Buildings and terrain unified – multidimensional dual data structure for GIS

Pawel BOGUSLAWSKI [a*] and Christopher GOLD [b,c]

[a]*Department of Architecture and the Built Environment, University of the West of England, Frenchay Campus Coldharbour Lane, Bristol, BS16 1QY, United Kingdom;* [b]*School of Computing and Mathematics, University of South Wales, Pontypridd, United Kingdom;* [c]*Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu, China*

(Received 15 July 2015; final version received 11 August 2015)

3D city models are widely used in many disciplines and applications, like urban planning, disaster management, environmental simulation, etc. Usually, the terrain and embedded objects like buildings are taken into consideration. A consistent model integrating these elements is vital for GIS analysis, especially if the geometry is ac-companied by topological relations between neighbouring objects. Such a model allows for more efficient and errorless analysis. The memory consumption is another crucial aspect when the wide area of a city is considered – light models are highly desirable. Three methods of the terrain representation using the geometrical-topological data structure – the dual half-edge – are proposed in this paper. The integration of buildings and other structures like bridges with the terrain is also presented.

Keywords: multidimensional modelling; GIS; data structures

∗Corresponding author. Email: pawel.boguslawski@uwe.ac.uk

1. Introduction

GIS as it was originally envisaged was largely concerned with the polygon-based description of the land ([1]). Shortly afterwards terrain elevation was included, based on grid structures – see Burrough ([2]) for an extensive description of this approach, together with its important use in water runoff simulation. Various interpolation techniques were developed to estimate these grid values from the original data points – see Lam ([3]) for an early survey. Sometime after this, triangulations (TINs) were developed to model the variation in the terrain elevations by connecting the observed elevation values – see, among others, Gold et al. ([4]). After some time it was recognized that this structure, based on the Voronoi/Delaunay duality, formed a strong basis for precise data-fitting surface modelling ([5]). It was not long before graphics systems started "planting" simple building models on the landscape: examples of this are available from various commercial vendors.

The primary limitation of this approach is that, with no link between the building and the ground, any form of simulation, such as water runoff, cannot take the building into account: in addition, these are only simple building exteriors — just another 2D surface.

CityGML addresses the second of these issues by defining the classes and relations between a variety of objects, such as walls, doors and windows, and defines several hierarchical levels ([6]). However, the system does not address the topological relations between rooms, or between the rooms and the exterior terrain: such a system would be incomplete for personnel evacuation planning, for example. Another approach ([7]) modifies the terrain model to permit bridges and tunnels, by taking advantage of the quad-edge data structure ([8]) to build CAD-type b-rep structures: again, this is only a 2D structure, the exterior of a complex polyhedron.

Attempts to address the issue of escape route planning have been made by Lee ([9]), Slingsby and Raper ([10]) and Pu and Zlatanova ([11]), among others: these are largely based on floor plans, again 2D surfaces, with connecting stairways. The dual half-edge (DHE) ([12]) addresses the problem in a fully 3D fashion, where rooms are volume entities and the dual graph expresses the relationships between rooms – which may be easily accessed through doors or, with more difficulty, through walls or ceilings. This permits interior navigation. However, this DHE structure differs from that of the exterior terrain, which is modelled by a 2D structure such as the quad-edge, making it difficult to integrate interior and exterior navigation for escape route planning. This paper attempts to resolve this problem by integrating the terrain and buildings into one model in order to perform spatial analysis which should be of particular value in the urban environment.

2. The terrain and buildings in a 3D city model

The geometric-topological concept of the 3D city model was presented by Gröger and Plümer ([13]). They emphasize the difference between terrain and building dimensionality: in GIS the terrain is usually represented with a surface model which is 2D, 2.5D or 2.8D while buildings and urban structures are represented with solids (bounded by surfaces) which are 3D models.

In mathematical terms the terrain surface is 2-manifold – the neighbourhood of each point is topologically equivalent to a disc. In other words, each point of the surface divides space into two regions: inside and outside of the object; if any point does not divide space into two regions, then the object is non-manifold. For example, the surface of a sphere, a cube, or a torus is 2-manifold; two polyhedra joined by a vertex, edge, or face are non-manifolds.

A building can be represented as a simple block (a solid) or a complex structure including interior: often represented as a cell complex. A single solid is 2-manifold.

The topological difference between the terrain surface and the surface of a solid is that the former has exactly one unbounded face (at the boundary of a terrain model) while this face is not present in the latter (all the surfaces are used to "seal" the solid): therefore, a terrain surface is equivalent to a disk while a surface of a solid is equivalent to a sphere ([13]) – see Figure 1.

