Topological queries in a space partitioning model: Definition, visualization and exports of results

Amelia Städtler[1] and Maximilian Sternal[1]
[1]Chair of Computing in Civil Engineering, Technische Universität Berlin, Germany
E-mail(s): amelia@die-fabelhafte-welt-der-amelia.de, maximilian.sternal@tu-berlin.de

Abstract: A modern building model is usually digital (BIM) and contains the geometry of the elements and spaces. The topological relationships of their elements, on the other hand, are often insufficiently described. However, many civil engineering applications do rely on the topological relationships between modeled elements. Two examples for that being energy modeling and the computer assisted navigation through indoor spaces. This paper investigates how topological queries can be run on an exemplary polyhedral space partitioning model. Therefore, a brief introduction to the principles of polyhedral space partitioning is presented. Current research approaches for topological models, spatial information and graph-based building models are reviewed. Topological queries are presented that derive neighboring relations from the investigated polyhedral space partitioning model. These queries are subsequently used deriving and visualization graph models that represent the full topology of the space partitioning model. Finally, the topologic information stored in the space partitioning model is exported into data structures with a focus on topological aspects: the Building Topology Ontology (BOT) and the OGC IndoorGML standard. Benefits and limitations of the space partitioning model, in regard to the export to such data structures, are discussed. And outlook is given on future possibilities of connecting geometric description, semantic data and topological and spatial information based on polyhedral space partitioning.

Keywords: Data Modeling, Topological Data Structures, Building Topology Ontology (BOT), IndoorGML

1 Introduction

Digital building models include geometric and semantic information of various kinds, for example materials and technical features. The topological relationships of their elements, on the other hand, are often insufficiently described. This is pointed out, for example, by Diakité and Zlatanova [1], for the prominent BIM standard Industry Foundation Classes (IFC). However, many civil engineering applications do rely on the topological relationships between modeled elements. One example for that being the computer assisted navigation through indoor spaces [2]. Graph-based models are an essential tool for describing relationships in digital workflows in the architecture and engineering industry (AEC). One relationship, investigated in this paper, is the spatial configuration of “topological primitives” like edges or cells. Such objects are part of geometric descriptions in geo-information-systems (GIS) and building information models (BIM), but the neighbourhood configurations between objects are often insufficiently described. However, it is possible to model a holistic geometry including topological relationships using a polyhedral space partitioning (*PPSpace*).

The *PPSpace* [3] project comprises a modeling kernel that creates and modifies a polyhedral space partitioning model, explicitly storing neighborhood relations of topological objects. Queries are implemented and described in this paper for a *PPSpace* space partitioning model. An example for such a query would be: “Return all cells that are neighbors of and share a face with this cell”. It shall be shown that, based on these queries, the export of a *PPSpace* space partitioning model to existing topological data structures in a civil engineering context, such as *BOT*, is possible.

This paper demonstrates how topological information can be exported from such a partition model to existing graph data structures. In the following chapter, the background for graph-based application models in AEC is outlined and related work of topological data structures for computational geometry is reviewed. Central features of the presented polyhedral space partitioning approach are introduced. It is shown how queries to the space partition can return topological relations and how these can be used to build graphs. Subsequently, these queries are used to export topological information, with a minimal subset of semantics, from a partition model to established topological data models of the AEC industry. Finally, an outlook to future applications and extensions to this concept is given.

2 Topological Queries

Topological queries are traditionally performed over spatial databases, which store primarily "regions" as two-dimensional geometry [4]. Spatial predicates can be computed by testing to regions against each other. Romanschek et al.[5] presented a novel approach based on space partition in the two-dimensional space. While this concept has been proven to be robust and extensible to the three-dimensional space [6], [7], the data structure used in this paper is not computed from an existing B-Rep, but constructed from scratch and topological queries can be performed without geometrical computation. The following chapters describe related approaches to topological information and introduces the main features of the *PPSpace data model* and its relations.

2.1 Related Work

Topology, as described by Paul [8], stems from the mathematical theory of continuous functions and denotes the connectivity between, for example, the rooms of a building and their links like doors or walls. Paul outlines the connection between CAD applications and the underlying topological concepts of topological space and homeomorphism of spatial data. A very similar approach is taken by the *topologic.app* tool [9]. It provides methods for visual programming to evaluate non-manifold cell complexes in terms of topological relationships.

