ELSEVIER

Computers & Geosciences 30 (2004) 419–428

COMPUTERS & GEOSCIENCES

www.elsevier.com/locate/cageo

Topological models and frameworks for 3D spatial objects

Siyka Zlatanova[a,*], Alias Abdul Rahman[b], Wenzhong Shi[c]

[a] *GIS Technology Section, Delft University of Technology, Thijssweg 11, 2629 JA, Delft, The Netherlands*
[b] *Center for Geographic Information and Analysis (CGIA), Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia*
[c] *Advanced Research Centre for Spatial Information Technology, Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong*

Received 9 December 2002; accepted 2 June 2003

Abstract

Topology is one of the mechanisms to describe relationships between spatial objects. Thus, it is the basis for many spatial operations. Models utilizing the topological properties of spatial objects are usually called topological models, and are considered by many researchers as the best suited for complex spatial analysis (i.e., the shortest path search). A number of topological models for two-dimensional and 2.5D spatial objects have been implemented (or are under consideration) by GIS and DBMS vendors. However, when we move to one more dimension (i.e., three-dimensions), the complexity of the relationships increases, and this requires new approaches, rules and representations. This paper aims to give an overview of the 3D topological models presented in the literature, and to discuss generic issues related to 3D modeling. The paper also considers models in object-oriented (OO) environments. Finally, future trends for research and development in this area are highlighted.
© 2004 Elsevier Ltd. All rights reserved.

Keywords: 3D topological models; 3D data structures; Object-oriented models; 3D analysis; Spatial operations

1. Introduction

Currently, geographic information system GIS professionals and users are content with the capability of existing GISs (i.e., two-dimensional (2D) GISs). These systems can perform numerous 2D spatial analyses and applications. The Open GIS Consortium has agreed on Simple Feature Specifications (geometry) and Complex Feature Specifications (topology). The first implementations of the OpenGIS, SQL/SFS (which became available in 1999), marked an important step forward in the development of GIS, and OpenGIS became a part of the mainstream ICT. As the world we are living in has three or more dimensions, we have to manipulate three-dimensional (3D) spatial data instead of 2D spatial data.

Two types of models (geometrical and topological) have been examined in many studies. The geometrical models are more intuitive and easier to implement. Several mainstream DBMSs (Oracle, Ingres, Informix, IBM and DB2) support spatial objects organized in geometrical models. Some of them even follow the Open GIS standards. Many GIS and CAD packages (MapInfo, ArcGIS, MicroStation, AutoCAD) use geometrical models of DBMS (Zlatanova et al. 2002). Most geometric types supported by DBMS can display 3D spatial objects as 2D objects with 3D coordinates, but their spatial operations are still 2D. Real 3D objects and their corresponding validation functions remain to be implemented.

The evolution of topological models into the third dimension is rather complex when compared to the geometrical models. Many GIS packages construct 2D topological models, and some CAD packages provide tools to check topological consistency (e.g., GeoParcel, MicroStation), and some mainstream DBMSs have 2D topological implementations in their development

*Corresponding author.
E-mail addresses: s.zlatanova@otb.tudelft.nl (S. Zlatanova), alias@fksg.utm.my (A.A. Rahman), lswzshi@polyu.edu.hk (W. Shi).

0098-3004/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cageo.2003.06.004

agendas. 3D topology is still being researched. The third dimension introduces a number of new issues in representing the objects (primitives, rules and constraints) and in detecting their relationships (topology, order, etc.). The suitability of the topological models in 3D for different applications also varies.

This paper addresses difficulties of designing 3D topological models and representing the relationships between them. The paper is divided into three sections. The first reviews various models and discusses the concepts behind them. The second concentrates on frameworks for detecting relationships. The final section concludes the discussion and suggests avenues for further research.

2. Review of 3D topological models

One general question raised when referring to topological models is whether it is possible to have one 3D topological model, that is suitable for all types of applications. The answer is negative. The design of 3D topological models is always closely related to the specific requirements of a particular category of application. For example, space partitioning (full, embedding) depends on the types of queries that have to be represented. In case of many neighborhood operations between 3D objects (such as for geological bodies), full partitioning is recommended. However, if the objects are surrounded by 'free space' (e.g., buildings), the embedding approach is more appropriate. Another aspect relating to application are the types of simple objects (e.g., point, line, surface, body) and primitives used to describe the objects, in 0-dimension (0D), one-dimension (1D), 2D and 3D. In many cases, 0D and 2D primitives are sufficient; e.g., for describing buildings. The last aspect is related to the rules of construction: the types of interrelationships allowed between objects, planarity and convexity rules, and so forth. For example, if only triangles are allowed, many redundant subdivisions of original polygons will be performed, such as windows on a building wall.

In the following, we will give a brief overview of several 3D topological models focusing on the three aspects as mentioned above: space partitioning, supported objects and primitives, and constructive rules.

Two main groups of data structures were found in previous studies: those that maintain objects and those that maintain relationships. In the first group (object-oriented, OO), it is mostly the relationships between objects that have to be derived; in the second (topology-oriented), it is the representation of the objects. Many data structures, for example, that maintain an explicit storage of objects, also maintain an explicit storage of relationships; i.e., singularities.

