-date land information system containing records of interests in land (rights, restrictions and responsibilities)". Cadastral systems maintain rights and limited rights as attribute data on parcels that are described geometrically. In those systems parcel boundaries are maintained in 2D.
In the research concerning 3D cadastre (Stoter and Ploeger, 2002) a prototype has been built in which 3D physical objects are modelled in a DBMS. In the prototype rights established for 3D physical objects are also defined in 3D.
An important question was how the z-coordinates of these 3D objects should be defined: with absolute values (in the national coordinate frame) or relatively, with respect to the surface (e.g. 6 meter below the surface). Absolute z-coordinates are not influenced by surface changes, furthermore the definition of the surface level (the reference level used for values with respect to the surface) is sometimes not clear, finally when using z-coordinates with respect to the surface it is complicated to define the actual geometry of 3D objects. Therefore the most sustainable solution is to define 3D objects with absolute z-coordinates, especially in rural areas. In some urban areas it seems sufficient to define z-coordinates with respect to the surface because surface levels are flat and because DTM's based on laserscan data are easier to generate in rural areas than in urban areas (see figure 1).

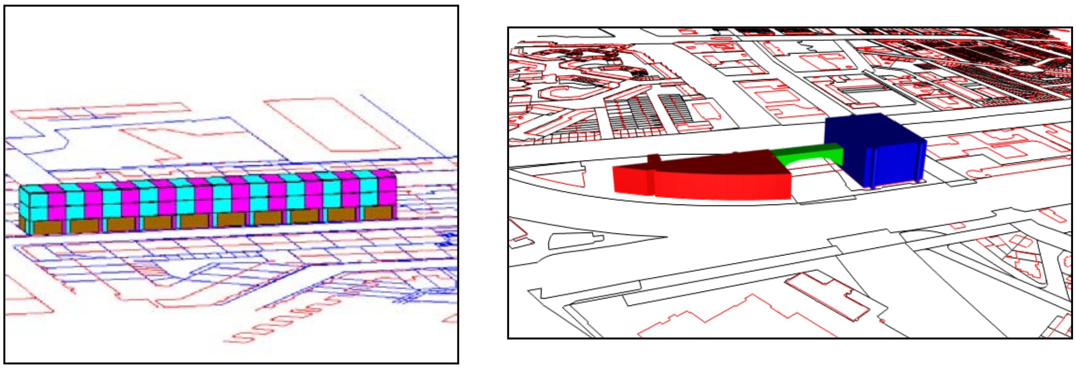

Figure 1: The heights of the buildings in these figures are defined with respect to the surface. Since surface level in urban areas is relatively 'flat', this is sometimes sufficient for representing the 3D situation.

In the prototype two case studies were carried out in rural area: 1) two pipelines and 2) a railway tunnel (Stoter and Ploeger, 2003). These 3D objects are known in absolute z-coordinates (in the Netherlands National Ordnance Datum: NAP). Those 3D descriptions do not reveal where the 3D objects are located with respect to the surface and with respect to the parcels on the surface: are the 3D objects situated above or under the ground, what is the depth of the tunnel and pipelines?

To know the vertical position of 3D objects, the surface of the parcels is also needed. Obtaining the height surface of parcels by the integration of parcel boundaries and point heights is the topic of this paper. Lenk (2001) already performed a study on the integration of 2D GIS data and height information.

Using one z-coordinate for each parcel is not sufficient. Assigning height to the nodes describing parcel boundaries is also not sufficient, because with this only the height is known on the location of the parcel boundary and not within the parcel itself. In this research we look for a solution where the surface covering a parcel is modelled in 2.5D.

First we use an example to illustrate why parcel surfaces are needed in the 3D cadastre, followed by a description of the datasets used. Then the TIN's (Triangular Irregular Network) generated in this research are described: an unconstrained and a constrained TIN, together with the data structures. The actual integration of height information and parcel boundaries is explained by describing how a parcel surface can be extracted from the DBMS. We end with conclusions.

2. INTEGRATING HEIGHT DATA AND PARCEL BOUNDARIES

For a 3D cadastre we are interested in the combination of parcel boundaries and height data in order to obtain a parcel surface that can be combined with 3D objects (such as tunnel, cables, pipelines). This indicates where the 3D object is located with respect to the surface level.

