

A Spatial Reasoning Framework Based on Non-Manifold Topology

Nicholas Mario Wardhana¹, Wassim Jabi², Aikaterini Chatzivasileiadi³, Nikoleta Petrova⁴

^{1,2,3,4} Welsh School of Architecture, Cardiff University

^{1,2,3,4} {wardhanan|jabiw|chatzivasileiadia|petrovan}@cardiff.ac.uk

Keywords: *Non-Manifold Topology, Topologic, Spatial Reasoning, Pathfinding*

Abstract

Non-Manifold Topology (NMT) has been previously proven to be an appropriate representation of interconnected architectural and spatial structures. This paper further explores the suitability of NMT for spatial reasoning purposes. A literature survey is done to identify the necessary components of a spatial reasoning framework, and an adaptation of such framework based on NMT is presented. This paper also describes the implementation of an NMT-based spatial reasoning framework, its integration into the Topologic software library which the authors develop, as well as the implementation challenges. Finally, a study case on pathfinding on an NMT model, which has been generated from a Building Information Modelling (BIM) structure, is presented and analysed.

Introduction

The concept of architectural space is essential in building design, representation, and reasoning (Ching 2014). It has been argued that before the fabrication models are produced, buildings are typically perceived as a set of interconnected spaces (Curtis 1996). Seen from this perspective, spatial reasoning provides information about the relationship among the spaces, the spatial information attached to every individual space, as well as with the objects and individuals occupying or surrounding the spaces. Such information can be used as feedback to improve the building design, as well as to provide live information to the user of the building e.g. in the presence of a hazardous situation.

Building design and analysis workflows have traditionally used the concept of manifold topology, which centres around the idea of the boundaries of a three-dimensional entity separating the interior from the exterior. In contrast to this, Non-Manifold Topology (NMT) offers a formal and consistent topological framework which allows mixed-dimensional entities and junctions on the boundaries. This, therefore, allows internal subdivisions of a space separated by internal boundaries. As a consequence, the adjacency information among the spaces as well as other entities is explicitly stored in the building model and spatial queries can be conveniently performed. While NMT is commonly considered as defects in current practice, it has been shown that NMT can better represent the concept of interconnected spaces and support spatial as well as topological queries (Aish and Pratap 2013; Jabi et al. 2017). There have been precedents in the use of NMT for spatial reasoning. For example, spatial reasoning on NMT has been used to perform social sustainability analysis on vernacular courtyard houses and tall buildings (Al-Jokhadar 2018).

Due to the inherent support for spatial queries, it is hypothesised that the use of NMT in spatial reasoning will enhance the current building design and analysis workflows. It is therefore crucial to devise a spatial reasoning framework that is built on top of the concept of NMT. This paper seeks to build a foundation to meet this aim and presents two objectives. Firstly, a review will be conducted on academic papers as well as software libraries on spatial information typically used in the architectural design processes, as well as existing spatial reasoning frameworks and algorithms. Secondly, a spatial reasoning framework based on the concept of Non-Manifold Topology will be presented and established from the review.

Non-Manifold Topology

The NMT framework presents a formal and consistent relationship among the multi-dimensional constituent entities. The authors' earlier work (Chatzivasileiadi, Wardhana, et al. 2018) identified eight entities, namely (sorted in descending order based on the dimensional complexity) Cluster, CellComplex, Cell, Shell, Face, Wire, Edge, and Vertex. The hierarchy of these entities is shown in Figure 1. It is possible to navigate from one entity to another with higher or lower level of dimensionality. Boolean operations can be used to modify NMT models, and the resulting model will retain the non-manifold property.