2.1. *3D topological models with explicit representation of objects*

2.1.1. *3D FDS*

The formal data structure (FDS) was the first data structure to consider spatial objects as an integration of geometric and thematic properties (Fig. 1). A conceptual model and 12 conventions (rules for the partitioning of physical objects) define the structure (Molenaar, 1990). Rikkers et al. (1993) proposed mapping the model into a relational database. The model assumes the full partition of space (similar to the planar partition in 2D space). Besides the feature related to a thematic class, four elementary objects (*point, line, surface* and *body*) and four primitives (*node, arc, face* and *edge*) can be distinguished. *Arcs* and *faces* cannot intersect by convention unless *a node* and *an arc* are created. Singularities are permitted in such a way that *arcs* and *nodes* can exist inside *faces* or *bodies*. The role of an *edge* is dual; i.e., to define the border of a *face* (relationship *face-arc*) and to establish an orientation of a *face*, which is needed to specify the *left* and *right body*. The number of *arcs* constituting an *edge* is not restricted. *Arcs* must be straight lines and *faces* must be planar. The *surface* has one outer boundary and may have several non-nested boundaries; i.e., may have holes or islands. The *body* has one outer surface and can have several non-nested bodies or holes.

The fundamental rule of 3D FDS is the concept of a single-valued map; i.e., the *node, arc, face* or *edge* can appear in the description of only one geometric object of the same dimension (Molenaar, 1998). A single-valued approach can partition the space into non-overlapping objects, thus ensuring 1:1 relationships between primitives and objects of the same dimensions, like *surfaces* and *faces*. Primitives of different dimensions, however, can overlap; for instance, *node-on-face, arc-on-face, node-in-body* and *arc-in-body* relationships are stored explicitly.

3D FDS are used by many to incorporate 3D objects. For example, Shibasaki and Shaobo (1992) implemented

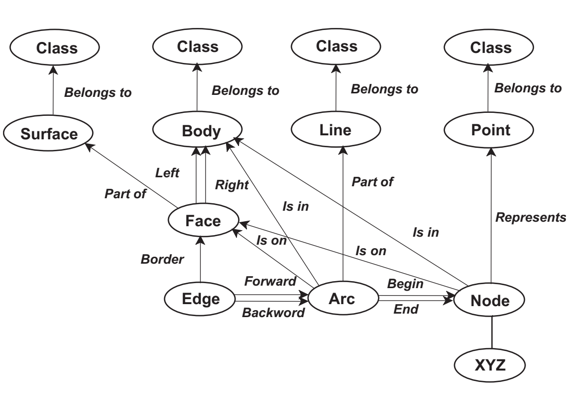

Fig. 1. 3D Formal Data Structure (3DFDS) (Molenaar, 1990).

the model for the maintenance and visualization of 3D city models. De Hoop et al. (1993) investigated possible relationships (based on the nine-intersection model) for 3D FDS. The CC-modeler presented by Grün and Wang (1998) records 3D reconstructed objects in a schema similar to that of 3D FDS but extended to incorporate textures per face.

2.1.2. TEN

The next model, the TEtrahedral Network (TEN) (Fig. 2), was introduced by Pilouk (1996) to overcome some difficulties encountered by 3D FDS in modeling objects with indiscernible boundaries, such as geological formations, pollution clouds, and so forth. TEN, employing a simplex-oriented approach, was proposed to represent 3D objects in the real world (Carlson, 1987). Like3D FDS, it has four primitives (*tetrahedron, triangle, arc* and *node*) and the subdivision of the space is full. It should be noted that this model has a real 3D primitive. In the relational implementation, the *arc-node* relationship is stated in the ARC table; the TRIANGLE table contains the *tetrahedron-triangle-edge* link. A *body* is composed of *tetrahedrons*, a *surface* of *triangles*, a *line* of *arcs* and a *point* of *nodes*. The general rule for creating the model is based on the fact that each *node* is part of an *arc*, each *arc* is part of a *triangle* and each *triangle* is part of a *tetrahedron*. The constraints are simple and very strict: everything is classified into arc, triangles and tetrahedrons. Singularities are not permitted.

2.1.3. SSM

The Simplified Spatial Model (Fig. 3) was the first topological structure that focused on visualization aspects of the queries. It was designed to serve web-oriented applications where spatial queries need to be visualized on the screen as 3D models (Zlatanova, 2000). The model does not require the full partition of space; i.e., all of the objects are embedded in 3D. The simple objects are four, but the primitives used are only two; i.e., *node* and *face*. This is the first representation that avoids the storage of a 1D primitive. The model removes the uniqueness of the relationship *arc/face* in 3D space; i.e., one *arc* can be part of more than two *faces*. However, two successive nodes can implicitly define this primitive. A 3D primitive is not maintained, and *faces* represent the 3D objects. The rules used to describe objects are slightly different from the first two. *Faces* must be planar and convex. Singularities are allowed, with *node-in-face* and *face-in-body* stored explicitly. The orientation of the *faces* is also stored explicitly, and the order of the nodes describes a *face*.