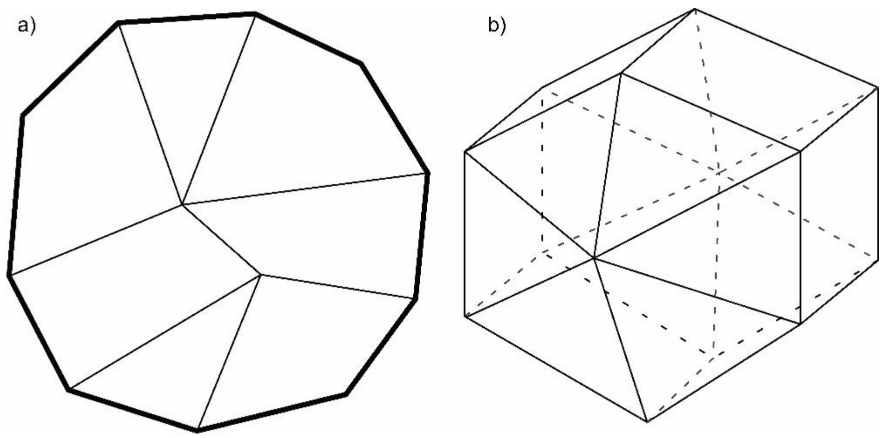

Figure 1. 2-manifolds: (a) a surface – equivalent to a disk; (b) a solid – equivalent to a sphere.

Connected terrain and building may form a 2-manifold model when only a building exterior is present (Figure 2(a)). This is called 2.8D Map ([13]) and is a common representation of city models which focus on the external terrain (often with a street network) and the shapes of buildings (often with rasterised facades). These kinds of models may be used for tourist applications, e.g. virtual city tours; or air pollution simulations, e.g. air circulation along a street and between buildings.

A building represented as a separate object attached to the terrain forms a non-manifold model (Figure 2(b)): a building intersects with the terrain. The bottom surface of the building must fit the terrain at the connection. This produces a set of intersection edges, each shared by three surfaces: the terrain, a building wall, and the bottom surface of the building. This representation is useful for applications where semantic information attached to a building is required, e.g. a building volume, an address, etc.

An extended version of the previous model consists of buildings including underground parts (Figure 2(c)): they intersect the terrain but the bottom surface of the building does not need to fit the terrain surface. The building may be represented: as a single object; as a composition of over- and underground- parts; or a complex with the indoor structure. The complex representation with rooms, walls and furniture is used in emergency management applications, for example, to find escape routes from a building.

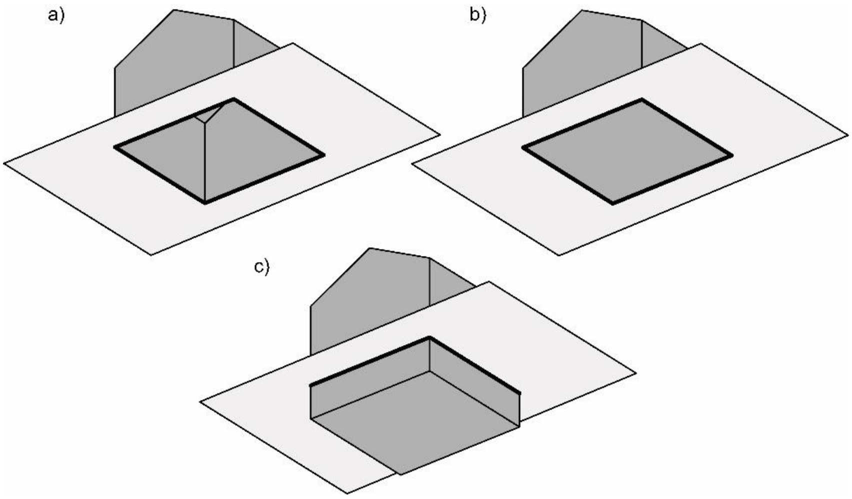

Figure 2. Connections between the terrain and building: (a) 2-manifold model – only a building exterior is present; (b) non-manifold model – a building is put on the terrain; (c) non-manifold model – a building consisting of over- and underground- parts intersects with the terrain.

2.1 Dual half-edge (DHE)

The DHE data structure was inspired by the quad-edge (QE) ([8]) and augmented quad-edge (AQE) ([14]) data structures. The first was developed to represent 2-manifolds where each quad-edge consisted of four quads representing one geometrical edge and its dual. The second data structure was designed for 3D cell complex construction, with the dual graph providing the connectivity between the volume elements. Unfortunately the AQE does not have easy incremental construction operations, thus the DHE and a set of construction operators were proposed ([15]).

A geometrical-topological model proposed in this paper for building and terrain representation is based on a simple idea – it consists of edges and nodes; edges bounded by nodes and linked together form a cell; several cells are connected into a cell complex by edges, but in the dual space. These dual edges also represent primal faces. The dual nature of the model is defined by 3D Poincaré duality rules where each primal node has an associated dual cell, a primal edge – a dual face, a primal face – a dual edge, and a primal cell – a dual node (see Figure 3).