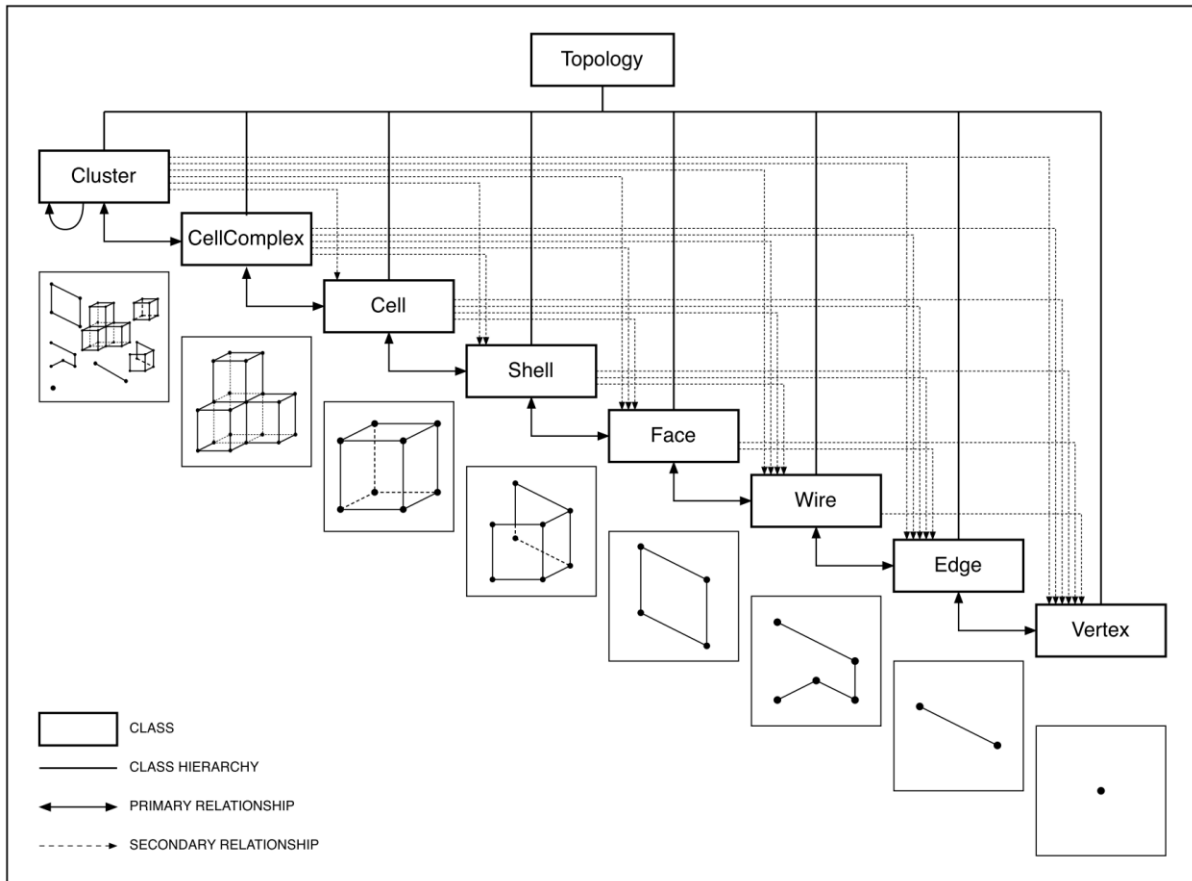


Figure 1. The class hierarchy of the Non-Manifold Topology entities

The authors have developed a software library called Topologic (<https://topologic.app/>) on the basis of NMT to facilitate the design and analysis of architectural models (Aish et al. 2018; Jabi et al. 2018). This library is implemented with a multi-layer architecture. The core package, written in C++, contains topological classes and functionalities, including navigation queries. A higher-level layer is written on the .NET platform on top of the core package and it serves as plugins to host applications, including Dynamo (Figure 2) and Grasshopper. These plugins wrap the classes and functionalities in the core package, as well as provide two-way mechanisms to convert an entity between the geometric representations in the host applications and the topological representations in the library. In the earlier publications (Chatzivasileiadi, Lannon, et al. 2018), the authors have presented workflows to use this library in building performance and structural analyses.