To illustrate this, a pipeline is examined that was used as case study in the 3D cadastre research. In this case study the NAM (Nederlandse Aardolie Maatschappij: company owning an important part of the network of natural gas in the Netherlands) provided us with 3D information on a pipeline located in the study area (see figure 2).

Figure 2 is a combination of 2D parcel boundaries and a 3D pipeline (defined with absolute z-coordinates). It does not show where the pipeline is positioned with respect to the surface: in this specific case the 3D pipeline (which has absolute z-coordinates between 5 and 10 meter) appears above the surface (the 2D parcel boundaries are positioned in the z=0 plane).
In figure 2, on the right the dashed lined shows the projection of the 3D pipeline on the plane where the z-coordinates equal zero (the plane where the 2D parcel boundaries are positioned), which shows that the 3D pipeline is located (5 to 10 meter) above the parcel boundaries: this is not correct since it is an underground pipeline.

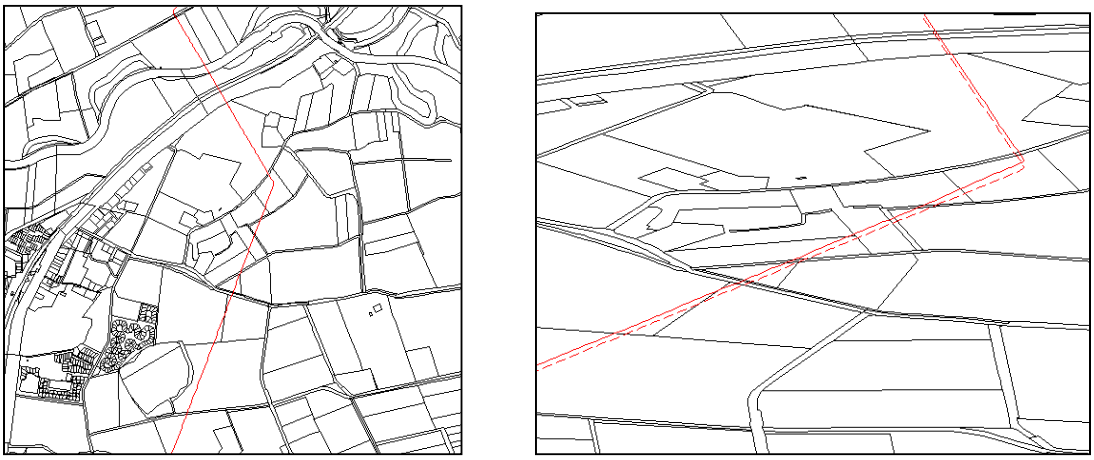

Figure 2: A pipeline defined in 3D combined with 2D parcels; the dashed line in the figure on the right is the projection of the pipeline on the plane where the z-coordinate equals zero (which is the plane where the 2D parcel boundaries are positioned)

We described the parcel boundaries in 3D, by assigning z-coordinates to the nodes of the parcel boundaries. A DTM (Digital Terrain Model) represented by an unconstrained TIN of laser altimetry points was used for the extraction of the z-coordinates (see further). The 3D parcel boundaries are drawn with dashed lines in figure 3.

On the locations of the parcel boundaries it is now possible to determine what the depth (or height) is of the pipeline. However, within the surface of one parcel this still is not clear. Therefore the parcel surface needs to be obtained. When the height surface of a parcel is known, this information can also be used to find the actual surface of the parcel (instead of the projected surface which is currently registered in the cadastral system).

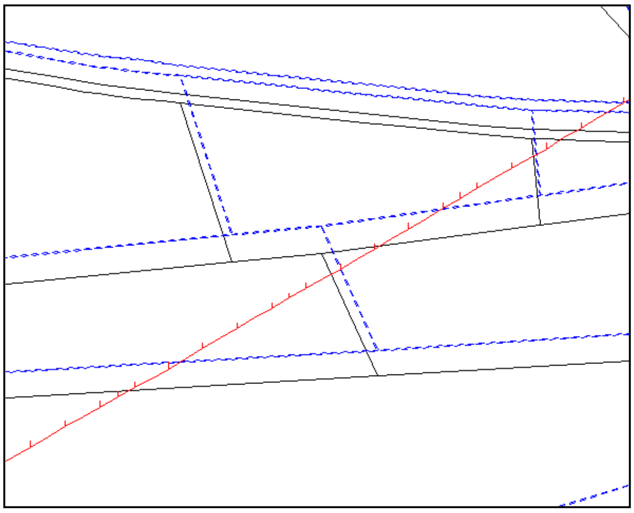

Figure 3: Parcel boundaries defined in 3D (dashed lines) give insight in where the pipeline is positioned with respect to the surface. Sticks indicate the distance between the pipeline and the surface.