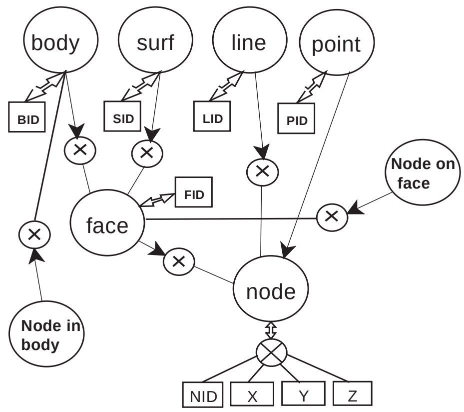

Fig. 3. Simplified Spatial Model (SSM) (Zlatanova, 2000).

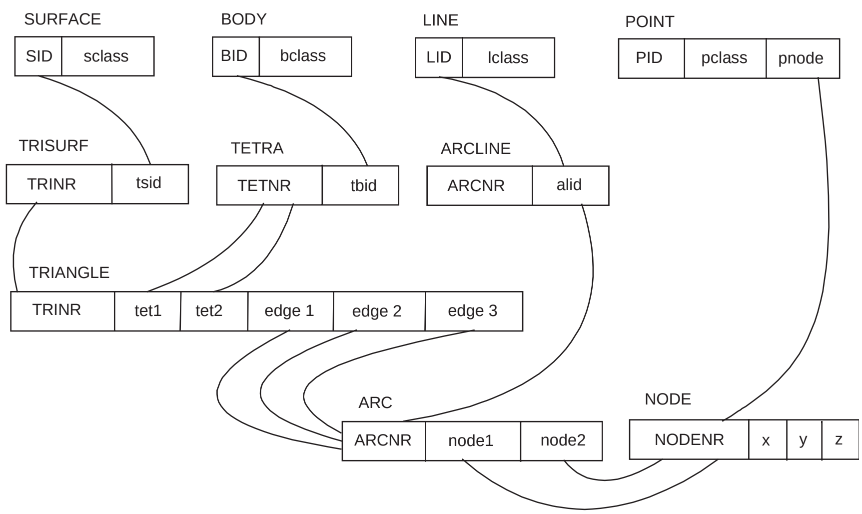

Fig. 2. TEtrahedral Network (TEN): relational implementation for 3D (Pilouk, 1996).

2.1.4. UDM

The Urban Data Model (Fig. 4) is built on a full partition of space and represents the geometry of a *body* or a *surface* by planar convex *faces* (Coors, 2003). Each *face* is defined by a set of *nodes*. Similar to SSM, a 1D primitive is not supported. Two convex planar *faces* are adjacent to each other if they share at least two *nodes*. The orientation of a *face* is stored implicitly. The constructing rules are stricter than SSM. In the relational representation of the model, every *face* having more than three *nodes* is decomposed into triangles, and the FACE table contains only three columns; i.e., the IDs of the three triangle *nodes*. As with 3D FDS, *face-body* relationships are explicitly stored in the FACE table. The partition of the objects is higher and all of the surfaces have to be triangulated. Depending on the complexity of the surfaces (e.g., the number of windows on a wall), this triangulation may increase the number of databases. However, in the case of simple façades (e.g., without windows), the constant number of columns in the FACE table compensates for the number of elements increased for maintenance. Singularities are reduced relatively; i.e., the *node-on-face* and *arc-on-face* relationships are resolved.

2.2. Object-oriented models

The models mentioned are mapped in a relational DBMS, which is often considered less appropriate for describing real-world objects. Abdul-Rahman (2000) utilized the FDS model (Molenaar, 1998) to construct a 3D TIN based on spatial objects in an OO environment; i.e., by using the commercial OO DBMS, also known as the Persistent Object and Extended Technology (POET OO DBMS). The schema of the model is illustrated in Fig. 5, where 3D objects (such as boreholes) are represented by a series of 3D TINs primitives (i.e., tetrahedral). TIN nodes represent point objects, TIN edges represent area objects, TIN surfaces (triangles) represent area objects and 3D TINs represent solid objects. The model works with four spatial primitives (nodes, lines, surfaces, and solids).

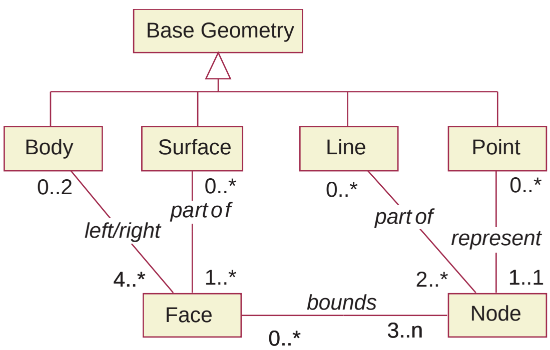

Fig. 4. Urban Data Model (UDM) (Coors, 2003).

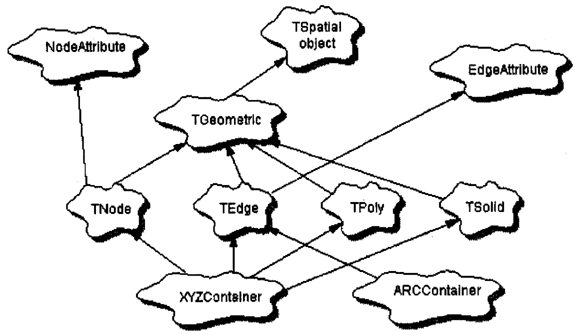

Fig. 5. 3D TIN-based OO model (Abdul–Rahman, 2000).