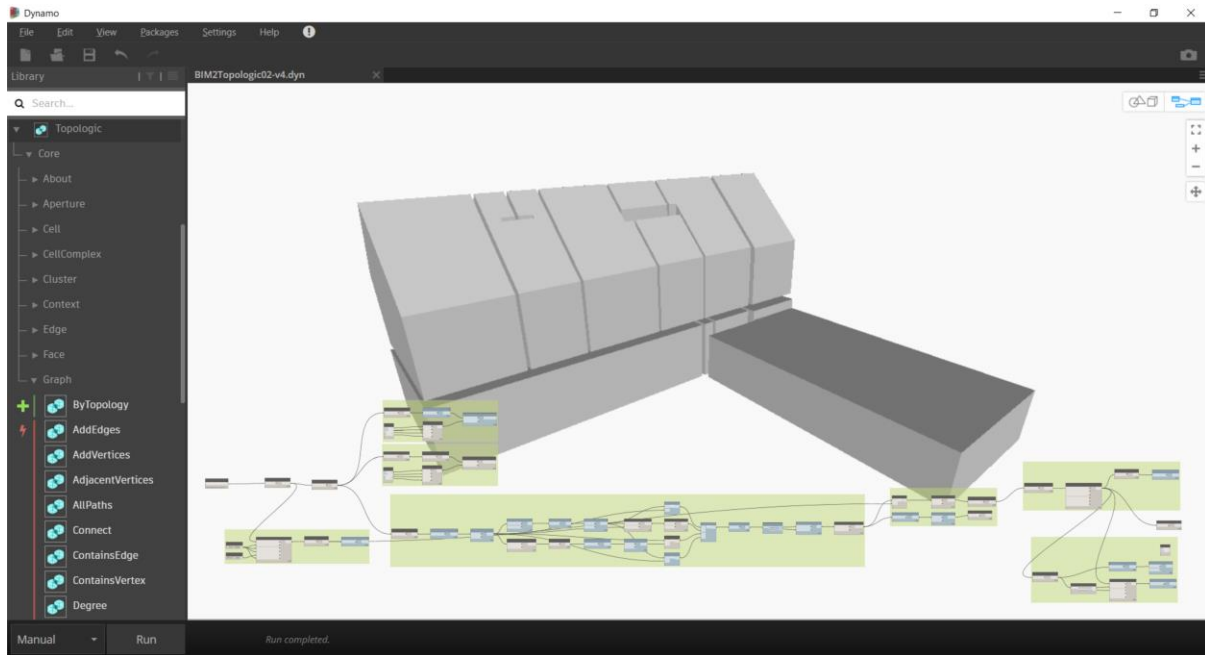


Figure 2. Topologic running in Dynamo

Literature Review

In order to design a spatial reasoning framework, a review on existing spatial data schemas and spatial reasoning frameworks is presented in this paper. Spatial data in this context refer to the located data related to every specific entity (Laurini and Thompson 1992). These data do not only refer to the location and orientation of the entity, but also non-location semantic data, such as the name and intended use of a room, as well as the number of occupants. Each of these data schemas and spatial reasoning frameworks may have seen considerable development, therefore it is beyond the scope of this paper to present a detailed literature review.

Spatial Data Models

This paper reviews four common data models and file formats from Building Information Modelling (BIM), Computer-Aided Design (CAD), and Geographical Information Systems (GIS), namely the Industry Foundation Classes (IFC) (buildingSMART International Ltd 2013; International Organization for Standardization 2018), gbXML (Green Building XML (gbXML) Schema 2019), X3D (Web3D Consortium 2019), and CityGML (Gröger and Plümer 2012). These data models are analysed for their support for three categories of data, namely the semantic attributes that can be attached to a spatial entity; the explicit topological relationships between two connected entities, either directly or indirectly; and the implicit topological relationships between two disjoint entities.

IFC

The IFC is an open data model which is specifically designed to facilitate interoperability and collaboration between the different teams in a BIM project. It was first conceptualised by a consortium which was initiated by Autodesk in 1994, and it is now maintained by buildingSMART. The IFC schema provides geometric, topological, and application-specific attributes. As part of being an integrated federated 3D model, IFC supports domain-specific attributes. These attributes are related to the building controls, plumbing and fire protection, structural elements, structural activities, heating, ventilating, and air-conditioning (HVAC), electrical, architectural, and construction management.

IFC supports explicit topological relationships between two entities of a boundary representation of an object under the `IfcTopologicalRepresentationItem` superclass, which immediately consists of

IfcConnectedFaceSet, IfcEdge, IfcFace, IfcFaceBound, IfcPath, IfcVertex, and IfcLoop. In IFC4, the implicit topological relationships can be categorised into 6, namely Assignment, Association, Connection, Declaration, Decomposition, and Definition. Out of these 6 categories, the Connection and Decomposition relationships are relevant for spatial reasoning.

CityGML

CityGML is the international standard for the 3D representation of an urban model, with support for geometric, topological, and semantic attributes. It has seen applications such as in disaster handling, emergency responses, pathfinding, as well as energy analysis. In terms of semantic attributes, CityGML provides pre-defined object types which are categorised into one of the thematic modules, namely core (whose attributes inherited by other object types), relief, building, bridge, tunnel, transportation, water body, vegetation, city furniture, land use, groups, and generic (used for further extensions).

The topology of the objects is described by the explicit, hierarchical connectivity between the geometric objects. While surfaces need to be planar and 2-manifold, CityGML offers some support for non-manifold topology in the form of Cell Complex. This feature enables modelling of a wall shared by two spaces. Two other yet related spatial relationships are the *relativeToTerrain* and *relativeToWater* attributes, which respectively describe the position of an object relative to the ground surface or the water surface. The *relativeToTerrain* attribute allows five values, namely *entirelyAboveTerrain*, *substantiallyAboveTerrain*, *substantiallyAboveAndBelowTerrain*, *substantiallyBelowTerrain*, and *entirelyBelowTerrain*. The *relativeToWater* attribute has similar values, replacing the keyword *Terrain with WaterSurface. In addition, it has the temporarilyAboveAndBelowWaterSurface to represent objects affected by water tides.

GBXML

GBXML is an XML schema that was designed to support data transfer in a BIM for further analysis. It has seen primary use for energy analysis, therefore the semantic attributes support this application, including variables related to heating, cooling, lighting, and space occupancy. GBXML also provides explicit 3D topological relationship between the geometric entities in the form of a Campus, Building, Space, Surface, PolyLoop, and CartesianPoint. However, there does not seem to be support for implicit relationship among the entities.

X3D

X3D is a standard to represent 3D computer graphics scenes and objects. This file format is a successor of the Virtual Reality Modeling Language (VRML), and provides classes and attributes to support a wide array of application domains, including scientific visualisation, CAD and architecture, GIS, animation, 3D printing and 3D scanning, as well as Augmented/Virtual/Mixed Reality applications. While the classes and attributes provided in the standard are mostly for geometric, topological, and visualization purposes, X3D provides Metadata classes to store custom attributes. Some further mechanisms have been proposed to extend its capabilities to support semantic data exchange. For example, X3D can be paired with the Resource Description Framework (RDF) to provide scene-independent ontology (Pittarello and De Faveri 2006).