3. DESCRIPTION OF DATASETS AND THEIR DATA STRUCTURE

For the integration of height data and cadastral data, two types of registration are used: the cadastral data maintained by the Netherlands' Kadaster and height data maintained by Rijkswaterstaat (Dutch Ministry of Transport and Public Works). In this research we combine these registrations by using a DBMS approach.

3.1 Terrain model

For the terrain model we use a dataset representing the DTM of the Netherlands, i.e. AHN (Actueel Hoogtebestand Nederland) (Van Heerd et al., 2000). The AHN is a dataset of points with heights obtained with laser altimetry with a density of one point per 25 square meters (with a validation accuracy of 5 cm). The dataset used is a selection of the AHN covering an area of 5 km by 6 km, consisting of 1198029 data points.

3.2 Parcel boundaries

The used parcel boundaries are a selection of the cadastral database of the Netherlands. This selection contains 2225 parcels (the whole country consists of 6.4 million parcels). In the cadastral DBMS parcel boundaries are organised in the geometrical model and parcels are topologically stored according to the rules of the wing-edged data structure (Oosterom and Lemmen, 2001). Every parcel contains a reference to the first edge. The realisation of the polygons can also be performed in the DBMS (Oosterom et al., 2002).

4. GENERATING TIN FOR THE INTEGRATION OF POINT HEIGHTS AND PARCEL BOUNDARIES

An initial requirement is that all data is maintained in a DBMS. The main reason for this is that the cadastre maintains huge amounts of data, both geometrical data (consisting of 17,642,437 parcels and 45,868,700 boundaries for which history is also maintained) and attribute data (rights, restrictions, subjects). In this research we use therefore a DBMS: Oracle Spatial 9i (Oracle, 2001) to store the datasets. Also the extraction of the parcel surfaces is performed in the DBMS.

4.1 Unconstrained TIN

To be able to combine parcel boundaries with the point heights, first a TIN (Triangular Irregular Network) was generated using only the point data, by means of triangulation software called Triangle (Shewchuk, 1996). The TIN is generated with Delauny triangulation (Worboys, 1995) and consists of 2393676 triangles and 1198029 nodes. The triangulation was performed outside the DBMS. The resulting TIN (containing x-,y-,z-coordinates on point data and the TIN defined with references to those points) was inserted in the DBMS in a topological model. The UML model of the TIN is shown in figure 4.

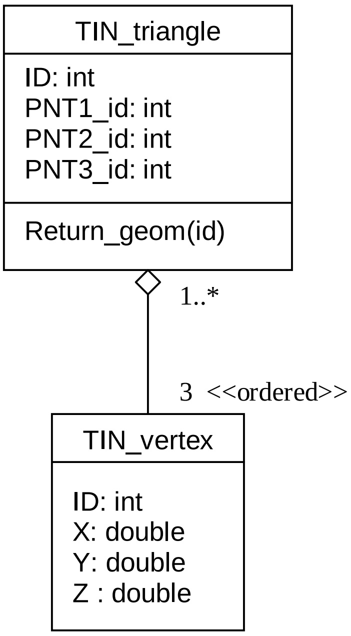

Figure 4: UML class diagram of TIN storage

In the topological model two tables are stored: one table contains references to the ids of the points for every triangle (three references for every triangle):

```
SQL> describe tin
 Name                                      Null?    Type
 ----------------------------------------- -------- ---------------
 ID                                                 NUMBER(8)
 PNT1_ID                                            NUMBER(8)
 PNT2_ID                                            NUMBER(8)
 PNT3_ID                                            NUMBER(8)
```