For geo-information models, Clementini et al. [4] identify the need for GIS databases to be able to process queries with “topological constraints”. Daum and Borrmann [10] have presented a spatial query language that is based on the well-known 9-intersection matrix to analyse a given boundary representation using geometric calculations in order to describe e.g. "touch" and "contain" relationships between objects. However, such computations are complex and must be performed again after every modification. Research projects for 3D cadastral models have recently presented approaches for including spatial configurations, e.g. [11]–[13], which highlights the existing need for better capabilities or representing topology in AEC on the building and city scale.

2.2 Polyhedral Space Partition

The *PPSpace data model*, presented in [3], is a polyhedral space partitioning that structures the point set of Euclidian space. This point-set topology can be modified using atomic merge and split operators [14]. The partition system consists of two parts, an non-oriented user model and a core model. The user model represents a point-set topology and consists of nodes, edges, faces and cells, each defined by their boundaries. These domains represent manifolds that can be non-convex, multiply-connected and unbounded, as the Euclidean space is unbounded as well. The geometry of this model is specified by point coordinates, directions and planes. The core model is oriented and stores topological references. This concept builds upon existing approaches such as Radial Edge[15], Partial Edge[16] and Dual-Half Edge[2] and thereby combines features of space partitioning and topological data structures. Oriented "arrow bundles" and "twin facets" provide links to compute outer and inner polygons, radial and dihedral cycles and neighbouring or consisting cells.

2.3 Relations

Through the internal structure of the core model, topological information can be derived directly, without any geometric calculations. This is supported by the separation of geometrical and topological data, which are each represented by a different sub-model. Queries can be run on a partition model that identify neighboring relations between user model elements. The actual topological relations are determined in the core model, while the super-ordinate queries on the user model serve as an interface for other applications and tasks.

Comic and de Floriani [17] exhaustively describe existing topological data structures and relations. A relation is classified $R_{j,k}(P)$, where P is the parameter to start the query, e.g. an edge or cell, j and k indicate the dimension of the start and target of the relational contact query. A contact query for *adjacent* domains of the same dimension (j = k) would be to return all cells in contact with one specific cell ($R_{3,3}(C)$). The implementation in this paper extends this concept by defining an additional dimension for the contact, to obtain also all cells that share at least one edge $R^1_{3,3}(C)$ or node $R^0_{3,3}(C)$. Furthermore, queries for *boundary* relations (j < k) are specified to return, e.g. all nodes in the boundary of a face $R_{2,0}(F)$ or all edges *connected* (j > k) through a node $R_{0,1}(N)$.

Contact queries are often describe to only return elements that are in contact with the parameter. However, during the export of graph data structures the need arose to also obtain the model elements through which contact elements are connected to the parameter, because this information is right at hand when the query is run, but difficult to obtain later.

3 Export

Based on the implemented topological queries (as described in section 2), links and paths within the model can be calculated. A partition model can be translated to a graph that contains topological information. This is achieved by translating model objects to graph vertices. Topological relations between these model objects are represented by graph edges, which connect the parameter object of a topological query to its result objects. Because the user model serves as an interface of the space partition, graphs are determined only for user model objects while the calculation relies on the links in the underlying core model.

Figure 1 shows an example of adjacent boxes (A-E), surrounded by two other cells (X-Y) that are eventually embedded in a partitioned unbounded outer space (c0X). The graph in figure 2 symbolizes the topological relations of connected cells. The generic open-source graph framework *JGraphT* is used to export and visualize semantic-free *GraphML* graphs. The following chapter will extend this concept by mapping and classifying cells according to existing engineering standards.

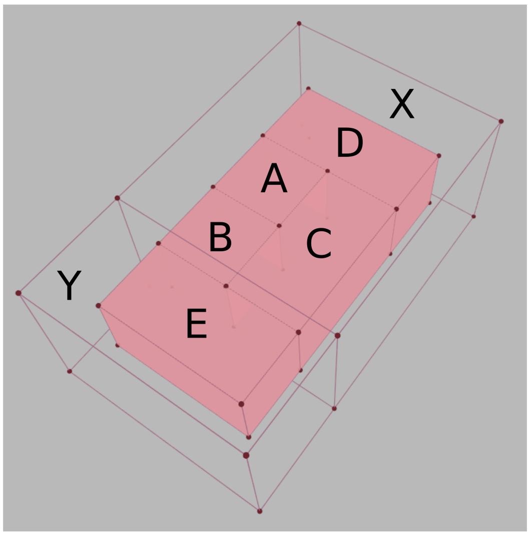

Figure 1: A test fixture for the derivation of graphs. The boxes A, B, C, D and E are encased in the boxes X and Y.