Spatial Reasoning Frameworks

Numerous spatial reasoning frameworks have been proposed in the past and used in various applications. This section reviews spatial frameworks widely used in Architecture, as well as in other areas including computer games and Geographical Information Systems (GIS). Identified groups of frameworks include the Region Connection Calculus (RCC) (Randell, Cui, and Cohn 1992), Directional Reasoning Calculus (Frank 1992), space syntax (Hillier et al. 1976), dual graph (Boguslawski and Gold 2016), pathfinding (Toll et al. 2018), sound propagation (Raghuvanshi and Snyder 2018), and room lighting reasoning (Smith 2006).

Region Connection Calculus

Region Connection Calculus (RCC) provides a formal topological connection between two entities called *regions*. Its RCC-8 variant describes 8 fundamental relationships, namely Disconnected (DC), Externally Connected (EC), Equal (EQ), Partially Overlapping (PO), Tangential Proper Part (TPP), Tangential Proper Part Inverse (TPPi), Non-Tangential Proper Part (NTPP), and Non-Tangential Proper Part Inverse (NTPPi). These relationships are shown in Figure 3.

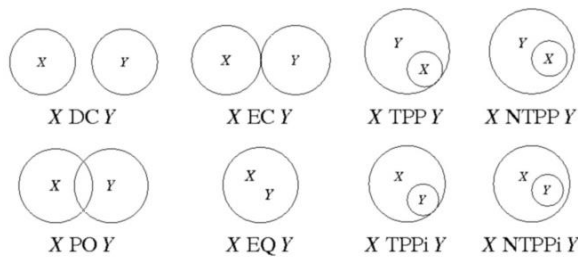


Figure 3. The 8 configurations in Region Connection Calculus

Directional Reasoning Calculus

Two models have been defined by Frank (1992) regarding directional reasoning in 2D space: the cone-based model and the projection-based model. The first one assumes that, given a point, a direction relation between two arbitrary points on a 2D space can be represented by one of nine directions: *East (E)*, *West (W)*, *South (S)*, *North (N)*, *North East (NE)*, *North West (NW)*, *South East (SE)*, *South West (SW)*, and *Identical (O)* for the direction of the other point (Nam, S and Kim, I. 2015). The projection-based model produces four directional partitions, e.g. north-west, north-east, south-east and southwest. An example of the two models is shown in Figure 4. Regarding 3D space and in the context of building information modeling, a point-based approximation, such as the centre of mass, is normally used. However, this rough approximation often results in unexpected results considering the user's intuition (Borrmann and Beetz 2010). A few directional reasoning calculi exist for the 3D space and some examples are the Rectangle Algebra (Balbiani et al. 1999) and the Cardinal Direction Calculus (CDC) (Goyal & Egenhofer 1997; Zhang et al. 2009).

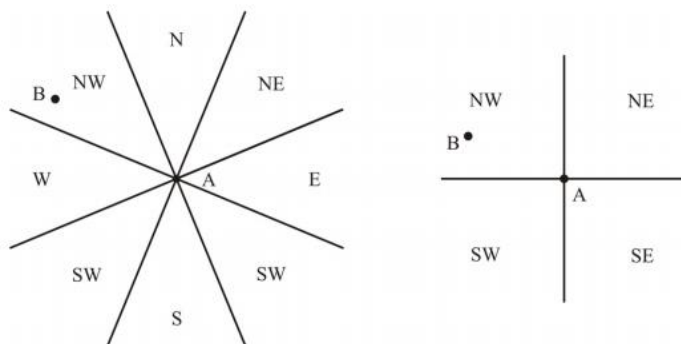


Figure 4. Frank's (1992) cone-shaped (left) and projection-based (right) models of directional relationships between points (Borrmann and Beetz 2010)

Space Syntax

Space syntax focusses on the analysis and decoding of the connection between spaces and how different types of connection shape human activity and social interactions. It is used as a theoretical but also a technological tool by architects to represent and analyse space, contributing to the exploration of design ideas for their proposals. Space syntax includes a set of analytical, quantitative and descriptive tools for analysing the spatial formations in different contexts: buildings, cities, interior spaces or landscapes

(Hillier and Hanson, 1984, Hillier, 1996). Metrics such as depth and integration are used to indicate proximity and accessibility.

Dual Graph

A *graph* is a data structure which is commonly used to explain the connectivity among several objects, and consists of vertices and edges. The dual graph is a type of graph which represents the dual structure of another topological entity. Typically, a cell and a face can be represented as a vertex at their centroids. The connection between adjacent cells or faces can then be represented as an edge connecting their centroids. A variant of dual graph is the Dual Half-Edge (DHE) (Boguslawski et al. 2016), which can describe the connection between the interior rooms of a building. It has seen applications in pathfinding and escape route computation.

Pathfinding

The primary objective of pathfinding is to find one or more paths between two locations in an environment by minimising a criterion, for example the distance or the number of vertices travelled. It is an established research area in numerous disciplines, including Architecture, Robotics, and Computer Games. The state-of-the-art pathfinding structures including the DHE (Boguslawski et al. 2016) and the navigation mesh (van Toll et al. 2018), which is akin to a mesh or, topologically, a shell consisting of convex polygons overlaid on the planar surfaces in a 3D environment. Pathfinding algorithms are typically based on graph search algorithms, such as the Dijkstra's algorithm (Dijkstra 1959).