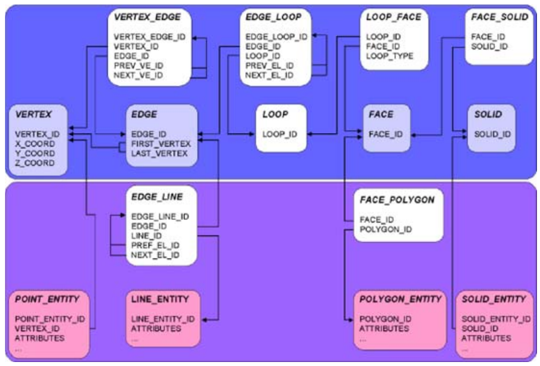

Fig. 6. SOMAS (Pfund, 2001).

Simple topological relationships between the primitives of the TIN-based objects can be established, such as point–line, point–surface, point–solid, line–surface, line–solid and surface–solid. All of the constructed classes of the model are then mapped according to the schema of the POET OO DMBS database.

Other solutions of explicitly structures-maintaining objects are presented by the Solid Object Management System (SOMAS) (Pfund, 2001) or by the model of de la Losa and Cervelle (1999). Figs. 6 and 7 show the conceptual models. The structures, however, are not implemented in a DBMS.

The authors of the OO model proposed the order of the *faces* with respect to a common edge to be explicitly maintained in the model. Thus, the normal vector of each *face* is determined by the direction of the edge and may not always be directed outside of the 3D object.

2.2.1. OO3D

Shi et al. (2003) developed an OO data model to handle complex 3D objects in GIS (OO3D) (Fig. 8). The conceptual OO3D data model is developed based on the principle of OO data modeling. This model is founded on the following three basic geometric elements: node, segment and triangle. The abstract geometric objects are defined accordingly. These include points, lines, surfaces and volumes. Second, the corresponding 3D logical model is designed based on the defined abstract objects and the relationships between them. Third, a formal representation of the 3D spatial objects is provided in detail. The model is applied in a piece of 3D GIS-developed software—SpaceInfo.

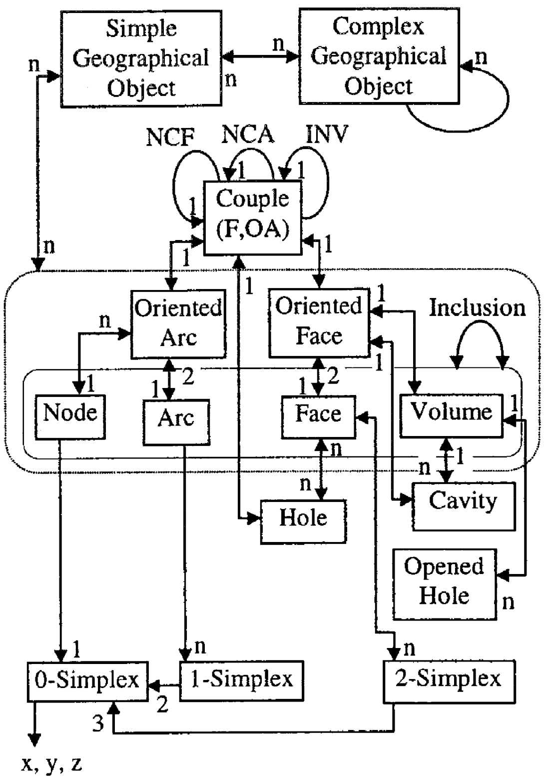

Fig. 7. OO-model of de la Losa and Cervelle (1999).

The proposed model can handle complex objects, such as complex buildings and TV towers, which is an essential function for the building of large-scale cyber cities. The proposed data model is proving to be very efficient, particularly in visualization and rendering. The experimental results of the model demonstrate more compacted data volumes and improved visualization speeds for 3D objects than the existing models.

2.3. *3D structures with explicit representations of relationships*

The second type of topological model has only one representative; i.e., the spatial model introduced by Brisson (1990) and extended by Pigot (1995). It is viewed as the tuple model. It defines cells and cell complexes by the fundamental properties of a manifold. The subdivision of the space is full. A clear separation between objects and primitives does not exist. The model is based on four primitives, called cells. The description of each cell gives the reference to all of the neighboring cells from all dimensions. The relationships (0: non-existent, 1: existent) can be organized in a table with four columns representing the four cells. For example, to describe a 3-cell cube, the table will contain 16 records. The initial works do not allow singularities, but in some further extensions of the model (Mesgari, 2000), singularities are permitted; for example, a 0-cell inside a 2-cell, a 2-cell inside a 2-cell (holes), a 3-cell inside a 3-cell (tunnels). To classify a spatial object, one should keep track of information on which cells belong to which objects. Under these circumstances, spatial objects can be described as a set of 3-cell, 2-cell, 1-cell and 0-cell tuples.

