ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume X-4/W2-2022
17th 3D GeoInfo Conference, 19–21 October 2022, Sydney, Australia

EFFICIENT SPATIO-TEMPORAL MODELLING TO ENABLE TOPOLOGICAL ANALYSIS

M.W. Jahn*, P. Kuper, M. Breunig

Karlsruhe Institute of Technology (KIT), Geodetic Institute, Karlsruhe, Germany
(markus.jahn, paul.kuper, martin.breunig)@kit.edu

Commission IV, WG IV/9

KEY WORDS: Spatio-Temporal Model, Spatio-Temporal Topology, Boundary Representation, City Model, CityGML

ABSTRACT:

Time-dependent analysis scenarios such as heat, wind or flood analysis in cities and in landscapes need a correct and consistent modelling of geometry and topology over time. However, hitherto efficient time-dependent geometry models and topological analysis based on a mathematically sound theory were neglected when modelling objects in the built and natural environment. This is surprising as incorrect topological relationships over time such as not fitting neighbourhoods of surfaces or solids inevitably lead to wrong analysis results. In this paper we propose the combination of a spatio-temporal geometry model together with a topological schema to provide accessible and consistent objects over time. Where an efficient spatio-temporal geometry model reduces redundant geometric data and enables spatio-temporal queries, an efficient topological model minimizes the number of relations as far as possible and enables robust topological queries. The geometry model uses the concepts of point tubes, delta storage as well as net components and pre- and post-objects to enable the change of geometry and topology over time for natural structures, e.g., digital terrain models (DTM). Geometry here are the boundary and interior coordinates of the objects whereas topology here is interpreted in a wider sense than only focusing on geometrically induced topology to maintain topological consistency by the management of incidence relations. In addition, the topological schema introduces three basic bidirectional relation types to manage aggregations, abstractions and incidences in order to provide a general abstract topological schema for the management of complex intra- and inter-related spatio-temporal objects to enable the modelling of consistent complex topology over time. Finally, a conclusion is given highlighting the applicability of the approach and future research.

1. INTRODUCTION

Efficient and consistent access to the geometry and to the topology of 3D shapes over time are an important precondition to enable advanced analyses in cities, landscapes or geological formations. Natural 3D shapes such geological solids are usually modelled by simplicial complexes, particularly triangles and tetrahedra. A major reason for this is the planarity of these elements in the respective dimension.

The use of simplicial complexes has its limitations and multiple drawbacks when used for built objects such as buildings or tunnels. For this reason, other data models such as CityGML and Industry Foundation Classes (IFC) are used for 3D city models and single buildings. Known problems are in particular the planarity of individual elements, such as house walls. Also, the topology or neighbourhood relationships between the individual elements can also be problematic. However, surfaces and solids are defined by polygons rather than by simplicial complexes.

The combination of natural and built objects such as landscapes and buildings is needed for multiple applications, e.g., wind or landslide simulations (Breunig et al., 2017). For the latter, spatial-temporal concepts are required to successfully manage such data. Therefore, the merging of models from both worlds is not trivial, but nevertheless an important task.

Earlier we proposed the combination of multiple concepts for an efficient handling of spatio-temporal data with a strong focus on the geometry of the data. Therefore, we combined the concepts of point tubes, delta storage, net components, and pre- and post-objects. The implementations are done within *DB4GeO* (Kuper, 2016), (Breunig et al., 2016), (Kuper, 2018).

As described in our previous work (Jahn et al., 2017) a spatial topology can be defined by spatial simplices on point tubes for moving and morphing geo-objects and a spatio-temporal topology can be defined by spatio-temporal polytopes wherever to use point tubes or not. A definition for topological consistency is also given in (Jahn et al., 2017). In (Jahn and Bradley, 2021) we focused on the spatial algorithms to create watertight volumetric models from *BREPs* (Boundary Representations). In (Jahn and Bradley, 2022) we focused on the topological algorithms to create watertight volumetric models from *BREPs*. This included more details on the implementation of a general topological model for geo-objects by introducing a *Property Graph Model* with three bidirectional relation types concerning aggregation, abstraction and incidence relations. This paper focuses on this abstract concept to establish a general topological model for spatio-temporal geo-objects and its usability in order to combine time-step based spatio-temporal geometric data modelling and 4D topological analysis through which these two worlds can be combined. Implementations are done within *DB4GeOGraphS* (Jahn, submitted in 2021), a graph-based extension of *DB4GeO*.

2. RELATED WORK

Usually geodata are modelled in 2D space, e.g., points, linestrings and polygons with 2D coordinates to model points of interests (POI), ways and areas. Such a data model is suitable for multiple applications and realized in the OGC standard

* Corresponding author

Simple Feature Access (Simple Feature Access Specification, 2011). But there are also several applications in 3D space and also temporal changes needed to be modelled accordingly. For instance, geological changes are usually modelled in continuous processes (Langran and Chrisman, 1988), (Le et al., 2013). However, when modelling the development of buildings or entire cities, discrete changes are in the focus of interest and have to be specified accordingly (van Oosterom and Stoter, 2010).

For the modelling of natural objects and, in particular, soil subsurfaces and all kinds of irregular surfaces simplicial complexes are frequently used, particularly for geoscientific applications. Natural objects are composed of *d-simplicial complexes*. Therefor the elements of the complexes are from the same dimension (Bär, 2007). According to Egenhofer et al. the model of simplicial complexes is a topological data model suitable for the management of spatial data (Egenhofer et al., 1990). Initially, the model was used within a 2D space. Later, the model was also used in 3D space (Alms et al., 1998).

The handling of time within spatio-temporal data led to the development of temporal geographic information systems (TGIS) (Worboys, 1994), (Güting and Schneider, 2005). When the processed data are of geoscientific origin, such a system is called a spatio-temporal geoscience information system (TGSIS) (H.H., 2014). TGSIS usually work with spatio-temporal data based on d-simplicial complexes that describe continuous phenomena.