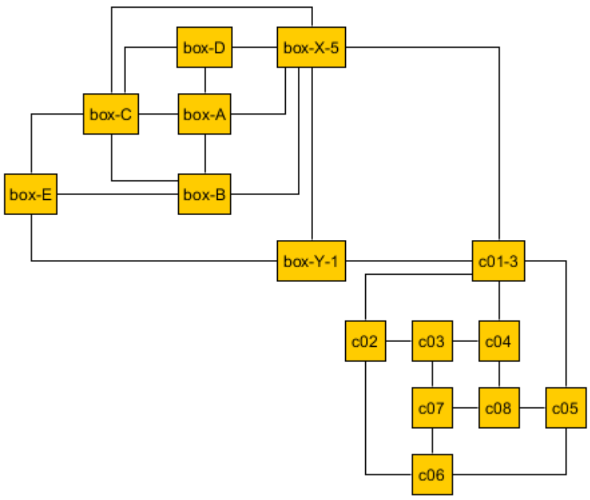

Figure 2: A graph representing topological relations of the test fixture. The graph vertices represent cells, a graph edge implies a common face. $R^2_{3,3}$

3.1 Building Topology Ontology

The *Building Topology Ontology* (BOT) is an *OWL* ontology that describes topological relationships specifically in a building context [18]. As an ontology, BOT represents semantic information. A *PPSpace* partition model contains geometrical and topological information about its objects, but it cannot be inferred if the object represents a building element, a building space or something completely different. It is therefore necessary to wrap a partition model in an export model that contains the BOT classifications for the relevant partition model objects. The BOT classes [18] considered in this paper are:

- **Space**: has a 3D extend with physical or conceptual meaning
- **Element**: building elements like walls, beams etc.
- **Interface**: qualifies the relationships of Spaces and Elements

BOT furthermore defines properties, which mostly describe topological relations between classes and entities [18]. The partition model contains such information. This paper investigates how these properties can be automatically derived from a partition model. Adjacency, expressed by the properties *adjacentZone* and *adjacentElement*, can be derived from a partition model through topological queries. The result of a topological query are partition model objects that are adjacent to the parameter object.

A partition model object separately refers to its inner boundary elements by referencing a single element of each inner region in the core model. Therefore, containment, expressed by the properties *containsZone, hasSpace, hasElement and containsElement*, can also be inferred from the partition model. The other elements belonging to an inner region can be determined through model traversal, for example a depth-first search. In a partition model, contained elements are always also adjacent elements. This is acceptable because adjacency and containment properties are not disjoint in BOT. However, an element that is connected to the boundary of its encasing element is considered adjacent, but not contained. This might not always be a good estimation, especially in civil engineering where, for physical reasons, everything is usually connected to the floor (for example a column in a room).

Intersection, expressed by the properties *intersectsZone* and *intersectingElement*, is strictly excluded from a partition model and can therefore not be exported to BOT. An implementation is introduced that derives a BOT ontology from a *PPSpace* partition model, according to the ideas presented above. The BOT ontology is written to a Turtle file. Figure 3 shows the visualization of a partition model. Figure 4 shows a graph visualization of the BOT ontology that was derived from this partition model, using the implementation. The graph visualization was produced using the *SPARQL* visualizer provided by Mads Holten Rasmussen [19].

This shows that the topological relations defined as properties in BOT, particularly adjacency and containment, can be automatically exported from a *PPSpace* partition model, using the implemented topological queries. Intersection cannot be exported, because it is not covered by the partition model. However, a valid BOT ontology can still be created without.

3.2 IndoorGML

IndoorGML is an 'open data model and XML schema for indoor spatial information'. It features a cellular space model with additional semantic, geometrical and topological information [20]. The cellular space consists of identifiable cells that do not intersect, which resembles the structure of the partition model. The central aspect of an IndoorGML model is the *Structured Space Model (SSM)*. A graphical representation is depicted in figure 6. The SSM distinguishes between the primal and the

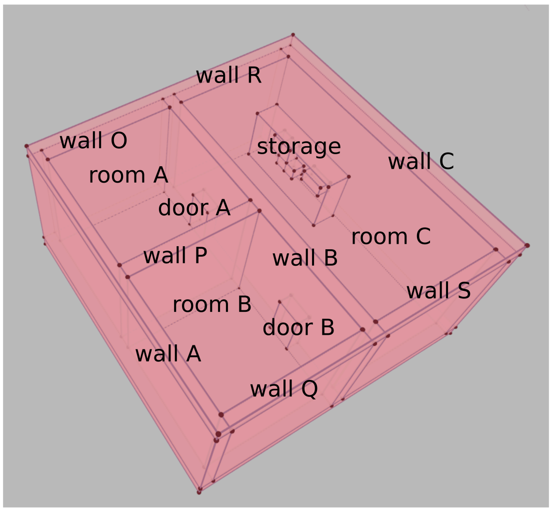

Figure 3: A visualization of the model for testing the BOT export.