2.4. *Comparison of different models*

The advantages and disadvantages of a model change subject to application. For example, the arbitrary number of nodes per face can be seen as an advantage

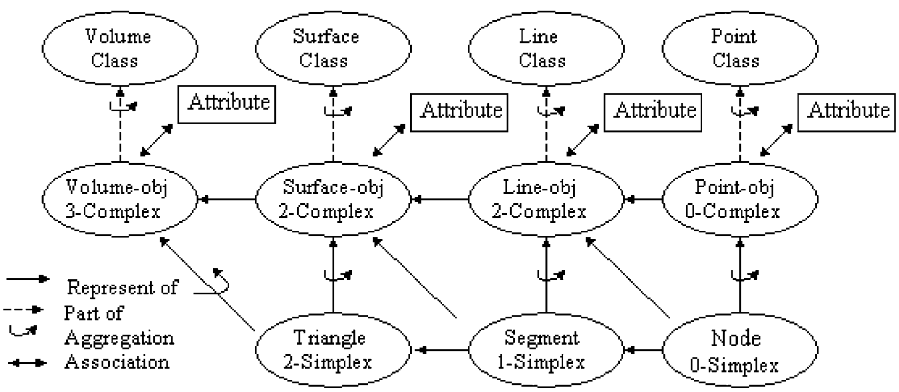

Fig. 8. OO3D model of Shi et al. (2003).

or disadvantage, depending on the applications. The modeling of complex 3D objects (e.g., buildings) is convenient, since an inappropriate partitioning (from the user's point of view) is not necessary and the faces on the boundary can represent a 3D object. However, the same freedom in face description may lead to problems in visualization, as the rendering engines can only handle triangles. Furthermore, the operators for consistency check become very complex. Another example is the face–body relationship. Navigating through 3D objects is easy, but in some cases (e.g., in urban areas) non-significant data may be stored (i.e., ''open air'' also has to be stored as a right body).

The major problem with TEN occurs at the stage of modeling. Since the space is completely subdivided into *tetrahedrons*, the interiors of objects (e.g., buildings), as well as open spaces, are also decomposed into *tetrahedrons*. Such subdividing hinders the formation of 3D man-made objects. Pilouk (1996) suggests that these objects be represented by 3D FDS features in TEN. However, the subdivision of *triangles* furnishes the data needed for displaying graphic information. In this respect, TEN and UDM are perhaps the optimal models for the visualization of surfaces. The maintenance of *triangles* solves other modeling problems such as holes or the explicit storage of relationships like *arc-on-face* and *node-on-face*. An additional disadvantage of TEN is its much larger database size compared with other representations, and the need to process *tetrahedrons*, which is not required for visualization.

The omission of *arcs* enables data structures (SSM, UDM) to benefit form the significantly faster data traverse. However, the navigation of rough *surfaces* (e.g., ''following the shortest path'') may become time-consuming. The representation of *bodies* as a set of *faces* (e.g., SSM) can extract the geometries of the objects, but navigational queries may be disturbed since the co-boundary of *face–body* relationships is not explicitly maintained (i.e., it has to be derived).

The cell tuple data structure provides the largest spectrum of topological relations between cells and complex cells. The model is built on solid mathematical foundations, thus promising easy maintenance. With respect to visualization, the extraction of faces and points is a simple operation, as the links between cells are stored explicitly. Data obtained from the tuple representation, however, lacks an indication of order. Supplementary records are needed to establish the clockwise or anti-clockwise order of cells (note that the cyclic order is ensured). Assuming a relational implementation, the entire body of tuple information is available in one relational table. On the one hand, there is no need to perform JOIN operations to select any data. On the other hand, the size of the table grows tremendously, which slows down the speed of SELECT operations. For example, the records for a simple box occupy twice as much space as in 3D FDS.

One of the major advantages of OO3D models, such as OO3D, is that they are capable of handling complex 3D objects. This further improvement is crucial, particularly for developing a cyber city for large cities where many complex objects exist, such as numerous buildings.

Some of the OO models are designed with compact characteristics, for example, the OO3D model has the basic elements of node, segment and triangle. This design differs from the TEN and 3D FDS models, which do not contain any arc elements, reducing data storage when spatial objects are constructed. However, the topological relationships are not stored explicitly. The performance of some of the spatial analysis-related applications may not be as efficient as that of other 3D models.

It is rather difficult to compare models using only the references in the literature. The topological models are implemented under different conditions, for example, different DBMSs and different server and client configurations, and tested with different data sets. Zlatanova (2000) presented performance tests for SSM and 3D FDS with respect to visualization queries. The queries can be 'translated' as 'extract and visualize all the objects according to a given condition (e.g., ID $<$ 100).' Tests are performed under the same computer configurations, DBMSs and data sets. The results of the tests showed that SSM gives significantly a better performance than FDS.

3. Spatial relationhsips frameworks

3.1. Frameworks for representing spatial relationships

Three different approaches to encoding spatial relationships are discussed in the literature; i.e., metric, topology and order. The metric is a pure computational approach based on the comparison of numerical values related to the location of the objects in space. For example, the spatial relationship between a house and a parcel (e.g., inside, outside, to the south) can be clarified by a point-in-polygon metric operation performed for each point constituting the footprint of the building. The order establishes a preference based on the mathematical relation ''$<$'' (strict order) or ''$\leqslant$''(partial order), which allows for a tree-like organization of objects. For example, if a building is inside a parcel, the spatial relationship is represented as ''building $<$ parcel.'' The applicability of representing spatial relationships has been investigated by Kainz (1989). Kainz argued that it has advantages in the expression of inside–outside relationships.

Topology allows the encoding of spatial relationships to be constructed on the neighborhoods of objects, regardless of the distance between them. The main property of topology, the invariance under topological transformations (i.e., rotation, scaling and translation), makes the computer maintenance of spatial relationships appropriate. The following section discusses the general framework of topology.