Spatial and spatio-temporal databases should offer an efficient and user-friendly data management (Worboys and Duckham, 2004), (Menninghaus et al., 2016), (Breunig et al., 2016). Schäben et al. developed a spatio-temporal DBMS that is based on the object-relational geodatabase PostGIS (Le et al., 2013), (H.H., 2014), (Le et al., 2014), (Weihed, 2015), (Gabriel et al., 2015). The DBMS manages spatio-temporal data based on simplicial complexes and follows the snapshot model. Breunig at al. developed an object-oriented approach of a DBMS for spatio-temporal data based on polytope complexes (Breunig et al., 2016).

The topology of data based on simplicial complexes is of interest for multiple applications and therefore forms its own field of research (Salnikov et al., 2018), (Thomsen et al., 2008). This paper focuses on the development of real 4D topology for efficient spatio-temporal data management.

3. GEOMETRY MODEL

We developed an efficient spatio-temporal data model based on the concepts point tubes, delta storage, net components, and pre- and post-objects.

The primitives of our data model are points, segments, triangles and tetrahedra, i.e., d-simplicies that form d-simplicial complexes with $d \in \{0, 1, 2, 3\}$ in a 3D space. Such a d-simplicial complex is considered as a 3D model which can be extended into a 4D model (3D + time) when multiple 3D models are used as time steps (Langran and Chrisman, 1988):

Definition 3.1. *Let $m(t_n)$ be the 3D model for time t_n at time step n. Then we define:*

$$m_n := m(t_n) \times \{t_n\}$$
$$m(t_n) + m(t_{n+1}) := m_n \cup m_{n+1}$$

Then the 4D model is defined as:

$$m := \bigcup_{n\in\mathbb{N}} m_n = \sum_{n\in\mathbb{N}} m(t_n)$$

The *Point Tube Concept* separates the topology of a net from the vertices of a d-simplicial complex (Kuper, 2010), (Breunig et al., 2013), (Breunig et al., 2016). These vertices are managed in so-called *Point Tubes*, a special tube-like data structure. Due to the *Delta Storage Concept*, stationary vertices are referenced and not stored again.

When modelling the continuous movement of natural structures based on simplicial complexes, subregions of the model are usually more volatile than others. There are situations where parts of a 4D model $m = \{A(t_0, t_1, t_2, t_3), B(t_0, t_3)\}$ have different temporal discretization, e.g., part A is realized for the time steps t_0, t_1, t_2 and t_3 while part B is realized of time steps t_0 and t_3. Such situations must be handled properly. The problem exists at the border regions of such parts. It must be ensured that the topology of the simplicial complex is not "torn apart". The so-called net components were developed and combined with the *Point Tube Concept* (Kuper, 2018). The net components correspond to parts of a 4D model with a shared temporal discretization. For every time step of the respective net component, the vertices are explicitly handled in point tubes, while the net topology of the simplicial complex is handled separately. When there is a request for the model at a specific time, e.g., $m(t_x)$, with $t_0 \leq t_x \leq t_3$, the corresponding vertices are calculated by interpolating the point tubes of all net components involved. Successive time steps are examined for similarities to reuse such parts across multiple time steps (Strathoff, 1999) and (Siebeck, 2003). This concept led to a reduction of storage space, the acceleration of spatial and spatio-temporal operations and extended the field of applications.

Furthermore, 4D models which follow the snapshot model can change their net topology depending on time. Such a change occurs, for instance, when the model is developed anew for a given time step (Lautenbach and Berlekamp, 2002). Sometimes the net topology is refined in order to reflect certain aspects of changes. Therefore, a concept was developed that is based on the ideas of (Polthier and Rumpf, 1994) and is capable of handling such net topology changes by the use of pre- and post-objects which exist at the same time-step, equal in shape but with a different net topology.

The geometry model uses the spatial net topology on top of point tubes by introducing the net components and ensures that the spatio-temporal neighbour-objects of those net components are topologically consistent. The implementations lead to an implicitly defined topology of the corresponding polytope complexes of the net components. An explicit defined topology of polytope complexes is provided in the following.

3.1 Aggregations

Based on the spatio-temporal model described in (Jahn et al., 2017) different kinds of spatio-temporal aggregation types exist e.g., *Spatial4DCollection*, *Net4D*, *Component4D*, *Sequence4D* and *Element4D* where each higher aggregation level consists of objects of the lower aggregation level (see Figure 1).

An *Spatial4DCollection* object manages four different *Net4D*s separated by their dimension (for points, curves, surfaces and

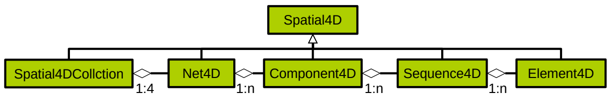

Figure 1. Spatio-temporal geometry types (Jahn and Bradley, 2021)

volumes). This is the highest aggregation level and is basically used as an object to structure results of any geometric algorithm. The following aggregation levels exist for each dimension separately. A *Net4D* object is the topological sum of *Component4D* objects. A *Component4D* object is an implementation of a polytope complex. Therefore, it is a collection of topologically consistent *Sequence4D* objects and realizes most of the basic operations concerning polytope complexes. A *Sequence4D* object is a temporally ordered collection of topologically consistent *Element4D* objects, the polytopes, which represent a moving or morphing d-dimensional simplex between two time steps, as described by (Polthier and Rumpf, 1994). Therefore, the d-dimensional *Sequence4D* object describes the trajectory of a d-dimensional simplex. A d-dimensional *Sequence4D* object also provides a list of neighbouring *Sequence4D* objects to link its maximal $(d+1)$ neighbouring *Sequence4D* objects which share the same temporal discretization, just like a d-simplex-element in 3D space is linked with its $(d+1)$ d-dimensional neighbours on each of the $(d+1)$ $(d-1)$-dimensional border simplices, the gluing objects to build manifolds. The interpretation of dimension d in the context of *Sequence4D* objects is given by the moving and morphing d-dimensional simplex, e.g., an *Triangle4DSequence* object is geometrically spoken 3-dimensional, an ordered collection of prisms in 4D space, but the moving triangle is 2-dimensional just as a moving point (e.g a point tube) is geometrically spoken 1-dimensional, but the moving shape is 0-dimensional. A detailed description can be found in (Rolfs, 2005).