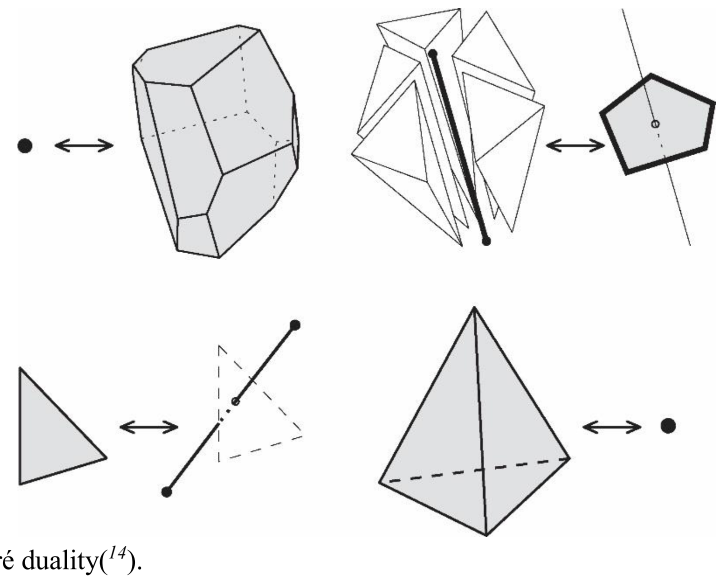

Figure 3. Poincaré duality([14]).

Technically, the model is using the boundary representation (b-rep) ([16]) and consists of two graphs: one in the primal space – considered as the geometry of the model, and the other in the dual – a graph of connections between cells. Each edge of the graph has two parts, i.e. half-edges. The DHE element, which is an atomic element, consists of two connected half-edges – one in the primal, and one in the dual. However, only two DHEs connect into an edge form a minimal topological element, hence a valid model.

All edges in the model are connected with pointers. A half-edge consists of five pointers: V, S, N_V, N_F and D, where: V – is a pointer to an associated vertex; S – a second half-edge of the edge; N_V – a next half-edge around a shared vertex (counter-clockwise (CCW) looking from outside of a cell); N_F – a next half-edge around a face (CCW looking from outside of a cell); and D – a pointer to the dual half-edge. S, N_V and N_F pointers allow for navigation in a 2-manifold cell. The full set of navigation operators is based on these pointers, and also includes compound operators for navigation: around a shared vertex and face in a clockwise (CW) direction, around an edge (a radial edge) – in both directions, and also between the adjacent faces of two neighbouring cells.

The construction process is based on Euler operators ([17]). They are widely used in CAD systems. Each operator makes a minimal change in a model while the topological integrity is preserved. Basically, a cell is built up by adding edges and vertices one by one. For example, *Make Edge and Vertex* (*MEV*) creates a new edge and new vertex, and connects the edge with the model at a given location. Each operator is accompanied by a paired reverse operator, thus any change in a model can be undone, e.g. *Make Edge and Vertex*, *Kill Edge and Vertex*. Euler operators were designed for a single 2-manifold cell construction. Non-manifold models require an extended set of operators ([18]).

2.2 Modified DHE

The data structure presented in the previous section is suitable for 3D cell complexes. The terrain can also be modelled in the same way but this will be shown in the next section. This is a memory consuming method. Thus, a modified version of the standard DHE was developed which is similar to the QE data structure. The main difference is in the pointer connections of a separate edge: 1) In the standard DHE, the D pointer of a primal edge (e) points at the dual edge and D in the dual edge points back at the same primal edge ($e.D.D=e$) (see Figure 4(a)), while in the modified version D plays the same role as Rot in QE – D in the dual edge points at the second half-edge of the primal edge ($e.D.D=e.S$) (see Figure 4(b)). 2) N_V in the dual edge points at itself in the standard DHE ($e.D.N_V=e.D$) (see Figure 4(c)) while in the modified version N_V points at the second half-edge ($e.D.N_V=e.D.S$) (see Figure 4(d) – for the sake of clarity the edge is drawn on a sphere (grey dotted line)). In Figure 4, a primal edge is represented as a black solid line, while a dual edge – as a grey solid line.

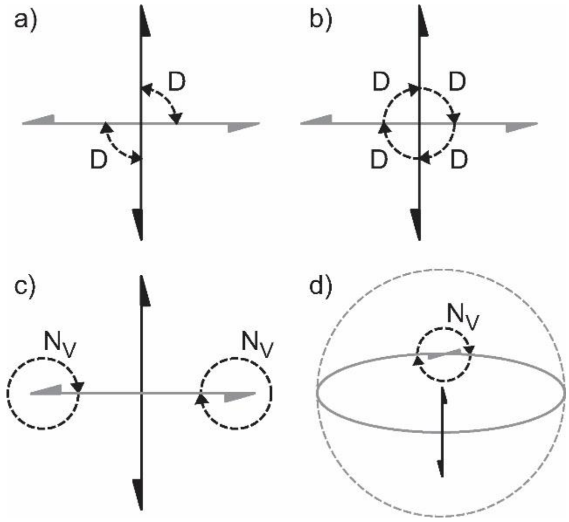

Figure 4. Differences between the standard and modified DHE – pointers: (a) *D* in the dual edge points back at the primal edge (standard DHE); (b) *D* in the dual points at the second part of the primal edge (modified DHE); (c) N_V in the dual edge points back on the same dual edge (standard DHE); (d) N_V in the dual edge points at the second end of the dual edge (modified DHE).