In the second table the coordinates of the points are stored together with their ids:

```
SQL> describe tin_vertex
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 ID                                                 NUMBER(8)
 X                                                  NUMBER(12,3)
 Y                                                  NUMBER(12,3)
 Z                                                  NUMBER(12,3)
```

In this way, every point is stored only once. A function has been written to generate ('realise') the geometry of the triangles (3D polygons) based on the topological tables. The function returns a 3D polygon of type mdsys.sdo_geometry (the spatial data type of Oracle). The geometry is stored as a view on the topological model by means of the following function:

```
create view tin_geom as select id, return_geom(id)shape from tin_c;
```

An advantage of having the geometries is that geometries are recognized by CAD/GIS software that can make a connection to the DBMS. In this way it is possible to visualise the data stored in the DBMS as is done in the figures in this paper.

4.2 Constrained TIN

In order to integrate the AHN with the parcel boundaries a constrained TIN was also generated, using the parcel boundaries as constraints. Again the Triangle software was used (outside the DBMS). In Triangle you can choose whether the edges that are constraints should be divided to make more optimal triangles or not, which means that the original parcel boundaries may exist as edges in the TIN. At first we did not divide the parcel boundaries in order to preserve the original boundaries in the TIN. To make optimal triangles the boundaries used for the constrained TIN can be divided based on criteria, e.g. avoid angles smaller than a threshold value, avoid areas larger than a threshold value. The created triangles near parcel boundaries did not give a good result (see figure 5, on the right), therefore in future research we will experiment with dividing the parcel boundaries in the triangulation process.

We assigned z-coordinates to the nodes of parcel boundaries by projecting them in the unconstrained TIN. Height values for the nodes were then calculated by interpolating the z-coordinates.

The constrained TIN contains 1500074 triangles and 783415 nodes and is again stored a topological model, with a geometrical view on top of it.

4.3 Optimal data structure

The ideal case is just storing the point heights and the parcel boundaries in the DBMS and to generate the TIN of the area of interest on user's request in the DBMS, without storing the TIN in the DBMS. This is more efficient because no data transfer (and conversion) is needed from DBMS to TIN software and back. This will be topic for future research.

5. EXTRACTING PARCEL SURFACE FROM THE DBMS

To obtain a parcel surface all triangles that are covered by one parcel need to be selected. This can be done by either the unconstrained or the constrained TIN (see figure 5).

In the unconstrained TIN (figure 5, left) the selection represents an area larger than the parcel itself since triangles cross parcel boundaries. In the constrained TIN (figure 5, right) each triangle belongs to exactly one parcel and therefore the selection of triangles exactly equals the area of a parcel, which offers better results

To compute the actual area in 3D space of a parcel, the constrained TIN is also a better option as can be concluded from figure 5. The actual area of a parcel in 3D can be computed by adding up all areas of the triangles covering one parcel.

Since the constrained TIN offers better possibilities, the query to extract a parcel surface from the DBMS is performed on the constrained TIN.