This is an explicitly defined topology for polytope complexes, since each *Element4D* object is explicitly modelled. But the previously described concepts of point tubes, delta storage, pre- and post-objects, and net components can be applied to retrieve the spatio-temporal shapes of each aggregation type.

3.2 Borders

The spatio-temporal model handles border operations differently than the spatial model. Where the spatial model provides only one operation to retrieve the border geometries, the spatio-temporal model provides three operations to retrieve all parts of the border of a *Spatial4D* object. Figure 2 demonstrates the spatio-temporal objects and their border.

The first operation is called *getStart()* and retrieves the *Spatial3D* object of the first time step. If the *Spatial4D* object is a *Net4D* or a *Spatial4DCollection*, every *Component3D* object of the first time steps of each *Component4D* is collected into a *Net3D* object and returned. In case of *Component4D*, *Sequence4D* and *Element4D*, the spatial representation of the first time step is returned. The second operation is called *getEnd()* and returns the last time step(s) just as the *getStart()* operation. The third operation, called *getBorder()*, returns the spatio-temporal border of the *Spatial4D*. This operation returns null if the *Spatial4D* object is a *Sample4D*. This operation will return an array of *Point4DSequences* or *Point4DElements* if the *Spatial4D* object is of type *Curve4D*. This operation will return an array of *Segment4DSequences* or *Segment4DElements*

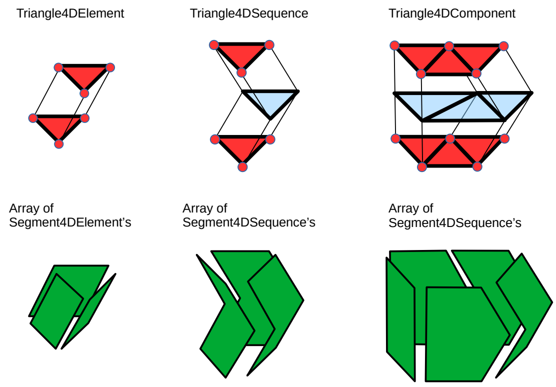

Figure 2. Three *Surface4D* aggregation levels and their border objects (red and green)

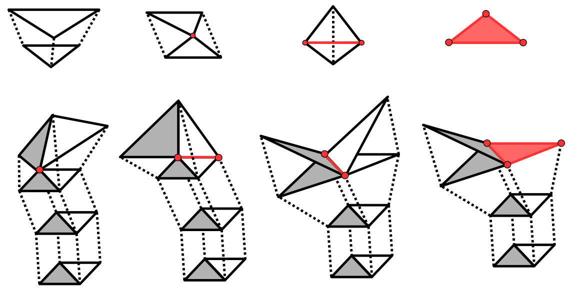

Figure 3. Triangle-polytope (top) spatial redundancies (red) and triangle-polytope complex (bottom) spatial redundancies (red).

if the *Spatial4D* object is of type *Surface4D*. And finally, this operation will return an array of *Triangle4DSequences* or *Triangle4DElements* (prisms) if the *Spatial4D* object is of type *Solid4D*.

3.3 Redundancy and splitting

For polytope complexes, there are a couple of different ways of dealing with spatio-temporal changes technically, as mentioned in (Jahn et al., 2017). One way leads to temporal redundant referencing of temporal-sequences or redundant referencing of equal temporal-sequence parts. Another way leads to spatial redundant referencing of the same simplex (or parts of it) within the polytope as pre- and post-simplices (or pre- and post-parts of it, respectively). In general, redundancies are always to be expected at the spatio-temporal boundary polytopes to which at least one neighbour does not change spatially, since the common points of the boundary polytope do not spatially change with the neighbours that do not change spatially.

Figure 3 shows possible spatial redundancies of a triangle-polytope (top) and a triangle-polytope complex (bottom). The spatio-temporal volume of the spatially redundant triangle polytope (right, top) is the area of the triangle times the length of the associated time interval. The spatio-temporal area of its edge segment polytope is the length of the respective triangle edge segment times the length of the associated time interval, etc.

Figure 4 shows different splitting methods when a triangle polytope sequence is extended by a triangle polytope. Each splitting method yields different spatio-temporal redundancies and different spatio-temporal topological settings (and inconsistencies) of the resulting triangle-polytope complexes.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W2-2022-137-2022 | © Author(s) 2022. CC BY 4.0 License.

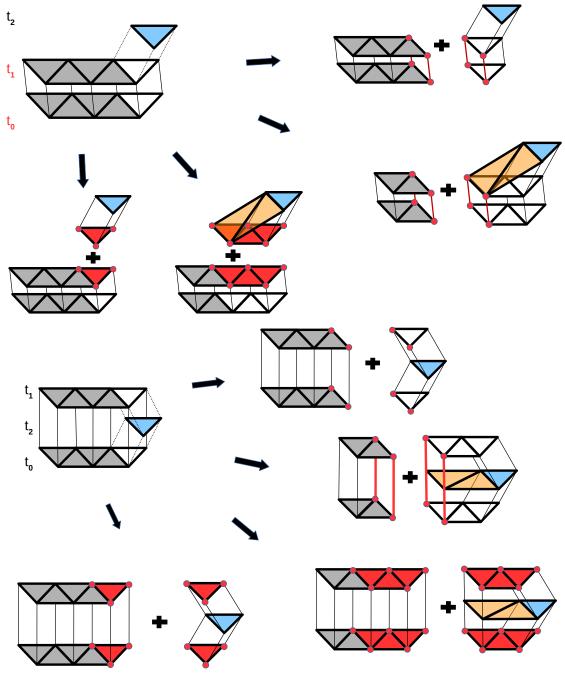

Figure 4. Split methods of a triangle-polytope complex when a triangle-polytope sequence is extended by a triangle-polytope.