Also the construction process is different. However, operators used in the modified version are the same as the low level operators used by Euler operators in the standard version ([15b]). The only operators used in the modified version are the same ones used in the QE method – *MakeEdge/KillEdge* to create a single edge, and *Splice* – to connect the edge with the model. It is possible to develop a set of Euler operators for the modified DHE as it was done for the QE data structure ([7]).

There are also some modifications in the compound navigational operators to provide correct structure traversal.

The modified DHE can be used for 2.8D modelling (2 manifold models with vertical faces and overhangs). Fully 3D models are not possible – however, the connection between 3D models created using the standard DHE (e.g. buildings) and models created using the modified DHE (e.g. the external terrain) is possible. Because the modified version is limited, some pointers are redundant and can be replaced with compound operators: N_F and *S* can be removed. It means that the number of pointers for a half-edge in the modified version can be reduced to three: N_V, *D*, and *V*. A primal and associated dual edge, consisting of four half-edges, requires 12 pointers – the same number as in the case of the QE. The meaning of the pointers is the same in both cases N_V=*next*, *D*=*rot*, and *V*=*org*. Thus, it is possible to claim that the modified DHE is an equivalent for the QE.

3. Unified terrain and building models

The standard DHE is suitable for modelling building interiors as a cell complex – cells representing rooms are linked with neighbouring cells by a shared face ([12, 15b]). To guarantee that each face is associated with another face an external cell was introduced. This is necessary in order to provide consistency at the boundaries of the model where the cells do not have neighbours on each face.

In 3D city models buildings are spatial objects. Despite the fact that they are simple blocks or complex models including interiors, using the DHE they are always represented as cell complexes – even a simple block consists of two cells: internal and external. Thus the terrain can be linked with the building exterior into one consistent model. If the terrain is a continuous model it is necessary to intersect it with the building exterior and the shared part should be removed.

There are three possible methods of terrain representation using the DHE: 1) the terrain is tessellated into thick cells forming a complex; 2) the terrain is represented as a mesh of flat double-sided faces; 3) the terrain is represented as a mesh of polygons, e.g. a triangulated surface. The advantages and disadvantages of these methods are described below.

Method 1 is easy to implement, as the external terrain is modelled in the same way as buildings ([12, 15b]) - it consists of thick cells (each cell has a volume). For example, cells can be extruded from a 2.5D Digital Terrain Model (DTM) by setting a relatively small height. Then they can be linked with a building by shared faces on a boundary between them. Figure 5 shows the situation where one of the terrain cells (grey) is connected to a building by shared faces (dashed). One of the faces is linked to a door cell – this connection identifies the door as an exit from the building.

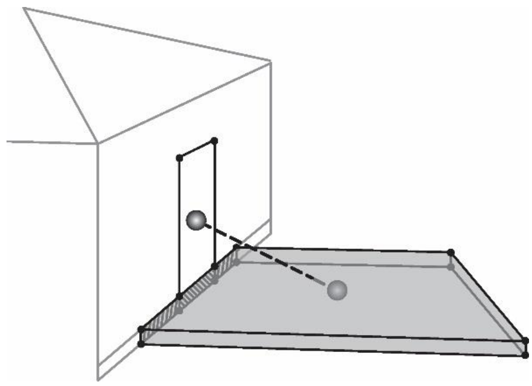

Figure 5. A building-terrain connection. The terrain is represented as 3D cell complex.

As buildings and the terrain form a consistent model represented as one graph, analysis (e.g. escape routes calculation) can be performed using the same methods for the building interior and the external terrain. Navigation inside buildings can be limited to the 'walkable' connections between rooms which have a door in between, while navigation between terrain cells is allowed without limitation.

Ease of navigation is one of the advantages. The main disadvantage is a large storage cost – one cell representing a rectangular piece of terrain (the grey cell in Figure 5) consists of 32 DHEs including the internal and external parts; each DHE consists of ten pointers (five in the primal and five in the dual); assuming each pointer takes four bytes of memory then 1,280 bytes are necessary to store the cell. Cells on the model boundary are even more memory consuming as their external faces are connected only to the external cell (associated edges in the external cell are not shared by adjacent internal cells) – it takes 160 bytes more for each rectangular face (four DHEs). These calculations take the full version of the DHE into consideration. In the case of *Method 1* a simplified data structure can be used ([15b]). However, for the sake of description clarity it is not discussed in this paper.

Method 2 is based on the Cardboard & Tape construction method ([15b, 19]), where double-sided faces are linked by shared edges. The DTM is represented as a set of polygons: for example a triangulated mesh of elevation points can be straightforwardly reconstructed. Simplified, the resulting terrain model consists only of an external cell which can be connected to the external cell of the building by shared edges. The process is shown in Figure 6: a) firstly, a building represented by a box is put onto the terrain mesh consisting of flat double-sided polygons (the volume is zero); b) then the building external cell is connected to the terrain; c) the building external cell and terrain are merged into one cell, while the building interior (grey box) is not affected. Technically the building-terrain connection can be done in two ways: 1) building external faces which fit to terrain faces (e.g. the same face node coordinates) are merged – the central, biggest face of the terrain model in Figure 6(a)) is merged to the bottom external face of the building; 2) the terrain mesh is intersected with the building external cell and removed, then these two models are merged by shared edges. The final model is the same in both of these cases.