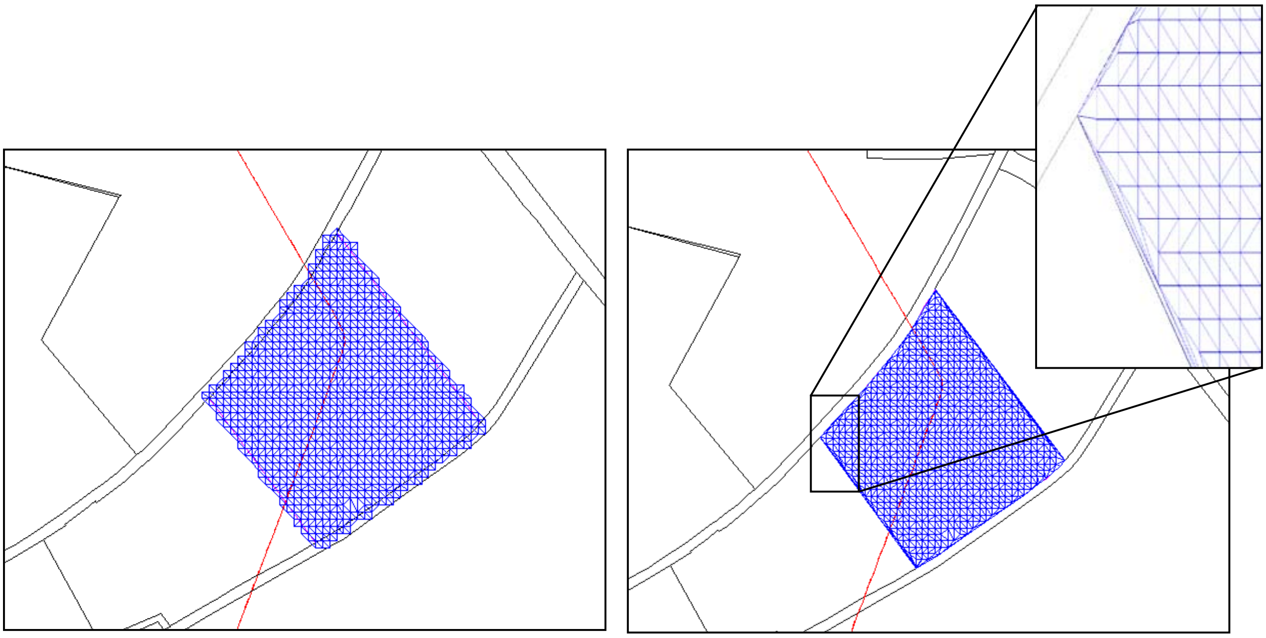

Figure 5: A parcel surface extracted form the DBMS on an unconstrained TIN (left) and based on a constrained TIN on the right. Note that triangles do not match parcel boundaries in the case of the unconstrained TIN.

Having the surface we now also can compute the position of the pipeline with respect to the surface (table 1). The values of "surface" were generated by projecting the pipeline in the TIN. The values for "with respect to the surface" could then be computed by subtracting the "z" value of the pipeline from the "surface" value.

```
x                 y          z           surface
wrt_surface
242847.22  512941.22  10.34  9.17208    1.16792
242847.21  512941.25  10.38  9.172072   1.207928
242847.17  512941.25  10.38  9.171944   1.208056
242845.24  512942.82  10.38  9.165616   1.214384
242844.80  512943.11  10.23  9.163164   1.066836
242843.01  512944.55   8.89  9.165188  -0.275188
242841.22  512945.95   7.58  9.170208  -1.590208
242840.92  512946.21   7.42  9.170436  -1.750436
242840.47  512946.54   7.33  9.17049   -1.84049
242839.11  512947.64   7.32  9.170586  -1.850586
242835.29  512950.64   7.23  9.197065  -1.967065
242821.29  512962.05   7.17  9.140288  -1.970288
242820.76  512962.44   7.16  9.119136  -1.959136
242811.67  512969.86   6.93  8.96213   -2.03213
```

Table 1: Pipeline combined with height surface results in z-coordinates at the surface level and z-coordinates with respect to the surface level. Note that the first part of the pipeline is located above the surface.

5.1 Querying

The query to obtain the triangles covering one parcel, is first the realisation of the geometries of triangles and then selecting the triangles that are located within a parcel. The selected realised geometries are the triangles of interest. To simplify the query, the realised polygons of parcels are stored explicitly in the parcel table.

To speed up the query a function-based index was built on the TIN table. A function-based spatial index facilitates queries that use locational information (of type sdo_geometry) returned by a function. The spatial index is created based on the precomputed values returned by the function:

```
insert into user_sdo_geom_metadata values(
'TIN_C', 'stoter.return_geom(id)', mdsys.sdo_dim_array
(
mdsys.sdo_dim_element('X', 0, 254330, .5),
mdsys.sdo_dim_element('Y', 0, 503929, .5)), NULL
);

create   index   tin_idx   on   tin_c(RETURN_geom(ID))   indextype   is
mdsys.spatial_index;
```

The spatial query to find all points or triangles that are located within one parcel can be performed in two ways (in Oracle Spatial terms) with the spatial operator (sdo_relate) and the spatial function (sdo_geom.relate). The spatial operator requires and utilises a spatial index and is therefore faster than the spatial function which does not use an index.