3.1.1. The 9-intersection model

The framework of the model (Egenhofer and Herring, 1990) utilizes the fundamental notions of general topology so that the topological primitives can investigate the interactions of the spatial objects. The topological primitives of a spatial object can be defined in each spatial model; hence, the framework can be applied to any spatial model. The basic criterion for distinguishing different relationships is the detection of empty and non-empty intersections between topological primitives. Depending on the number of topological primitives considered, two intersection models were presented in the literature. The first investigates the intersection of the interiors and boundaries of two objects. This results in a $2^4 = 16$ relationship between two objects. Exterior evaluation is adopted when two topological primitives are inadequate for differentiating many relations. In this case, the number of detectable relations between two objects increases to $2^9 = 512$. Eight relationships are possible between 3D and 3D objects, and they are given the following names: *disjoint, meet, contains, covers, inside, covered By, equal* and *overlap* (Fig. 9). For example, if the boundaries of the two objects intersect but the interiors do not, the conclusion is that the objects *meet*. Despite the criticism (i.e., that not all of the relationships are possible in reality, the intersections have not been further investigated and that many object intersections are topologically equivalent), the framework provides a systematic, easy-to-implement method of detecting spatial relations.

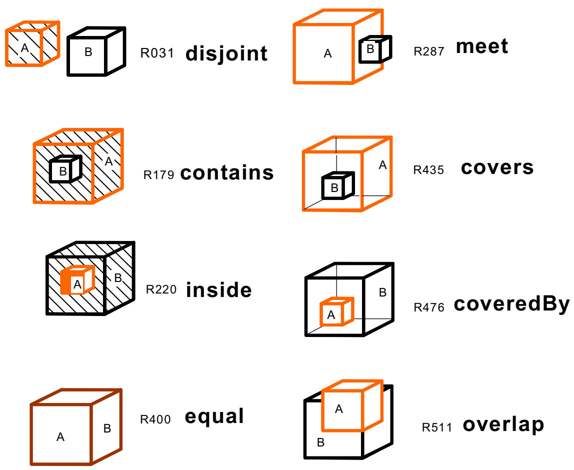

Fig. 9. 9-intersection model: possible relationships between 3D and 3D objects (Egenhofer and Herrring, 1990).

3.1.2. Voronoi-based spatial algebra for spatial relationships

Li et al. (2002) suggest a voronoi-based spatial algebra for spatial relationships. Appropriate operators from set operators are used in the solution to distinguish the spatial relationships between neighboring spatial objects. Three values (contents, dimensions and number of connected components) are employed as the computational results of the operation of the sets. The voronoi region of an object enhances the interaction of an object with its neighbors.

3.1.3. Uncertain topological relationships modeling

Shi and Guo (2002) presented a study of a formal representation of the topological relationships between uncertain spatial objects. First, after reviewing the related definitions and representation concerning uncertain spatial objects, they proposed a unified structure for representing certain or uncertain spatial objects. Second, they presented a framework to formally represent topological relationships between uncertain spatial objects. Third, they provided the algorithm to determine topological relationship.

3.1.4. Extended topological relationships in GIS

Liu and Shi (2003) proposed a further development on topological relations between any two objects in GIS. First, they adopted a new definition of the topological relations between two objects. Based on this new definition, the topology of the object itself and several topological properties (such as compactness, connectivity, first fundamental group, subspace topology, etc.), a sequence of topological relations between any two holeless objects is discovered. Based on the proposed extended topological relationships models, the number of topological relations between two infinites is found. These can then be approximated by a sequence of matrices. Furthermore, the topological relations between two convex sets can be approximated by a sequence of 4×4 matrices, which are the topological properties of $A^\circ \cap B^\circ$, $A^\circ \backslash B$, $B^\circ \backslash A$, $\partial A \cap \partial B$.

3.1.5. The dimensional model (DM)

The DM is another framework utilizing the order of points, which is related to the study of affine space (a subspace of the topological space) and convex shapes. The formal definition of the model can be found in Billen et al. (2002). Here, we will use a simple example for illustration. If one looks at a triangle in R^2, the points of order 0 are the vertices. The points on the edge have an order of 1 and the points of order 2 are all of the points that are "inside" the triangle. Applying this formalism, spatial objects can be described and their spatial relationships can be decoded. In the 3D Euclidean space (R^3), four types of dimensional elements are allowed; i.e., 0D, 1D, 2D and 3D elements. For example, a polygon has a 2D-element, a 1D-element and a 0D-element. The 2D-element coincides with the spatial object (i.e., the polygon). To represent the dimensional relationships between two objects, one has to consider all of the dimensional properties of these elements. For example, the dimensional relationships between two simple spatial objects of two dimension (i.e., polygons A and B) can be defined in the following order: first, check the dimensional relationship between the 2D element of A and all of the dimensional elements of spatial object B; then, check the dimensional relationship between the 1D element of A and all of the dimensional elements of spatial object B, etc.

The dimensional relationship can be *partial, total* or *non-existent*, depending on the interaction between the interiors of the objects (Fig. 10). The benefit gained from using these frameworks is flexibility while deciding on which dimensional elements are to be used. In general, a larger number of relationships can be distinguished compared to the 9-intersection model (see Fig. 11).