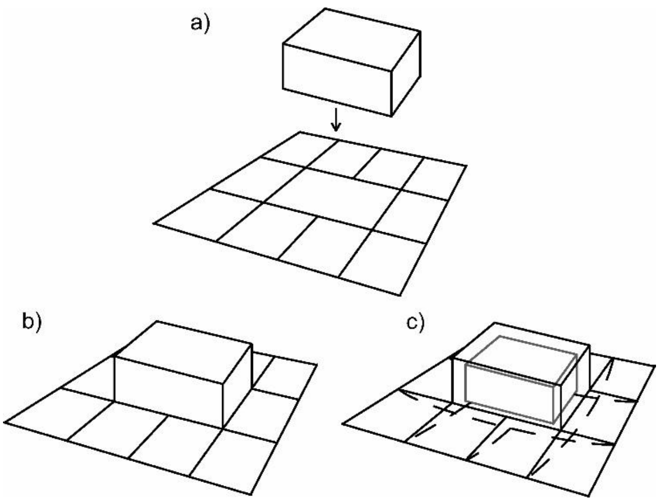

Figure 6. A building-terrain connection: (a) the terrain represented as a complex of double-sided faces and 3D building model before the connection; (b) the connected building and terrain; c) the terrain is linked to the building external cell.

Method 2 is more economical then the first one. Since the terrain is represented as a mesh of polygons instead of a cell complex, the memory consumption is much lower. A rectangular piece of terrain requires only eight DHEs, which takes 320 bytes – this is only 25% of the first method storage. However, there are also some disadvantages. One of them is that the same traversal algorithms cannot be used for the building interior and the terrain – in buildings navigation is based on a cell-to-cell traversal, while in the terrain, navigation from face-to-face within one cell is performed. Terrain polygons also need to be marked somehow to allow navigation only within the terrain to avoid using building external polygons – containing vertical walls.

The last proposed *Method 3* is the least memory consuming. However, the data structure and operators need some modifications. In this method the terrain is represented as a mesh of polygons – in a similar way as using quad-edges. Each polygon is represented as a loop of edges shared with adjacent polygons. Polygons can be of any shape – in a special case they are triangles.

The terrain construction process is simple – an edge which is a basic entity is created using the *MakeEdge* operator, and linked with the mesh using *Splice*. Building models are still cell complexes constructed using the 3D standard DHE data structure and operators. Buildings and the terrain are then merged by shared edges – to make it viable it may be necessary to add new edges to the building external cell. The resulting model consists of building internal and external cells linked with the terrain model – in the example shown in Figure 7(a)) the building is represented as a simple block. The terrain surface splits the model into the underground and aboveground parts – see Figure 7(b)) and Figure 7(c)) respectively. This representation is closely related to the concept of the geometric-topological 3D city model presented by Gröger and Plümer [13]).

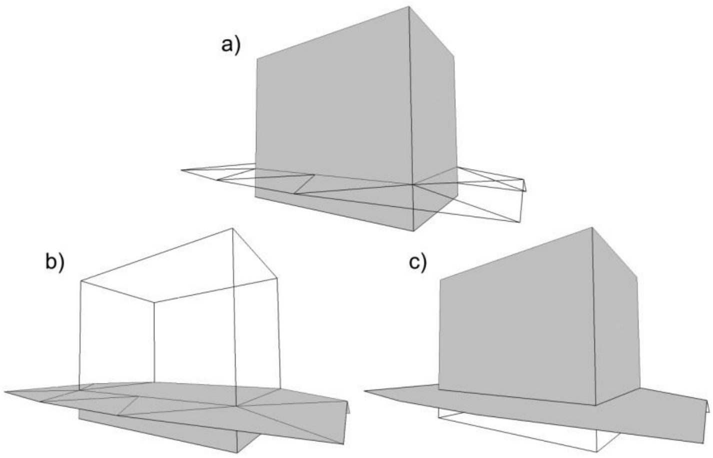

Figure 7. Building and terrain unified model: (a) the building internal cell; (b) the terrain with an underground part of the building external cell; (c) the terrain with an aboveground part of the building external cell.

The difference between buildings and the terrain is that the building model is, in a simple case, a complex of 2-manifold cells where navigation between cell edges and between cells is possible, while the terrain is a single 2-manifold cell. Thus navigation between edges of the mesh is realized with modified DHE operators, which allow navigation around a shared vertex, around a face, and to the dual structure.

One consequence of using two different data structures in one model is that they have to be distinguishable. One idea is to add a flag to the structure naming the type: standard (used for buildings) or modified DHE (used for the terrain). Additionally, the redundant N_F pointer which in the standard version is used to navigate around a face can be removed – in the modified version it can be replaced with a compound operator consisted of N_V and D pointers. Similarly the S pointer can be implemented as a compound of two D pointers – this is only allowed for the terrain model using *Method 3*.