4. TOPOLOGICAL MODEL

As described in (Jahn, submitted in 2021), the object model of *DB4GeOGraphS* is based on the *Property Graph Model* and OGCs *Simple Feature Model* and *General Feature Model* design patterns (see Figure 5). Every node of a graph within the *Property Graph Model* carries properties. The property types have been chosen from *Simple Feature Model* and *General Feature Model* where every feature is build up by a spatial part/property, a temporal part/property and a thematic part/property.

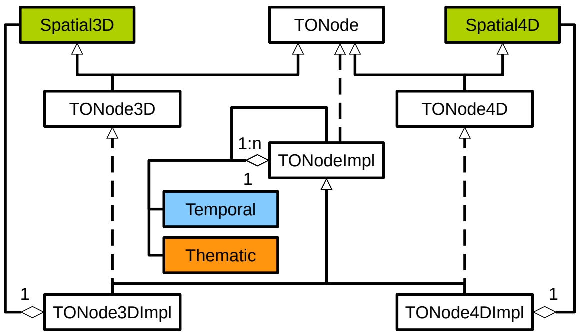

Figure 5. *Property Graph Model*, which follows *OGCs General Feature Model* (Jahn and Bradley, 2021)

The topological model uses three basic relation types (*part-of*, *border-of* and *generalization-of*, where *border-of* links a border object into the direction of an interior object e.g., an edge with its bounded faces) and their inverses (i.e., *composite-of*, *inner-of* and *specialization-of*, where *inner-of* links an interior object into the direction of a border object e.g., a face with its bounding edges). The choice of using exactly those three basic relation types is inspired by the object-oriented programming paradigm which uses abstractions and aggregations, and the incidence graph to model topological consistence. Therefore, modelling complex geo-objects where each part is topologically connected by incidence (e.g., a building as a solid with an awning as surface and an antenna modelled as curve), or aggregated to a more complex geo-object or interconnected to other geo-objects is possible. Even abstractions are possible (e.g., Level of Detail).

Figure 6 (top: spatial model, bottom: spatio-temporal model) illustrate two examples using *DB4GeOGraphS* model. It is to mention that all relations may be bidirectional or unidirectional. Figure 6 also illustrates the α_1- and α_2-involutions known from *G-Maps* (Lienhardt, 1991). An α_1 involution is applied by exchanging the dissimilar point ID of a segment with the dissimilar point ID of its neighbouring segment. This results in a movement along the border of C_0 or C_{12}. An α_2 involution is applied by exchanging the C_* IDs. This results in a movement from surface to surface using their common edge.

As already mentioned in (Jahn and Bradley, 2021), five basic node operations have been implemented within the spatial model which create (a) an aggregation node, (b) a number of sub-component nodes, (c) overlay nodes, (d) border nodes, or (e) a $d+1$-dimensional node from a d-dimensional node. The last operation is discussed in detail within (Jahn and Bradley, 2021) and (Jahn and Bradley, 2022).

4.1 Aggregation relations part-of and composite-of

As described in Section 3.1, the spatio-temporal properties yield five different aggregation levels. If the spatio-temporal property of a node within the *Property Graph* is a *Net4D* object, aggregation relations may be build to nodes which carry the contained *Component4D* objects. For each spatio-temporal aggregation level, *TONode4D* objects may be created in the same way.

A spatio-temporal predicate can be used to adjust the aggregation level which will be connected. This concept has the advantage to minimize the *Property Graph* if some aggregation levels are not needed to be part of the *Property Graph*.

4.2 Incidence relations border-of and inner-of

As described in Section 3.2, the spatio-temporal properties yield different incidence levels. In case of $d = 0$, borders do exist, the start and end points of the point tube. A $(d-1)$-dimensional border of the d-dimensional border does not exist because all d-dimensional border properties are closed (without boundary) e.g a moving and morphing triangle ($d = 2$) represented as prism in 4D space has a d-dimensional border which is a closed (without boundary) surface.

The border objects themselves have a certain aggregation level. A spatio-temporal predicate can also be used here to adjust the aggregation level which will be connected but as incidence relation in order to minimize the *Property Graph* as needed.

4.3 Abstraction relations generalization-of and specialization-of

Functionally independent relations can be created under this kind of relation type. It is the relation type where none of the previously discussed relation types can be applied. The abstraction relation has been applied to relate the results of the node operation which builds $d+1$-dimensional inner of an input node, which carries a d-dimensional *Net3D* object as spatial property. Figure 7 shows an example. It is clear to see that the border

https://doi.org/10.5194/isprs-annals-X-4-W2-2022-137-2022 | © Author(s) 2022. CC BY 4.0 License.