3.2. Spatial operators

Having specified the data structure and the framework for representing relationships, the next step is to define the operations to be supplied by the system. The operations describe all of the actions that can be performed on the data. First of all, operations to build

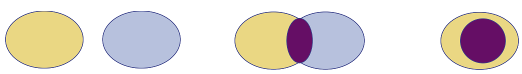

Fig. 10. Dimensional relationships: *non-existent, partial* and *total.*

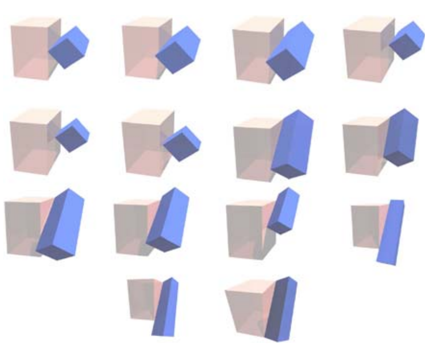

Fig. 11. Dimensional model: possible relationships between 3D and 3D objects.

a consistent data structure and to update it are investigated and developed. For example:

- operations to organize data according to the data structure; i.e., operations for planarity, convexity and discontinuity, as defined in the model;
- operations to check for consistency: the validation of the objects (e.g., polygon closed, body closed), node-on-line, node-on-face, node-in-body, line-on-face, line-in-body, intersection of lines, face-on-face, intersection of faces, face-in-body;
- 3D overlay, which is based on the same operation of consistency check and 3D editing;
- operation for 3D editing: the adding, deleting and updating of cells.

Apart from the constructing operators that have been mentioned, GISs facilitate a number of specialized operations such as selection, navigation and specialization. Molenaar (1998) described the GIS query as a selection operation with three components: data type specification, conditions and operations that have to be performed on the data. The selection can then be performed on semantics, geometry or topology. For example, "select the buildings (data type) higher than 15 m (condition) and show their ID (operation)." Sophisticated operations on data may obscure the boundary between query and analysis. Theoretically, an original operation and further processing can be encapsulated in a new operation. Many classifications of operations are included in the literature (Aronoff, 1995; Goodchild, 1987). In general, the operations can be subdivided in three large groups with respect to geometric and semantic characteristics and spatial relationships. Most interesting are the operations related to the geometry and to the spatial relationships. These operations can be classified as follows:

- metric operations are selection operations based on the shape and size of objects and on further computations; e.g., of distance, volume, area, length, center of gravity, intersect;
- position operations are selection operations based on position (without further processing); e.g., objects in a certain area;
- proximity operations are selection operations based on geometric characteristics and on the creation of new objects; e.g., a buffer, convex hull, union of objects;
- relationship operations are selection operations based on spatial relationships (without further processing); e.g., neighboring operations, overlay;
- network operations are selection operations based on spatial relationships and geometries, and on further processing with different levels of complexity; e.g., route planning;

- visibility operations are selections based on geometric characteristics and further processing; e.g., sign of view;
- semantic operations are selections based on semantic characteristics;
- mixed operations are selections founded on geometric and semantic characteristics.

Apparently, operations relating to the spatial relationships of the objects are highly influenced by data structure. As mentioned in the previous section, some of the structures may perform certain queries better than others. Moreover, it should be noted that spatial analysis can be performed on geometric models, as well. Many relational DBMSs offer support to spatial objects, especially geometric models; and supply a number of spatial operations, for example, validation, point-in-polygon, objects-within-distance, area, length, etc. The operations only make use of X and Y coordinates, although some of them accept 3D faces.

4. Conclusions

In this paper, we have given a brief overview on relational or OO topological models and discussed two frameworks for detecting spatial relationships between objects. With reference to the discussion on the advantages and disadvantages of the different models, we conclude that selecting an appropriate structure is a complex process and that the characteristics of the applications, for instance, objects of interest, resolution, required spatial analysis, etc., should be examined. A model that is good for 3D spatial analysis may exhibit a dissatisfactory performance on 3D visualization and navigation. Moreover, the relational or OO implementation of the model also has an impact on its performance.

Following the current trends for the integrated maintenance of spatial and non-spatial data, many DBMSs have already provided support to spatial objects. According to the abstract specifications of OpenGIS (Open GIS consortium Inc., 1999), spatial objects are stored in the database with their geometric and topologic representations to ensure consistency between the two models after conversion operations. This does not imply that all vendors have to accept one 3D topological model for implementation. As discussed above, different models may be suitable for the execution of specific tasks. Oosterom et al. (2002) proposed the maintaining of multiple topological models in one database by describing the objects, rules and constraints of each model in a metadata table. Such an approach will maximize efficiency and effectiveness in the provision of operations. Metric and position operations such as area or volume computations will be presented on the geometric model, while relationship operations such as "meet," and "overlap" will be performed on the topological model.

References

Abdul-Rahman, A., 2000. The design and implementation of two and three-dimensional triangular irregular network (TIN) based GIS, Ph.D. Dissertation, University of Glasgow, Glasgow, United Kingdom, 250p.

Aronoff, S., 1995. Geographic Information Systems: A Management Perspective. WDL Publications, Ottawa, 293p.