Room Lighting Reasoning

Room lighting information can be used to determine the behaviour of a virtual agent. In stealth computer games, for example, an agent may prefer to move along the shaded areas to reduce her/his visibility and provide protection from the opponents (Smith 2006).

Sound propagation

The purpose of sound propagation simulation is to determine how acoustic waves travel in a virtual environment, thus providing realistic audio experience in Computer Graphics applications. A polygon soup (i.e. a set of unordered polygons) remains commonly used in the latest techniques (Raghuvanshi and Snyder 2018). Similar to the lighting information, sound propagation can also be used for reasoning purposes for an agent's decision making (Smith 2006).

A Non-Manifold Topology-Based Spatial Reasoning Framework

Based on the review presented in the previous section, one disadvantage that is commonly shared by existing representation is the use of 2D representations, either in the form of images or surfaces, although they can be positioned in a 3D environment. It can therefore be argued that there is a lack of representation of space in spatial reasoning mechanisms. Based on this finding, this research presents a unified NMT-based spatial reasoning framework. The benefit is mutual: not only does this research extend the capability of the NMT framework, but also the capabilities of these spatial reasoning mechanisms will be enhanced by the spatial coverage and hierarchical consistency that NMT offers.

Figure 5 shows an NMT-based spatial reasoning framework. On top of the NMT kernel, the Semantic Knowledge provide a mechanism to attach domain-independent semantic data to a topology, offering flexibility to multiple application domains. The spatial reasoning framework layer uses both the topological entities as well as the spatial knowledge to perform spatial analysis. Finally, the host applications (which, in the case of Topologic, are currently Dynamo and Grasshopper) provide a means for visualisation and interaction purposes.

At the moment, Topologic provides partial support towards this framework, in particular with its Dictionary, Graph (as the generalisation of Dual Graph), and Pathfinding features. These, along with

the implementation challenges will be thoroughly discussed in this section. More features will be implemented and tested as future works.

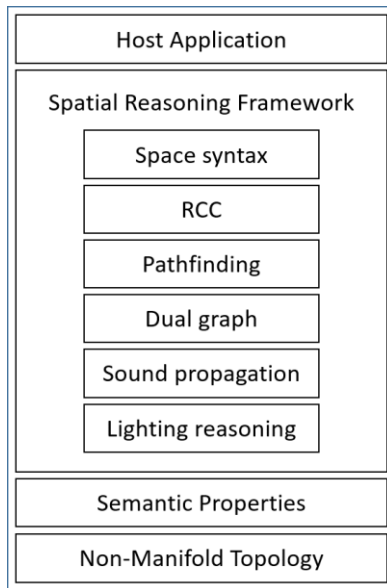


Figure 5. A spatial reasoning framework based on Non-Manifold Topology

Dictionary

The Dictionary system allows fully customisable semantic spatial attributes which are attached to the individual topological entities. It is application-independent, thus offering flexibility for utilisation in various applications. In this system, each topology is related to a hash map, which is a structure mapping unique keys to the values. Each of the key is a string, whereas currently the values can be an integer, a real number (a floating point), or a string. Support for other types of values, including a pointer to another entity, will be added in the future. A Dictionary can be attached to a topology using *Topology.SetDictionary* method, and queried from a topology using the *Topology.Dictionary* property. There is also partial support for Dictionary preservation across multiple operations, which keeps the Dictionaries of constituent topologies survive until the end of a workflow.

Graph

The Graph class in Topologic is implemented as an adjacency list, where in every vertex is mapped to a set of vertices it is connected to. This adjacency list is augmented with another list of edges, which describes the connecting topology between every pair of vertices. The Graph class contains a Dual Graph construction method from any type of topology, modifiers and queries related to the vertices and the edges, methods to perform pathfinding, and visualisation methods via conversion into the host geometries.

A dual graph can be generated from any kind of topology, taking into account the Aperture structures, such as windows and doors, in their presence. It should be noted that Topologic offers flexibility for an Aperture to be of any topological type, and can equally be attached to any kind of topology. The construction procedure is detailed in Table 1. It is also possible to get the topology of a Dual Graph as a Cluster, containing a set of edges and vertices.

Table 1. Dual graph construction rule

Type of input topology	Dual graph construction rule
Vertex	A graph consisting of the vertex. All existing apertures are connected between their centroids and the vertex.

Edge	A graph consisting of the edge. All existing apertures are connected between their centroids and the vertices of the edge.
Wire	A graph consisting of the wire. All existing apertures are connected between their centroids and the vertices of the wire.
Face	A graph connecting the face's centroid and the centroids of the face's bounding edges. All existing apertures are connected between their centroids and the face's centroid.
Shell	A graph connecting the centroid of each faces of the shell to the centroids of the face's bounding edges. The edges are distinguished between those shared by the faces of the shell (thus connecting the vertices at the centroids of two neighbouring faces) and those along the shell's external boundary, which are connected to only one face. All existing apertures are connected between their centroids and the face's centroid.
Cell	A graph connecting the cell's centroid and the centroids of the cell's bounding faces. All existing apertures on the bounding faces are connected between their centroids and the cell's centroid.
CellComplex	A graph connecting the centroid of each cells of the shell to the centroids of the cell's bounding faces. The faces are distinguished between those shared by the cells of the CellComplex (thus connecting the vertices at the centroids of two neighbouring cells) and those along the CellComplex's external boundary, which are connected to only one cell. All existing apertures on the faces are connected between their centroids and the bounded cell's centroid.
Cluster	A graph will be created for every constituent member according to its type, as written above. These graphs are then merged into the final graph.