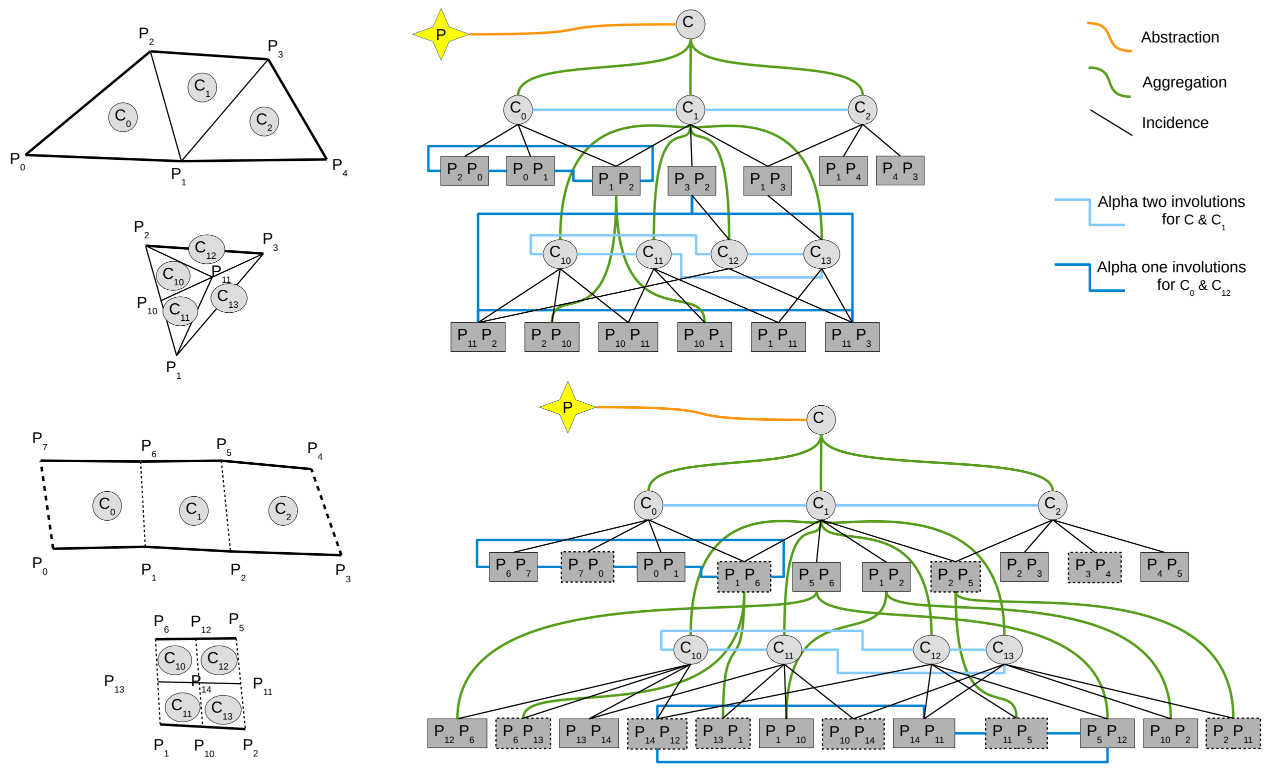

Figure 6. Example of a *Triangle3DComponent* object (top) and *Segment4DComponent* object C (bottom) using *Property Graph Model* together with a selection of *G-Maps* involutions (blue).

of the *Tetrahedron3DComponent* object which will be a closed (without boundary) *Triangle3DComponent* object (hull) is not equal to the *Triangle3DNet* object and that the *Triangle3DNet* object is not retrievable from the *Tetrahedron3DComponent* object by creating the border nodes. The algorithm looses input information (e.g., the shapes of the topologically inconsistent input *Triangle3DNet* object) and adds information by transforming a surface type into a solid type with information about inner points and outer points which are not distinguishable by the *Triangle3DNet* object which *Triangle3DComponent* objects are not well-connected to form one (without boundary) *Triangle3DComponent* object as one hull like the border of the *Tetrahedron3DComponent* object.

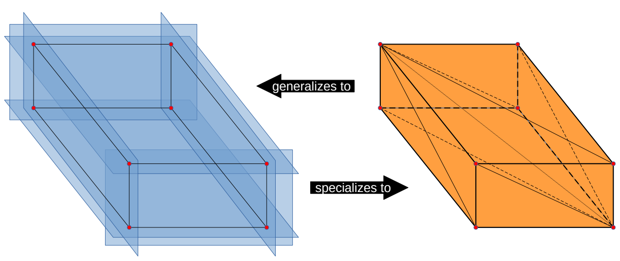

Figure 7. abstraction relation; BLUE: topologically inconsistent *Triangle3DNet* object made off six *Triangle3DComponent* objects; BLACK: surface intersections (left) triangle borders (right); ORANGE: *Tetrahedron3DComponent* object as result of the tetrahedralization

4.4 Spatial overlay spaces

The creation of topologically consistent overlay spaces is not a trivial task. Examples for spatial overlay spaces have been illustrated in (Jahn, submitted in 2021). A major issue there is the creation of d-dimensional objects from their *BREPs* in order to calculate the intersections between each d-dimensional object and not only the intersections of their boundaries by the use of the *BREP*-Geometries only. The major problem here is computational geometry to calculate the topologically consistent overlay space.

Figure 8 gives an example for a 3D (spatial) and 0D (temporal) topology using the above graph schema for a CityGML extraction of the city centre of Erfurt, Thuringia (Germany). The example itself does not have a geo-scientific background. The purpose is only the illustration. Surfaces for four different time steps were derived by elevating the z coordinates of the triangles from the Digital Terrain Model (DTM) which consist of points lower than some limit. The transformed triangles were glued together to shape simplicial complexes and collected into different surface nets for different limits to act as water surface snapshots for the different time steps.

Figure 9 shows the topology as a graph. The top presents the CityGML tree (black) together with the CityGML polygons (green head, yellow body). The CityGML polygons are grouped by the *BuildingPart/Building*-Tags (cyan head, yellow body), triangulated (cyan head, green body), tetrahedralized to tetrahedron complexes (green head, cyan body). Those tetrahedron complexes in turn were intersected with the DTM (green head, green body) and the four water surface nets (cyan head, green body) to create the overlay space (bottom) by adding the intersection geometries which are *part-of* two geometries (e.g., the water surface of some time step and a tetrahedron complex).