Billen, R., Zlatanova, S., Mathonet, P., Boniver, F., 2002. The dimensional model: a framework to distinguish spatial relationships. In: Richardson, D., van Oosterom, P. (Eds.), Advances in Spatial Data handling, 10th International Symposium on Spatial Data Handling. Springer, Berlin, pp. 285–298.

Brisson, E., 1990. Representation of d-dimensional geometric objects, Ph.D. Dissertation, University of Washington, Seattle, WA, 165p.

Carlson, E., 1987. Three dimensional conceptual modelling of subsurface structures. Technical Papers of American Society for Photogrammetry and Remote Sensing/American Congress on Surveying and Mapping Annual Convention, Baltimore, Vol. 4., pp. 188–200 (Cartography).

Coors, V., 2003. 3D GIS in Networking environments', Computer, Environment and Urbaqn Systems, Vol. 27 (4), pp. 345–357.

de Hoop, S., van de Meij, L., Molenaar, M., 1993. Topological relations in 3D vector maps, In: Proceedings of EGIS'93, Genoa, Italy, pp. 448–455.

de la Losa, A., Cervelle, B., 1999. 3D Topological modelling and visualisation for 3D GIS'. Computers & Graphics 23 (4), 469–478.

Egenhofer, M.J., Herring, J.R., 1990. A mathematical framework for the definition of topological relationships, In: Proceedings of Fourth International Symposium on Spatial Data Handling, Zurich, Switzerland, pp. 803–813.

Goodchild, M., 1987. A spatial analytical perspective on geographical information systems. International Journal of Geographical Information Science 1 (4), 327–334.

Grün, A., Wang, X., 1998. CC-modeller: a topology generator for 3D city models. ISPRS Journal for Photogrammtry and Remote Sensing 53 (5), 286–295.

Kainz, W., 1989, Order, topology and metric in GIS, American Society for Photogrammetry and Remote Sensing/American Congress on Surveying and Mapping Annual Convention, Vol. 4, Baltimore, pp. 154–160.

Li, Z., Zhao, R., Chen, J., 2002. A Vornori-based spatial algebra for spatial relations in GIS. Progress in Natural Science 12 (7), 528–536.

Liu, K.F., Shi, W.Z., 2003. Analysis of topological relationships between two sets. In: Shi, W.Z., Fisher, P.F., Goodchild, M. F. (Eds.), Proceedings of the Second International Symposium on Spatial Data Quality, March 19th–20th, pp. 61–71.

Mesgari, S.M., 2000. Topological cell-tuple structures for three-dimensional spatial data, ITC Dissertation number 74, ITC, The Netherlands, 200p.

Molenaar, M., 1990. A formal data structure for 3D vector maps. In: Proceedings of EGIS'90, Vol. 2. Amsterdam, The Netherlands, pp. 770–781.

Molenaar, M., 1998. An Introduction to the Theory of Spatial Objects Modelling. Taylor & Francis, London.

Oosterom, P.J.M.van, Stoter, J.E., Zlatanova, S., Quak, W.C., 2002. The balance between geometry and topology. In: Richardson, D., van Oosterom, P. (Eds.), Advances in Spatial Data Handling, 10th International Symposium on Spatial Data Handling. Springer, Berlin, pp. 209–224.

Open GIS Consortium, Inc., 1999. The OpenGIS abstract specification, topic 1: Feature geometry, Technical Report Version 4 (99-101.doc), Open GIS Consortium, 〈http://www.opengis.org/techno/abstract.htm〉.

Pfund, M., 2001. Topologic data structure for a 3D GIS. In: Proceedings of International Society for Photogrammetry and Remote Sensing Journal, Vol. 34. Part 2W2, May, Bangkok, Thailand, pp. 233–237.

Pigot, S., 1995. A topological model for a 3-dimensional Spatial Information System, Ph.D. Dissertation, University of Tasmania, Australia.

Pilouk, M., 1996. Integrated modelling for 3D GIS, Ph.D. Dissertation, ITC, The Netherlands.

Rikkers, R., Molenaar, M., Stuiver, J., 1993. A query oriented implementation of a 3D topologic data structure. In: Proceedings of EGIS'93, Vol.2. Genoa, Italy, pp. 1411–1420.

Shi, W.Z., Guo, W., 2002. Topological relationships between spatial objects with uncertainty. In: Shi, W.Z., Fisher, P.F., Goodchild, M.F. (Eds.), Spatial Data Quality. Taylor and Francs, London, pp. 50–61.

Shi, W.Z., Yang, B.S., Li, Q.Q., 2003. An object-oriented data model for complex objects in three-dimensional geographic information systems. International Journal of Geographic Information Science, Vol. 17 (5), pp. 411–430.

Shibasaki, R., Shaobo, H., 1992. A digital urban space model—a three dimensional modelling technique of urban space in a GIS environment. International Archives for Photogrammetry and Remote Sensing, Vol. XXIX. Part B4, Commission IV, Washington, DC, USA, pp. 257–264.

Zlatanova, S., 2000. 3D GIS for urban development. Ph.D. Dissertation, ITC, The Netherlands, 222pp.

Zlatanova, S., Abdul Rahman, A., Pilouk, M., 2002. 3D GIS: current status and perspectives. In: Proceedings of the Joint Conference on Geo-spatial theory, Processing and Applications, 8–12 July, Ottawa, 6p. CDROM.