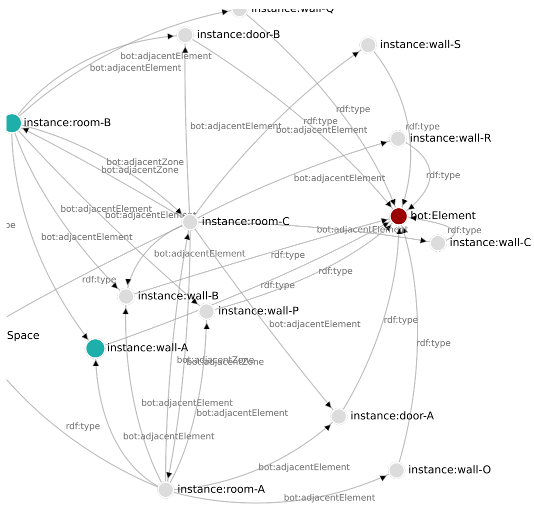

Figure 4: The exported ontology, loaded in the *SPARQL* visualizer tool.

dual space, and therein between geometrical and topological data. The primal geometry is stored in the polyhedral partition model. The dual space is represented by Node Relation Graphs (NRG). A NRG is based in the Poincaré-duality and represents three-dimensional objects through nodes and their common boundaries through edges [20]. Figure 5 shows a prototype visualization of primal and dual geometry for the special case of convex polytopes.

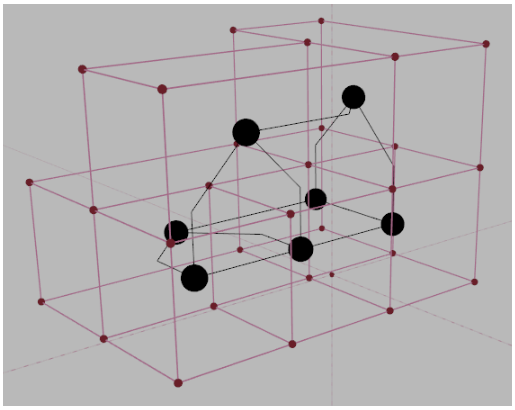

Figure 5: Primal and dual geometry of a test fixture.

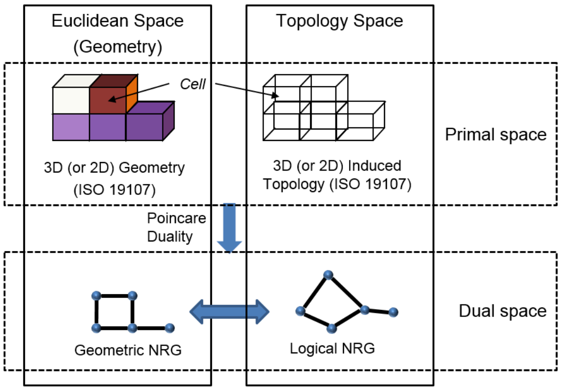

Figure 6: An overview over the Structured Space Model, as employed by *IndoorGML*. From: [20])

The dual topology, represented by a 'logic NRG', can be inferred from a *PPSpace* partition model. using the concepts for graph derivation presented in chapter 2. Contact queries $R_{3,3}(C)$ are able to determine cells that share a common boundary with the parameter 'C'.

4 Outlook

The implemented topological queries are able to derive topological information from a partition model without geometrical calculations. A large number of related applications can be based on the topological queries. The export to BOT and IndoorGML is generally possible and convenient when topological queries are used to derive topological information from the partition model. Currently, only test models have been analyzed and only a subset of its classes and properties have been considered in the export to BOT. Further implementation and validation of exported models must be performed to assess the usability of the presented approach. Open questions regarding geometrical robustness are investigated in current research.

As space partitioning maps the complete space including unbounded domains, this approach would be able to connect indoor and outdoor models. Several research projects aim to align building standards such as the IFC with urban models and navigation [1], [13], [21]. Multiple shortcomings of topological relations and geometric inconsistencies have been found and any alignment can benefit strongly from a holistic modeling of space and its relations.

References

[1] A. Abou Diakité and S. Zlatanova, “Valid Space Description in BIM for 3D Indoor Navigation:” *International Journal of 3-D Information Modeling*, vol. 5, 2016.