For pathfinding purposes, Topologic provides three methods, namely *Graph.ShortestPath*, which is based on the Dijkstra's algorithm (Dijkstra 1959); *Graph.Path*, which is based on greedy search; and *Graph.AllPaths*, which exhaustively searches for all possible paths between a pair of vertices, with an optional time-limit. By default, these methods minimises the number of visited vertices. However, it is possible to minimise another criterion by specifying the vertex or edge key in the Dictionary.

Implementation Challenges

Two implementation challenges are identified, and they will be addressed as future work. The first one is related to the preservation of the dictionaries in the entire workflow. This is particularly challenging when the workflow involves Boolean operations, which may introduce new topological entities, and remove parts of the input entities. In these cases, there needs to be an intelligent rule to decide how the dictionaries survive, evolve, or are passed to the new entities. The second challenge is related to adapting the existing spatial framework into the mixed-dimensional NMT. A number of frameworks, for example the Space Syntax, were originally designed for a rasterised 2D floor plans. There have been some modified frameworks to deal with 3D structures, for example the 3D isovist algorithm (Suleiman, Joliveau, and Favier 2013) and mixed-dimensional RCC (Izadi, Stock, and Guesgen 2017). However, to the best of the authors' knowledge, the state-of-the-art pathfinding systems either use the topologically 1D dual graph (Boguslawski and Gold 2016), or the topologically 2D navigation mesh (Toll et al. 2018). A more in-depth review and investigation will be needed in these areas.

Study Case

Topological models have been used to perform pathfinding and navigation in the interior of a building, for example to assist the occupants when finding the way to a different part of the building, and for fire evacuation purposes (Jamali, Abdul Rahman, and Boguslawski 2016). The presented study case shows how to perform pathfinding on an NMT model. Three steps are involved in the workflow. Firstly, a BIM model is converted into a Topologic entity in Dynamo. Secondly, a dual graph is derived from the

Topologic entity. Thirdly, pathfinding is performed on the generated dual graph. This workflow runs on Revit 2020, Dynamo 2.3.0 (developmental build) and Topologic 0.8.5, using the Sample Architectural Project BIM model which is packaged with Autodesk Revit (Figure 6(a)).

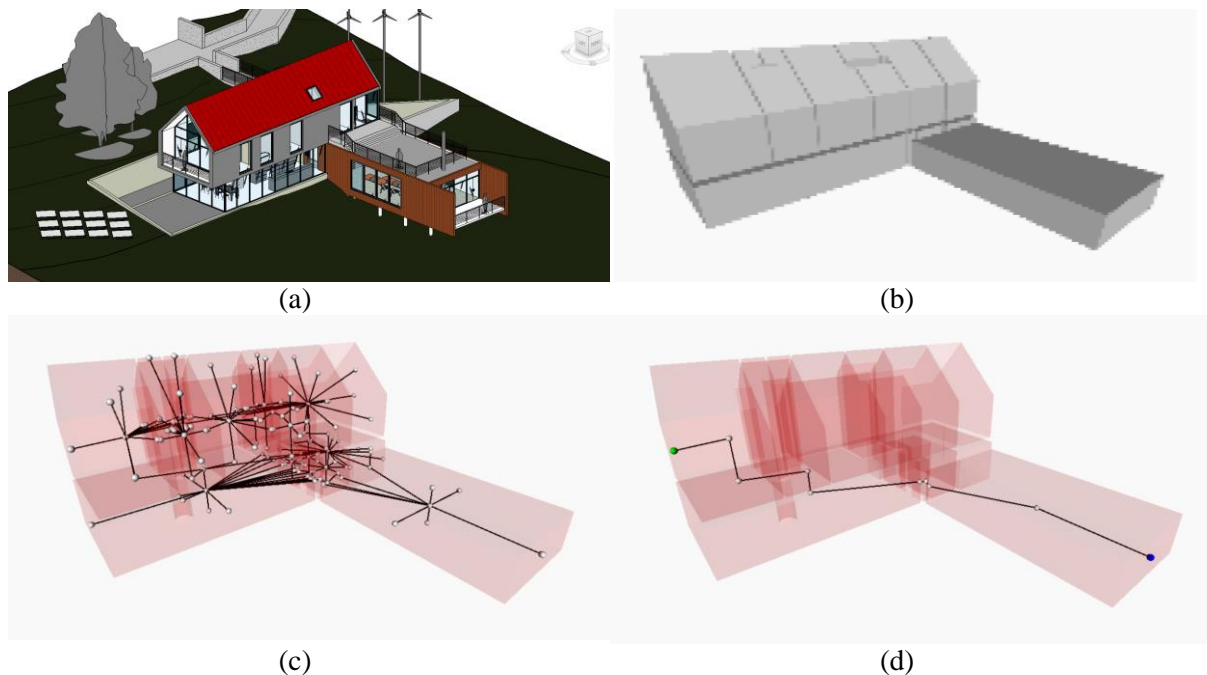


Figure 6. The pathfinding study case. (a) The original Sample Architectural Project BIM model in Revit. (b) The NMT cells, generated from the interior spaces of the BIM model, as shown in Dynamo. (c) The dual graph derived from the NMT model. (d) The shortest path between two vertices at different locations in the building.