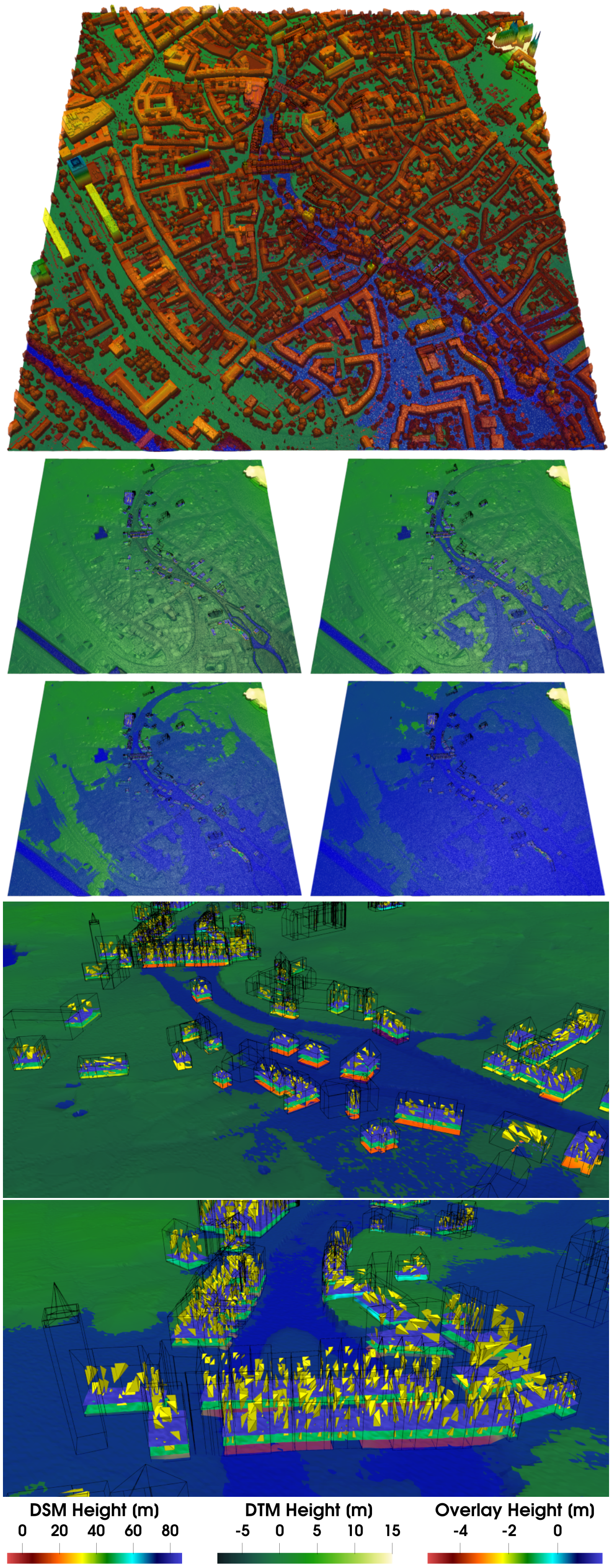

Figure 8. Digital Surface Model (DSM, coloured by height), Digital Terrain Model (DTM, coloured by height), CityGML (black), tetrahedralized CityGML (yellow) with shrunk tetrahedra, water surface (blue) at different times and intersection-geometries for each time (coloured by height).

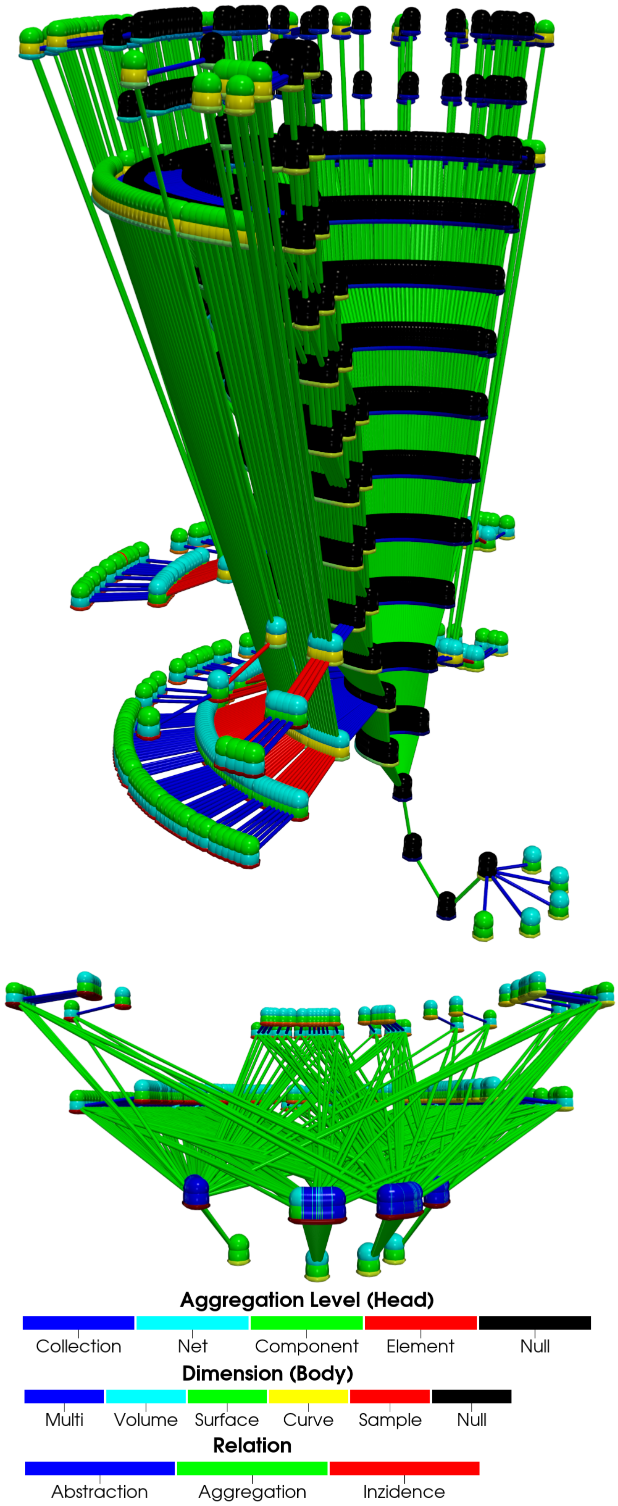

Figure 9. Topology of the geometries shown in Figure 8. Top: CityGML tree (black) with polygons as BREPs (green head, yellow body), grouped by *BuildingPart/Building*-Tag (cyan head, yellow body), triangulated (cyan head, green body), tetrahedralized (green head, cyan body) together with the DTM (green head, green body) and the four water surface nets (cyan head, green body). Bottom: DTM and the three water surface nets, which intersect the tetrahedralized CityGML. Intersection-nodes are *part-of* two nodes, respectively.