Unfortunately, in the current version of Oracle operators only work if the dimension of the operands is equal to each other. To illustrate the difference between the spatial operator and the spatial function, we stored the triangles of the TIN in 2D and queried which 2D triangles are inside one particular parcel (number 461, municipality: GBG00, section L):

```
/* query to obtain triangles covering one parcel

/* operator
select id, return_geom(id) shape from tin_c, parcels par where
parcel='  461' and
municip='GBG00' and
section=' L' and
sdo_relate(return_geom(id),par.geom,'mask=COVEREDBY+INSIDE,
querytype=JOIN') = 'TRUE';

/* function
select id, return_geom(id) shape from tin, parcels par
where
parcel='  461' and
municip='GBG00' and
section=' L' and
sdo_geom.relate(par.geom,
'COVEREDBY+INSIDE',return_geom(id),1)='TRUE';
```

In both cases the result is 1855 triangles.
In the constrained TIN the maximum number of triangles in a parcel is 31193, the minimum number is 15 and the average number is 817. For the unconstrained TIN these numbers are: 15121, 1 and 348.

The sdo_relate operator and the sdo_geom.relate function are the implementation of the 9-intersection model of Egenhofer (Egenhofer, 1992) in Oracle for finding binary topological relations between points, lines, and polygons. Each spatial object has an interior, a boundary, and an exterior. The boundary consists of points or lines that separate the interior from the exterior. The boundary of a line consists of its end points. The boundary of a polygon is the line that describes its perimeter. The interior consists of points that are in the object but not on its boundary, and the exterior consists of those points that are not in the object. Some of the topological relationships of the 9-intersection model have names associated with them. The following names are used in the examples above:

- INSIDE: returns INSIDE if the first object is entirely within the second object and the object boundaries do not touch; otherwise, returns FALSE;
- COVEREDBY: returns COVEREDBY if the first object is entirely within the second object and the object boundaries touch at one or more points; otherwise, returns FALSE.

6. CONCLUSION

In order to position 3D objects (tunnels, pipelines)defined with absolute z-coordinates with respect to the surface, the surface of parcels is needed. In this research two registrations are used: the parcel boundary registration by the Netherlands' Kadaster and the height registration by the Meetkundige Dienst of Rijskwaterstaat. The optimal way of the integration of height in the cadastre is when both registrations (parcel boundaries and point heights) are maintained in a DBMS.
The best process of the combination of the two registrations is obtaining all height points that are located in one parcel and then generate a constrained TIN (with the parcel boundaries as constraints). In the constrained TIN, triangles only belong to one parcel and the area of all triangles located within one parcel can be summed up to get the area of the parcel in 3D space.
In this research we used a study area to study the possibilities of the integration. Both registrations were inserted in the DBMS. The TIN, in this research generated outside the DBMS, is stored in the DBMS in a topological model with a geometrical view on top of it. The integration of the height data and the parcel boundaries makes it possible to position 3D objects with respect to the surface (what is the depth/height of a 3D object at this location?). In the future the TIN computation should be performed inside the DBMS.
One of the disadvantages of using a dense laser altimetry dataset is the resulting data volume and with that the poor performance of the queries. It is therefore relevant to look how the number of TIN nodes can be reduced by removing nodes that are not significant for the TIN taking the constraints of the parcel boundaries into account: remove nodes based on parameters such as maximum angle between two points and the difference between the original point and the reduced TIN. In our department a study has been carried out on iterative data reduction in an unconstrained TIN by detecting the characteristic point heights

(Penninga, 2002). Data reduction of the constrained TIN, while maintaining the quality, will improve the efficiency of the TIN considerably and will therefore be topic of future research.

ACKNOWLEDGEMENTS

We would like to thank the NAM and the Meetkundige Dienst of Rijkswaterstaat for using their data. We would also like to thank the Netherlands' Kadaster for using their data and for their support. Finally we would like to thank Peter van Oosterom for his comments on a previous version of this paper.

REFERENCES

Egenhofer, M.J. and J.R.Herring (1992), Categorising topological relations between regions, lines and point in geographic databases, in: The 9-intersections: formalism and its use for natural language spatial predicates, Technical report 94-I, NCGIA, University of California.

FIG (1995), Federation International des Géomètres (1995): The FIG Statement on the Cadastre, Publication No. 11, 1995.