Converting a BIM Model into a Non-Manifold Topology Model

In this step, the interior room spaces of the house are extracted from the BIM model as Dynamo solids. Topologic identifies these solids and converts them into topological cells. It should be noted that because the walls of the house are also modelled as solids, there are gaps with varying thickness separating these room spaces. Due to these gaps, it is not possible to group the cells into a CellComplex. Instead, they are grouped as a cluster, which is shown (as Dynamo solids) in Figure 6(b).

Deriving a Dual Graph from the NMT Model

This step begins by generating dual graphs from the individual cells. This results in small graphs in which the vertices at the centroids of the cells connected to vertices at the centroids of their respective bounding faces. While in an otherwise CellComplex-based model the graph would have just contained a single connected component, in this case, the smaller graphs are disjoint from each other due to the gaps between the cells. Therefore, a face matching procedure needs to be performed.

To match the shared faces of two adjacent cells, the following rule is used: two faces are considered neighbours if and only if:

- 1) They are parallel (in other words, the angle between their normal vectors should approximately be 0° or 180° within a small tolerance);
- 2) Their centroids must not coincide each other (so as to prevent zero-length edges), but under a maximum value (with this model, the value of 500 unit is used); and,
- 3) The perpendicular projection of one of the two faces onto the other one is entirely located within the other face.

Once every pair of neighbouring faces is identified, the vertices at their centroids are connected into edges, and merged with the previously generated small graphs inside the room cells to yield the final graph, as shown in Figure 6(c). Here, the spheres represent the vertices of the final graph, while the black lines correspond to the edges.

Pathfinding on the Dual Graph

Once the dual graph is generated, pathfinding can then be performed between any pair of rooms in the building. Figure 6(d) shows an example of finding a path between two extreme points, from the green sphere to the blue sphere, across the two storeys of the house, passing the white spheres on the bounding faces (walls). This path can then be used to guide a virtual agent which move between the green and blue spheres, for example in an evacuation scenario.

Conclusions and Future Work

The NMT structure possesses a potential for utilisation as the foundation of a spatial reasoning framework. The inherent support for multi-dimensional entities and adjacency queries will be particularly beneficial in this purpose. As part of the implementation of a spatial reasoning framework, the Topologic library provides a Dictionary system as well as the Graph class, which includes pathfinding measures. The identified challenges include retaining the Dictionary system in a workflow involving Boolean operations, as well as adapting existing spatial reasoning frameworks for a multi-dimensional entity.

Other forms of spatial reasoning framework will be added to Topologic. Further ahead, there is a potential for the creation of a Spatial Query Language (SQL) based on NMT, for example by extending from the SPARQL query language (Prud'hommeaux and Seaborne 2008), to simplify the formulation of spatial queries, as well as to enhance the spatial query capabilities of the framework. The concept of spatial reasoning can be used in other related areas. As an example, for digital fabrication, a rule-based system can be deployed on the conceptual zero-thickness skeleton of a building to intelligently place and trim solid building components, as well as to decide where to or not to deposit materials. Non-manifold cells with no adjacency may require a different way to handle compared to another cell which is heavily connected, and this can be conveniently determined in an NMT model.

Acknowledgements

This research project is funded by a Leverhulme Trust Research Project Grant (Grant No. RPG-2016-016).

References

- Aish, Robert et al. 2018. "Topologic: Tools to Explore Architectural Topology." In *AAG 2018: Advances in Architectural Geometry 2018*, Gothenburg, Sweden, 316–41.
- Aish, Robert, and Aparajit Pratap. 2013. "Spatial Information Modeling of Buildings Using Non-Manifold Topology with ASM and DesignScript." In *Advances in Architectural Geometry 2012*, Vienna: Springer Vienna, 25–36.
- Al-Jokhadar, Amer. 2018. "Towards a Socio-Spatial Parametric Grammar for Sustainable Tall Residential Buildings in Hot-Arid Regions Learning from the Vernacular Model of the Middle East and North Africa." Cardiff University.
- Balbani, P, Condotta, J-F and Farinas del Cerro, L 1999. 'A new tractable subclass of the rectangle algebra', Proc. of the 16th Int. Joint Conference on Artificial Intelligence (IJCAI-99).
- Boguslawski, Pawel, and Christopher Gold. 2016. "The Dual Half-Edge—A Topological Primal/Dual Data Structure and Construction Operators for Modelling and Manipulating Cell Complexes." *ISPRS International Journal of Geo-Information* 5(2): 19.