[2] P. Boguslawski and C. Gold, “The Dual Half-Edge—A Topological Primal/Dual Data Structure and Construction Operators for Modelling and Manipulating Cell Complexes”, en, *ISPRS International Journal of Geo-Information*, vol. 5, no. 2, 2016.

[3] P. J. Pahl, “Polyhedral Partition of Euclidean Space”, TU Berlin, (draft), 2021.

[4] E. Clementini, J. Sharma, and M. J. Egenhofer, “Modeling Topological Spatial Relations: Strategies for Query Processing”, en, p. 15, 1994.

[5] E. Romanschek, C. Clemen, and W. Huhnt, “A Novel Robust Approach for Computing DE-9IM Matrices Based on Space Partition and Integer Coordinates”, *ISPRS International Journal of Geo-Information*, vol. 10-11, 2021.

[6] J. Vetter and W. Huhnt, “Accuracy Aspects when Transforming a Boundary Representation of Solids into a Tetrahedral Space Partition”, in *Proceedings of the EG-ICE 2021 Workshop on Intelligent Computing in Engineering*, 2021.

[7] M. W. Jahn and P. E. Bradley, “A Robustness Study for the Extraction of Watertight Volumetric Models from Boundary Representation Data”, *ISPRS International Journal of Geo-Information*, vol. 11, no. 4, 2022, Publisher: MDPI.

[8] N. Paul, “Applications of continuous functions in topological CAD data”, *arXiv:1308.0256 [cs]*, [Online]. Available: http://arxiv.org/abs/1308.0256 (visited on 12/08/2021).

[9] W. Jabi and A. Chatzivasileiadi, “Topologic: Exploring Spatial Reasoning Through Geometry, Topology, and Semantics”, in *Formal Methods in Architecture, Springer*, 2021.

[10] S. Daum and A. Borrmann, “Processing of Topological BIM Queries using Boundary Representation Based Methods”, *Advanced Engineering Informatics*, vol. 28, 2014.

[11] U. Ujang, F. Anton Castro, and S. Azri, “Abstract Topological Data Structure for 3D Spatial Objects”, *ISPRS International Journal of Geo-Information*, vol. 8, 2019.

[12] L. Knoth, B. Atazadeh, and A. Rajabifard, “Developing a new framework based on solid models for 3D cadastres”, *Land Use Policy*, vol. 92, 2020.

[13] R. Jaljolie, K. Riekkinen, and S. Dalyot, “A topological-based approach for determining spatial relationships of complex volumetric parcels in land administration systems”, *Land Use Policy*, vol. 109, 2021.

[14] V. Galishnikova and W. Huhnt, “Polyhedral space partitioning as an alternative to component assembly”, in *ECPPM 2021 – eWork and eBusiness in Architecture, Engineering and Construction*, CRC Press, 2021.

[15] K. J. Weiler, *Topological structures for geometric modeling (Boundary representation, manifold, radial edge structure)*, Aug. 1986.

[16] S. H. Lee and K. Lee, “Partial entity structure: A compact non-manifold boundary representation based on partial topological entities”, en, in *Proceedings of the sixth ACM symposium on Solid modeling and applications - SMA ’01*, Ann Arbor, Michigan, United States: ACM Press, 2001.

[17] L. Čomić and L. D. Floriani, “Modeling and manipulating cell complexes in two, three and higher dimensions”, in *Digital Geometry Algorithms*, Springer, 2012, pp. 109–144.

[18] M. H. Rasmussen, M. Lefrançois, G. F. Schneider, and P. Pauwels, “BOT: The building topology ontology of the W3C linked building data group”, *Semantic Web*, vol. 12, pp. 143–161, 2021, Publisher: IOS Press.

[19] Rasmussen, Mads Holten. “Sparql-visualizer | bot: The basics”. (2021), [Online]. Available: https://w3c-lbd-cg.github.io/bot/tutorial/ (visited on 12/12/2021).

[20] J. Lee, K.-J. Li, S. Zlatanova, *et al.*, “Ogc® indoorgml 1.1”, Open Geospatial Consortium, Tech. Rep. 19-011r4, version 1.1, 2020. [Online]. Available: http://www.opengis.net/doc/IS/indoorgml/1.1 (visited on 12/29/2021).

[21] J. Yan, S. Zlatanova, and A. Diakité, “A unified 3D space-based navigation model for seamless navigation in indoor and outdoor”, *International Journal of Digital Earth*, vol. 14, 2021.