Assuming the flag is an extra one-byte field, a one rectangular piece of terrain takes 168 bytes and 136 bytes, including and without the N_F pointer respectively.

Figure 8 shows the differences between *Methods 2* and *3* – the terrain model in *Method 2* consists of double-sided faces, thus edges on the top side are repeated on the bottom (see Figure 8(a)); in *Method 3* the bottom side contains only one polygon (see Figure 8(b)) which can be considered as an external polygon of an infinite area enclosing the mesh of polygons on the top side.

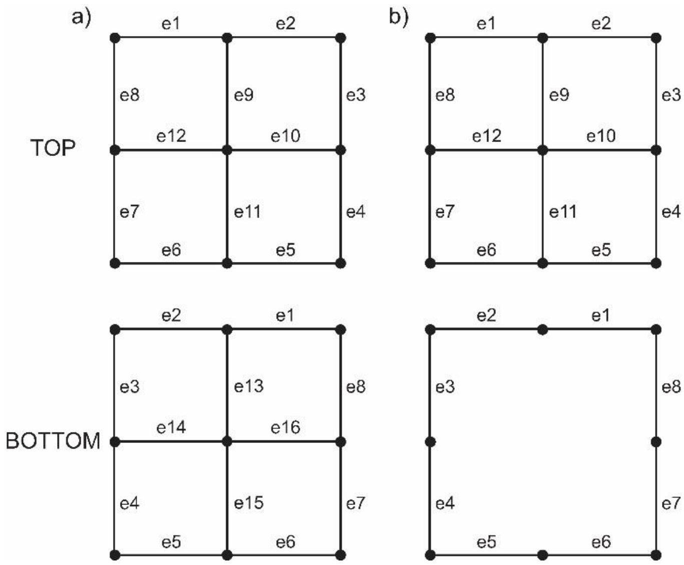

Figure 8. The different terrain representations in (a) method 2 and (b) method 3. e1-e16 – edges.

The memory consumption comparison concerning all presented methods of terrain representation is shown in Table 1. *Method 1* is significantly more memory consuming compared to others. But only this 'expensive' method allows one to use the same traversal algorithms as in the case of building interiors.

Table 1. Three methods of terrain representations with respect to memory consumption.

Terrain representation	Memory occupied by one rectangular cell (bytes)	Ratio (Method x/CC)
Cell complex (CC) – Method 1	1,280	100%
Double sided faces – Method 2	320	25%
Polygon mesh – Method 3	168 (including NF) 136 (without NF)	13% 11%

It should be noted that bridges, tunnels, and arcades can be modelled using the proposed methods. Connection of a bridge or tunnel with the terrain is crucial for 3D GIS analysis ([20]). *Method 1* and *method 3* were used for a bridge model construction (see Figure 9 and Figure 10 respectively). In the first example five cells were linked into one cell complex (see Figure 9(c)). The external cell is the sixth cell of the complex (see Figure 9(b)). Connections between cells are carried out with dual edges – grey edges in Figure 9(a). The dual structure is different in the second example – dual edges connect polygons in the primal (grey lines in Figure 10(a)). The model is represented as one cell (see Figure 10(b)); also dual edges form a single cell (see Figure 10(c)).

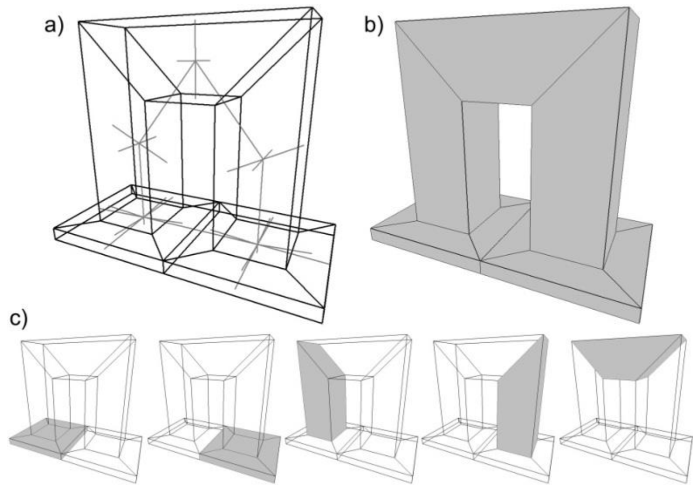

Figure 9. A bridge modelled using Method 1: (a) the primal (black lines) and the dual (grey lines) structures; (b) the external cell; (c) five internal cells.