- Borrmann, A and Beetz, J 2010 'Towards spatial reasoning on building information models', Proc. of the 8th European Conference on Product and Process Modeling (ECPPM), Cork, Ireland, pp. 1-6
- Chatzivasileiadi, Aikaterini, Simon Lannon, et al. 2018. "Addressing Pathways to Energy Modelling through Non-Manifold Topology." In *SimAUD 2018*, Delft, Netherlands.
- Chatzivasileiadi, Aikaterini, Nicholas Wardhana, et al. 2018. "Characteristics of 3D Solid Modeling Software Libraries for Non-Manifold Modeling." *Computer-Aided Design and Applications* 16(3): 496–518.
- Ching, Francis D. K. 2014. *Architecture: Form, Space, & Order*. 4th ed. John Wiley & Sons.
- Curtis, W. J. R. 1996. *Modern Architecture Since 1900*. Prentice Hall.
- Dijkstra, E. W. 1959. "A Note on Two Problems in Connexion with Graphs." *Numerische Mathematik* 1(1): 269–71.
- Frank, Andrew U. 1992. "Qualitative Spatial Reasoning about Distances and Directions in Geographic Space." *Journal of Visual Languages & Computing* 3(4): 343–71.
- Goyal, RK and Egenhofer, MJ 1997 'The direction-relation matrix: A representation of direction relations for extended spatial objects', UCGIS Annual Assembly and Summer Retreat, Bar Harbor, ME
- Gröger, Gerhard, and Lutz Plümer. 2012. "CityGML – Interoperable Semantic 3D City Models." *ISPRS Journal of Photogrammetry and Remote Sensing* 71: 12–33.
- Hillier, B 1996, *Space is the Machine: A Configurational Theory of Architecture*, Cambridge University Press, Cambridge
- Hillier, B and Hanson, J 1984, *The Social Logic of Space*, Cambridge University Press, Cambridge
- Hillier, B, A Leaman, P Stansall, and M Bedford. 1976. "Space Syntax." *Environment and Planning B: Planning and Design* 3(2): 147–85.
- Izadi, Azadeh, Kristin M Stock, and Hans W Guesgen. 2017. "Multidimensional Region Connection Calculus." In *30th International Workshop on Qualitative Reasoning*, Melbourne, Australia.
- Jabi, W, Aish, R, Lannon, S, Chatzivasileiadi, A and Wardhana, NM 2018. "Topologic - A Toolkit for Spatial and Topological Modelling." In *Computing for a Better Tomorrow - Proceedings of the 36th ECAADe Conference - Volume 2*, ed. S Kepczynska-Walczak, A, Bialkowski. Lodz, Poland, 449–58.
- Jamali, A., A. Abdul Rahman, and P. Boguslawski. 2016. "3D Topological Indoor Building Modeling Integrated with Open Street Map." In *International Conference on Geomatic and Geospatial Technology (GGT) 2016*, Kuala Lumpur, Malaysia: International Society for Photogrammetry and Remote Sensing, 111–17.
- Laurini, Robert, and Derek Thompson. 1992. *Fundamentals of Spatial Information Systems*. Academic.
- Pittarello, Fabio, and Alessandro De Faveri. 2006. "Semantic Description of 3D Environments." In *Proceedings of the Eleventh International Conference on 3D Web Technology - Web3D '06*, New York, New York, USA: ACM Press, 85.
- Raghuvanshi, Nikunj, and John Snyder. 2018. "Parametric Directional Coding for Precomputed Sound Propagation." *ACM Transactions on Graphics* 37(4): 1–14.
- Randell, David A, Zhan Cui, and Anthony G Cohn. 1992. "A Spatial Logic Based on Regions and Connection." In *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR' 92)*, eds. Bernhard Nebel, Charles Rich, and William Swartout. Cambridge, MA: Morgan Kaufmann, 165–76.

Suleiman, Wassim, Thierry Joliveau, and Eric Favier. 2013. “A New Algorithm for 3D Isovists.” In *Advances in Spatial Data Handling*, Springer, Berlin, Heidelberg, 157–73.

van Toll, Wouter, Atlas F. Cook Iv, Marc J. Van Kreveld, and Roland Geraerts. 2018. “The Medial Axis of a Multi-Layered Environment and Its Application as a Navigation Mesh.” *ACM Transactions on Spatial Algorithms and Systems* 4(1): 1–34.

Turner, A 2001 'Depthmap: a program to perform visibility graph analysis', Proceedings 3rd International Symposium on Space Syntax, pp. 31.1-31.9

Zhang, X, Liu, W, Li, S and Ying, M 2008 'Reasoning with Cardinal Directions: An Efficient Algorithm', AAAI 2008, pp. 387-392

[1] <http://www.gbxml.org/>

[2] <https://www.iso.org/standard/63141.html>

[3] <https://www.iso.org/standard/70303.html>

[4] http://www.roningamedeveloper.com/Materials/RandySmith_GDC_2006.ppt

[5] <https://www.w3.org/TR/rdf-sparql-query/>

[6] <http://www.web3d.org/what-x3d-graphics>