In DB4GeO, the net topology was separated from the vertices when handling 4D models. Therefore, the net topology can be reused for multiple time steps and needs to be combined with the vertices handled by point tubes when a representation of the model for a specific time is needed.

The topologies of the surfaces at each time step is derived from the DTM. But the graph of Figure 9 looks the same, since the *Point Tube Model* is evaluated at the different time steps by the use of some spatio-temporal interpolation function to create the four different water surfaces. In the example, the differences of the DTM and the water surfaces are the final minimized surfaces, which do not share a single triangle in 3D space.

4.5 Spatio-temporal overlay spaces

The previous time step based modelling of a spatio-temporal object by using snapshots is a problem, if a snapshot is not presented for the time step where the spatio-temporal object may intersect another object. This is especially the case if it is unknown when a moving and morphing object will intersect another object. Furthermore, if additional time steps need to be evaluated by the use of some spatio-temporal interpolation function, the topology of the snapshots may matter, e.g., a simple re-triangulation of a surface in 3D space can result in many shapes. The described geometry model is dealing with these issues.

A topological consistent overlay space of a set of spatio-temporal polytope complexes is created by intersecting the spatio-temporal polytope complexes, just like the creation of the topological consistent overlay space of spatial simplicial complexes. As previously described, the spatio-temporal core of *DB4GeOGraphS* is based on polytope complexes, which explicitly manages the inherent topology of the polytopes. The water surfaces of the previous example can be collected into one spatio-temporal polytope-complex by creating four copies of the DTM and exchanging the z-Coordinates as needed. The spatio-temporal triangle-polytope complex represents a moving and morphing surface, its elements, the polytopes are prisms in 4D space. Prisms are "solids" in 4D space. Therefore, the intersection object may also be a "solid", with a volume and all intersections can be found by the use of a computational geometry core which is able to deal with polytopes in 4D space without the try to numerically find the right times and snapshots of the spatio-temporal water body.

Since a computational geometry core to process polytope complexes efficiently in 4D space is not present, the example of moving and morphing surfaces represented as a prism complexes can be projected into the same temporal hyperplane and subdivided into a set of tetrahedron complexes to make use of a computational geometry core which is able to deal with simplicial complexes in 3D space. This step needs to care of prism complexes in 4D space, which do not produce a single tetrahedron complex in 3D space. This happens when any triangles of different times intersect, even implicit triangles interpolated from any "triangle tube" (*Triangle4DSequence* object) of the prism complex (*Triangle4DComponent* object or net component). Despite this over-complication, the situation reminds on the problem when dealing with *BREPs* where the intersection of objects within a *BREP* needs to be calculated. And in turn, the model of (Polthier and Rumpf, 1994) can be seen as a *BREP* model in 4D space, which manages the border of spatio-temporal objects only by memorizing multiple snapshots of large geo-objects and carefully interlinks the elements of those snapshots for spatio-temporal consistency purposes.

5. CONCLUSION AND OUTLOOK

The use of the point tube concept leads to an increase in efficiency and lower storage requirement for 4D models, which consist of multiple time steps (3D + 0D). The concept can easily be combined with the *Delta Storage Concept* and *Net Component Concept*. This paper combines the prior work, focusing on the efficient geometry handling, with topologically consistent spatio-temporal polytopes to establish a topology model in 4D (3D + 1D).

Therefore, the paper provided a graph schema to manage the topology of complex spatio-temporal geo-objects. The geometry is managed by *Point Tubes* or by explicitly defined polytope complexes in 4D space. The introduction of the three basic relation types makes it possible to manage complex geo-objects with their internal and external topology. The evolved topological spaces can be analysed by some graph query language (e.g., intersections are *part-of* two or more geo-objects). Furthermore, an explicit consistent topological model of the involved geo-objects by the creation of the overlay space reduces pre-processing time for repeating topological queries.

Nevertheless, the example motivated a spatio-temporal computational geometry to calculate the intersections and differences of polytope complexes. Geometry computations (e.g., intersection or difference) of moving and morphing n-dimensional polytope complexes need to be reduced to the computation of n-cell complexes by subdividing the polytopes into its simplicial representation. An explicit subdivision is necessary to clearly define the inner of the polytopes which are *BREPs*, just like the polygons and finally the buildings of a CityGML dataset, due to multiple ways of subdivisions by the use of different triangulation or tetrahedralisation methods which may lead to different shapes of the geo-object itself.

In our future research work, we will also focus on the role of topology within simulations such as heat propagation models in cities. Furthermore, distributed computing of geometric and topological database queries on big spatial data will be part of our investigations.

ACKNOWLEDGEMENTS

The Thuringian state office for land management and geo-information (TLBG) is thanked for their public geo-data repository. This work is partially funded by the Deutsche Forschungsgemeinschaft under grant numbers BR 2128 / 18-1 and BR 3513 / 12-1.

REFERENCES

Alms, R., Balovnev, O., Breunig, M., Cremers, A., Jentzsch, T., Siehl, A., 1998. Space-time modelling of the Lower Rhine Basin supported by an object-oriented database. *Physics and Chemistry of the Earth*, 23(3), 251-260.

Breunig, M., Borrmann, A., Rank, E., Hinz, S., Kolbe, T., Schilcher, M., Mundani, R.-P., Jubierre, J. R., Flurl, M., Thomsen, A., Donaubauer, A., Ji, Y., Urban, S., Laun, S., Vilgertshofer, S., Willenborg, B., Menninghaus, M., Steuer, H., Wursthorn, S., Leitloff, J., Al-Doori, M., , Mazroobsemnani, N., 2017. Collaborative multi-scale 3D city and infrastructure modelling and simulation. *The International Archives of the Photogrammetry,* *Remote Sensing and Spatial Information Sciences*, XLII-4/W4, 341–352. https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-4-W4/341/2017/.