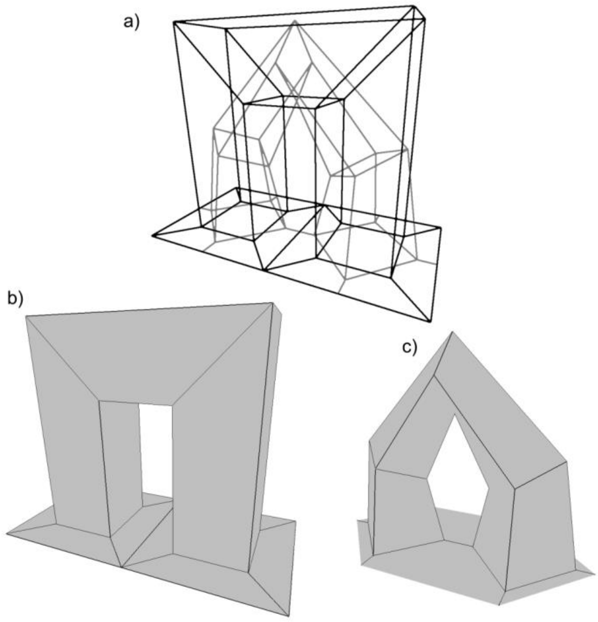

Figure 10. A bridge modelled using Method 3: (a) the primal (black lines) and the dual (grey lines) structures; (b) the primal cell; (c) the dual cell.

4. 2D+3D spatial analysis

Joining 2D and 3D models forms dual connections at the boundary between models which should be managed in a different way than in a case of separate 2D and 3D models. In order to keep navigation properties typical for each part, a connector cell was introduced as shown in Figure 11: the only special navigation procedure appears at the connector cell.

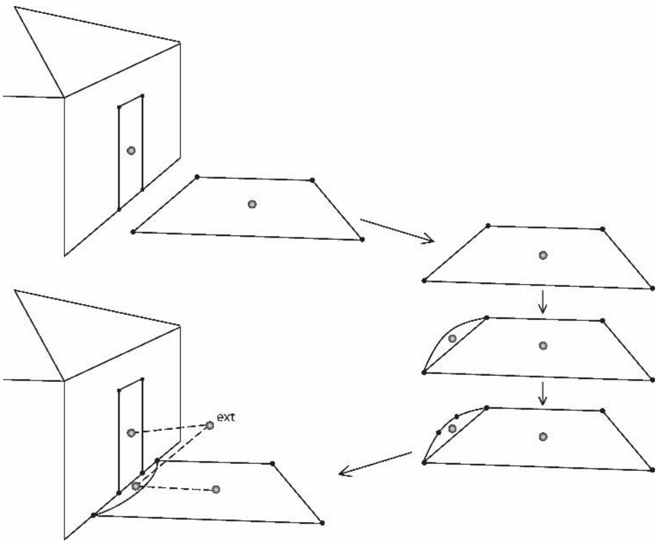

Figure 11. A process of connecting a 2D terrain model with a 3D building model: a new cell is added to the 2D model at the boundary which is used to connect with the external (ext) cell of the 3D model using shared edges.

Graph traversal algorithms (e.g. Dijkstra's algorithm) need to consider the connection between models of different dimensionality while normal navigation procedures apply to individual parts of the same dimensionality.

This problem does not appear in a scenario where the terrain is represented by 3D cells because the whole model has the same dimensionality – navigation is allowed: a) if there is a door between adjacent indoor cells, and b) between adjacent terrain cells. Indoor and terrain cells are recognized based on attributes attached to their dual nodes.

In 2D+3D model the connector detection may be done by checking the flag (introduced in the previous section) or by adding and testing a special attribute at the dual node representing the connector cell.

5. Conclusions

A DTM often includes a wide area, e.g. a city. Thus, a memory-efficient representation is crucial. The terrain can be linked with buildings into one structure in order to provide a consistent city model for further analysis. Three methods of terrain model representation using the DHE data structure were presented and compared in this paper. The first method is memory consuming but the same traversal algorithms as for building interiors can be used, while the last one is memory efficient but the data structure and construction operators need some modifications; different operators and algorithms are required for navigation.

The modified DHE is equivalent to the QE data structure which has strong mathematical foundations. Research findings described in this paper shows that non-manifold models with cells connected by edges can be used for combining models of different dimensionality into one multidimensional model.

It was also shown that structures like bridges or tunnels can be easily integrated with the DTM. This helps to avoid topological errors in GIS models: for example floating objects located above but not attached to the terrain.

Acknowledgements

The authors would like to thank sponsors for their support: research on the dual half-edge data structure was funded by the EPSRC and Ordnance Survey (New CASE Award) (2006-2010); Universiti Teknologi Malaysia and the Ministry of Science, Technology and Innovation (eScience 01-01-06-SF1046, Vot no. 4S049) (2011-2014).

Notes on Contributors

Pawel Boguslawski is a researcher in the Faculty of Environment and Technology at the University of the West of England (UWE), United Kingdom. His research is focused on 3D spatial modeling, data structures, and emergency response in the built environment. Currently, he is Co-Chairman of the ISPRS Working Group II/2 "Multiscale N-dimensional Spatial Data Representations, Data Structures and Algorithms" and ISPRS UV Helava Award jury member. He has been Conference and Programme Committee Chairman and reviewer in several journals.

Christopher Gold is the emeritus professor in Department of Computing and Mathematics at University of South Wales. He is also the visiting professor and lecturer in Southwest Jiaotong University, China, and Technical University of Malaysia, Malaysia. His research interests include 3D GIS, data structures, and algorithms.