Heerd, van, R.M., E.A.c. Kuijlaars, M.P. Teeuw and R.J. van 't Zand (2000), product specificatie AHN, March 15th 2000, Meetkundige Dienst, Rijskwaterstaat, MDTGM 2000.13.

Lenk, U. (2001), Strategies for integrating height information and 2D GIS data, presented at the joint OEEPE/ISPRS workshop- From 2D to 3D, establishment and maintenance of national core spatial databases", held Oct. 8-10, 2001 at Hannover, Germany

Oracle (2001) Oracle Spatial User's Guide and Reference Release 9.0.1 Part Number A88805-01, June 2001.

Oosterom. P.J.M. van, J.E. Stoter, S. Zlatanova and W.C. Quak (2002). The balance between Geometry and Topology, Spatial Data Handling 2002 Symposium, Ottowa, Canada, 9-12 July 2002.

Oosterom, P.J.M. van and C.H.J. Lemmen (2001), Spatial data management on a very large cadastral database, Computers, Environments and Urban Systems (CEUS), volume 25, number 4-5, pp. 509-528.

Penninga, F. (2002), Detectie van kenmerkende hoogtepunten in TIN’s voor iteratieve datareductie (in Dutch), Geo-informatiedag Nederland 2002, February 2002, Ede, the Netherlands.

Stoter, J.E., M.A. Salzmann, P.J.M. van Oosterom and P. van der Molen (2002), Towards a 3D Cadastre, FIG XXII/ACSM-ASPRS, 19 - 26 April, 2002, Washington, USA.Stoter, J.E.

Stoter, J.E. and H.D. Ploeger (2002), Multiple use of space: current practice of registration and development of a 3D cadastre, UDMS 2002, Prague, Czech Republic

October 1 - 4, 2002.

Stoter, J.E. and H.D. Ploeger (2003), Registration of 3D objects crossing parcel boundaries, to be presented at FIG workingweek 2003, April 2003, Paris.

Stoter, J.E. and P.J.M. van Oosterom (2002), Incorporating 3D geo-objects into a 2D geo-DBMS, FIG, ACSM/ASPRS, April 19-26 2002, Washington D.C. USA

Shewchuk, J.R. (1996), Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator, First Workshop on Applied Computational Geometry (Philadelphia, Pennsylvania), pages 124-133, ACM, May 1996.

Worboys, M.F. (1995), GIS, a computing perspective, Taylor and Francis, London, ISBN 0-7484-0065-6.

BIOGRAPHICAL NOTES

Jantien Stoter graduated in Physical Geography in 1994. She started her career as a GIS specialist/consultant, with the District Water Board of Amsterdam and Surroundings (1995-1997). From 1997 till 1999 she worked as a GIS specialist/consultant at the Engineering Office Holland Rail Consult. Since 1999 she has been an assistant professor in GIS applications, section GIS technology, Department of Geodesy, Delft University of Technology. Also doing a PhD on 3D cadastres. In this research the needs, possibilities, and constraints are studied for 3D cadastral registrations. The emphasis of the research is the implementation of the facility to incorporate 3D real estate objects (geo-objects) in the current 2D geo-DBMS of the Netherlands' Kadaster. The research of Jantien Stoter focuses on 3D GIS, modeling geo-objects in 3D, and geo-DBMSs.

Ben Gorte obtained a B.Sc in applied mathematics (1977) and an M.Sc. in computer science (1983). Between 1980 and 2000 he held various positions at the ITC (International Institute for Aerospace Survey and Earth Sciences) in Enschede, the Netherlands, lastly as an assistant professor for knowledge based analysis of remotely sensed imagery. In 1999 he obtained a Ph.D. degree with a thesis on this subject. Since 2000 he has been an assistant professor within the Photogrammetry and Remote Sensing Section, Department of Geodesy, Delft University of Technology.

CONTACTS

Jantien Stoter and Ben Gorte
Department of Geodesy
Faculty of Civil Engineering and Geosciences
Delft University of Technology
P.O. BOX 5030
2600 GA DELFT
THE NETHERLANDS
Tel. + 31 15 278 8136
Fax + 31 15 278 2745
Email: {j.e.stoter|b.g.h.gorte}@citg.tudelft.nl
Website: www.gdmc.nl/stoter