Breunig, M., Butwilowski, E., Golovko, D., Kuper, P., Menninghaus, M., Thomsen, A., 2013. "Advancing DB4GeO", Progress and New Trends in 3D Geoinformation Sciences. 193-210.

Breunig, M., Kuper, P., Butwilowski, E., Thomsen, A., Jahn, M., Dittrich, A., Al-Doori, M., Golovko, D., Menninghaus, M., 2016. The Story of DB4GeO - A Service-Based Geo-Database Architecture to Support Multi-Dimensional Data Analysis and Visualization. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117, 187–205.

Bär, W., 2007. Verwaltung geowissenschaftlicher 3D Daten in mobilen Datenbanksystemen. PhD thesis, Universität Osnabrück.

Egenhofer, M. J., Frank, A. U., Jackson, J. P., 1990. A topological data model for spatial databases. A. P. Buchmann, O. Günther, T. R. Smith, Y.-F. Wang (eds), *Design and Implementation of Large Spatial Databases*, Springer Berlin Heidelberg, Berlin, Heidelberg, 271–286.

Gabriel, P., Gietzel, J., Le, H. H., Schaeben, H., 2015. Gst: A network based datastore for geoscience data and geomodels and its implementationpromin…s contribution towards interoperability.

Güting, R. H., Schneider, M., 2005. *Moving objects databases*. Elsevier.

H.H., L., 2014. Spatio-temporal Information System for the Geosciences: Concepts, Data models, Software, and Applications. PhD thesis, Technische Universität Bergakademie Freiberg.

Jahn, M., submitted in 2021. Distributed & Parallel Data Management to Support Geo-Scientific Simulation Implementations. PhD thesis, Karlsruhe Institute of Technology.

Jahn, M., Bradley, P., 2021. Computing watertight volumetric models from boundary representations to ensure consistent topological operations. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, VIII-4/W2-2021, 21–28. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/VIII-4-W2-2021/21/2021/.

Jahn, M., Bradley, P., Al Doori, M., Breunig, M., 2017. Topologically consistent models for efficient big geo-spatio-temporal data distribution. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-4/W5, 65–72.

Jahn, M. W., Bradley, P. E., 2022. A Robustness Study for the Extraction of Watertight Volumetric Models from Boundary Representation Data. *ISPRS International Journal of Geo-Information*, 11(4). https://www.mdpi.com/2220-9964/11/4/224.

Kuper, P., 2010. Entwicklung einer 4D Objekt-Verwaltung für die Geodatenbank DB4GeO. Diplom thesis, University of Osnabrück.

Kuper, P., 2016. Spatio-Temporal Data Handling for Generic Mobile Geoinformation Systems. PhD thesis, Karlsruher Institut für Technologie.

Kuper, P. V., 2018. Efficient and practical handling of spatio-temporal data based on time-dependent net components. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4, 321–327. https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-4/321/2018/.

Langran, G., Chrisman, N., 1988. A Framework for Temporal Geographic Information. *CARTOGRAPHICA*, 25 No. 3, 1-14.

Lautenbach, S., Berlekamp, J., 2002. Datensatz zur Visualisierung der Zentraldeponie Piesberg in Osnabrück.

Le, H. H., Gabriel, P., Gietzel, J., Schaeben, H., 2013. An object-relational spatio-temporal geoscience data model. *Computers Geosciences*, 57, 104-115.

Le, H. H., Schaeben, H., Jasper, H., Görz, I., 2014. Database versioning and its implementation in geoscience information systems. *Computers Geosciences*, 70, 44-54.

Lienhardt, P., 1991. Topological models for boundary representation: a comparison with n-dimensional generalized maps. *Computer-Aided Design*, 23(1), 59-82.

Menninghaus, M., Breunig, M., Pulvermller, E., 2016. High-Dimensional Spatio-Temporal Indexing. *Open Journal of Databases (OJDB)*, 3, 1-20.

Polthier, K., Rumpf, M., 1994. A concept for time-dependent processes. Goebel et al. (ed.), *Visualization in Scientific Computing*, Springer, Vienna, 137–153.

Rolfs, C., 2005. Konzeption und Implementierung eines Datenmodells zur Verwaltung von zeitabhängigen 3D-Modellen in geowissenschaftlichen Anwendungen. Diplom thesis, Hochschule Vechta.

Salnikov, V., Cassese, D., Lambiotte, R., 2018. Simplicial complexes and complex systems. *European Journal of Physics*, 40(1), 13.

Siebeck, J., 2003. Concepts for the Representation, Storage, and Retrieval of Spatio-Temporal Objects in 3D/4D Geo-Information-Systems. PhD thesis, Faculty of Mathematics and Natural Sciences at the Rheinischen Friedrich-Wilhelms-Universität, Bonn.

Simple Feature Access Specification, 2011.

Strathoff, F., 1999. Speichereffiziente Verwaltung zeitabhängiger Geometrien für ein GeoToolKit. PhD thesis, University of Bonn.

Thomsen, A., Breunig, M., Butwilowski, E., Broscheit, B., 2008. *Modelling and Managing Topology in 3D Geoinformation Systems*. 229–246.

van Oosterom, P., Stoter, J., 2010. 5d data modelling: Full integration of 2d/3d space, time and scale dimensions. S. I. Fabrikant, T. Reichenbacher, M. van Kreveld, C. Schlieder (eds), *Geographic Information Science*, Springer Berlin Heidelberg, Berlin, Heidelberg, 310–324.

Weihed, P., 2015. *3D, 4D and Predictive Modelling of Major Mineral Belts in Europe*. Ed. Springer International Publishing.

Worboys, M., 1994. A Unified Model for Spatial and Temporal Information. *Comput. J.*, 37, 36-34.

Worboys, M., Duckham, M., 2004. GIS, a Computing Perspective. 448p.