References

1. Chrisman, N. R.; Dougenik, J. A.; White, D. In *Lessons for the design of polygon overlay processing from the Odyssey Whirlpool algorithm*, 5th Intl. Symp. On Spatial Data Handling SDH'92, Charleston, Bresnahan, P. J.; Corwin, E.; Cowen, D. J., Eds. Humanities and Social Sciences Computing Lab, University of South Carolina: Charleston, 1992; pp 401-410.
2. Burrough, P. A., *Principles of geographical information systems for land resources assessment*. Oxford University Press, USA: 1986; Vol. 12.
3. Lam, N. S. N., Spatial interpolation methods: a review. *Cartography and Geographic Information Science* **1983,** *10* (2), 129-150.
4. Gold, C. M.; Charters, T. D.; Ramsden, J. In *Automated contour mapping using triangular element data structures and an interpolant over each irregular triangular domain*, SIGGRAPH - 4th International Conference on Computer Graphics and Interactive Techniques, San Jose, USA, George, J., Ed. ACM, New York, NY, USA: San Jose, USA, 1977; pp 170-175.
5. Sibson, R., A brief description of natural neighbour interpolation. In *Interpreting multivariate data*, Barnett, V., Ed. Wiley: Chichester, 1981; pp 21-36.

6. Kolbe, T. H., Representing and Exchanging 3D City Models with CityGML. In *3D Geo-Information Sciences*, Lee, J.; Zlatanova, S., Eds. Springer Berlin Heidelberg: 2009; pp 15-31.
7. Tse, R., O. C.; Gold, C., M., TIN meets CAD: extending the TIN concept in GIS. *Future Gener. Comput. Syst.* **2004,** *20* (7), 1171-1184.
8. Guibas, L.; Stolfi, J., Primitives for the manipulation of general subdivisions and the computation of Voronoi Diagrams. *ACM Trans. Graph.* **1985,** *4* (2), 74-123.
9. Lee, J., A Three-Dimensional Navigable Data Model to Support Emergency Response in Microspatial Built-Environments. *Annals of the Association of American Geographers* **2007,** *97* (3), 512 - 529.
10. Slingsby, A.; Raper, J., Navigable Space in 3D City Models for Pedestrians. In *Advances in 3D Geoinformation Systems*, van Oosterom, P.; Zlatanova, S.; Penninga, F.; Fendel, E. M., Eds. Springer: 2008; pp 49-64.
11. Pu, S.; Zlatanova, S., Evacuation Route Calculation of Inner Buildings. In *Geo-information for disaster management*, van Oosterom, P. J. M.; zlatanova, S.; Fendel, E. M., Eds. Springer Verlag, Heidelberg: 2005; pp 1143-1161.
12. Boguslawski, P.; Gold, C. M.; Ledoux, H., Modelling and analysing 3D buildings with a primal/dual data structure. *ISPRS Journal of Photogrammetry and Remote Sensing* **2011,** *66* (2), 188-197.
13. Gröger, G.; Plümer, L., How to achieve consistency for 3D city models. *Geoinformatica* **2010,** *15* (1), 137-165.
14. Ledoux, H.; Gold, C. M., Simultaneous storage of primal and dual three-dimensional subdivisions. *Computers, Environment and Urban Systems* **2007,** *31* (4), 393-408.
15. (a) Boguslawski, P.; Gold, C., Euler Operators and Navigation of Multi-shell Building Models. In *Developments in 3D Geo-Information Sciences*, Neutens, T.; Maeyer, P., Eds. Springer: 2010; pp 1-16; (b) Boguslawski, P. Modelling and Analysing 3D Building Interiors with the Dual Half-Edge Data Structure. PhD, University of Glamorgan, Pontypridd, Wales, UK, 2011.
16. Stroud, I., *Boundary Representation Modelling Techniques*. Springer-Verlag: New York, 2006.
17. (a) Baumgart, B., G., A polyhedron representation for computer vision. In *National Computer Conference and Exposition*, ACM: Anaheim, California,

1975; (b) Braid, I. C.; Hillyard, R. C.; Stroud, I. A., Stepwise Construction of Polyhedra in Geometric Modelling. In *Mathematical Methods in Computer Graphics and Design*, Brodlie, K. W., Ed. Academic Press: 1980; pp 123-141; (c) Mäntylä, M., *Introduction to Solid Modeling*. Computer Science Press, Inc.: 1988; p 401.

18. Masuda, H., Topological operators and Boolean operations for complex-based nonmanifold geometric models. *Computer-Aided Design* **1993,** *25* (2), 119-129.
19. Boguslawski, P.; Gold, C., Construction operators for modelling 3D objects and dual navigation structures. In *3D Geo-Information Sciences*, Lee, J.; Zlatanova, S., Eds. Springer: 2009; pp 47-59.
20. Gröger, G.; Plümer, L., Topology of surfaces modelling bridges and tunnels in 3D-GIS. *Computers, Environment and Urban Systems* **2011,** *35* (3), 208-216.