TOPOLOGY MODELS AND RULES: A 3D SPATIAL DATABASE APPROACH

S. Salleh [1]*, U. Ujang [1], S. Azri [1]

[1] 3DGIS Research Lab, Department of Geoinformation, Faculty of Built Environment and Surveying, Universiti Teknologi Malaysia (UTM), 81310 Skudai, Johor, Malaysia – syahiirahsalleh78@gmail.com, (mduznir, suhaibah)@utm.my

KEY WORDS: 3D Topology, Topology Rules, Topology Model, Topological Relationships, 3D GIS, Spatial Database.

ABSTRACT:

Spatial databases are a key component of 3D applications that require the storage, management, and manipulation of 3D objects. Topological information that describes the containment, adjacency, and connectivity of objects within a 3D space is crucial for complex spatial analysis. The need to maintain optimal storage and computation within spatial databases recommends the use of lightweight approaches such as topology rules and models. However, current spatial databases use 2D topology mechanisms and limited 3D topology functions. This study conducted experiments on existing topology rules and models within ArcGIS and Oracle spatial database. Additional 3D topology rules were also implemented to determine topological relationships between 3D objects stored in Oracle. The 3D topology rules were based on a 36-Intersection Model (36IM) that describes intersections between objects from 0D to 3D. Based on the experiments, the 2D geodatabase topology rules within ArcGIS were able to determine topological relationships between the objects stored as 2D multi-surfaces datatype. The 2D topology rules and the topology model within Oracle were also unable to support topological relationships between 3D objects. The addition of 36IM topology rules could determine topological relationships and describe the dimensions of intersections. Evidently, the support for 3D topology within spatial databases depends on the availability of 3D datatypes. Most databases that support 3D datatypes have limited support for 3D topology, whereby existing topology mechanisms require the decomposition of 3D objects. The use of 3D topology rules supports the maintenance of topological information without breaking down 3D objects into lower dimension components.

1. INTRODUCTION

The use of 3D applications as a more accurate representation of the real world has necessitated spatial databases to be able to store and manage 3D data. In terms of geometries, 3D datatypes have been implemented to support solids, multi-surfaces and lines in 3D space. Topology which describes the relatedness of objects within a space is also important to provide containment, adjacency, and connectivity information. Topological relationships are a simple representation of topology that describes interactions between objects. Without topology to give meaning to it, 3D objects remain as a graphical output (Ujang et al., 2019). Therefore, topology is an accompaniment to object geometry (Barzegar et al., 2020). Topological information can be used in various 3D modelling, 3D analysis, and semantic feature extraction applications. A common problem faced in these applications is reconstructing topology or retrieving topological relationships.

Using topological relationships, various applications can benefit from enhanced capabilities. For example, in heritage house maintenance, the application of 3D city models with domain extension approaches enables better preservation and management (Mohd et al., 2017). Similarly, the development of a virtual 3D campus for a university, such as Universiti Teknologi Malaysia (UTM), utilises topological information to create an immersive and interactive environment (Salleh et al., 2021). These examples demonstrate the practical applications and value of incorporating topology into 3D applications.

In the context of spatial databases, topological information is maintained using a topology model or topology rules. A topology model is a schema that represents how the topological properties and relationships of objects are preserved (Ghawana and Zlatanova, 2012). These models broadly define the topological properties of an object without the actual physical structure of the definition (Ohori, 2015). On the other hand, topology rules are a set of rules that define valid topological interactions between objects or topological relationships. The implementation of topology rules provides topological connectivity information without the complexities of a topological data structure (Martinez-Llario et al., 2017). Both approaches are often implemented due to their lightweight nature and the fact that they require minimal storage.

This paper studied the use of topology models and topology rules for 3D objects within a spatial database. Parallel to the capabilities of spatial databases to handle 3D objects, topological representation of 3D objects should also be able to maintain adjacency, connectivity, and containment information. Both topology model and topology rules were tested on 3D objects stored within desktop GIS and RDBMS spatial database using existing and additional topology mechanisms. The following section presents related studies that implement additional support or extensions for topology within spatial databases. Next, Section 3 explains the methodology for this study, which includes experimentation of existing topology mechanisms and an additional implementation of 3D topology rules. The results and discussions are presented in Section 4. Finally, the conclusion is presented in Section 5.

2. RELATED STUDIES

Topology has long been utilised in applications and spatial analysis that requires connectivity information. Availability of 3D data and spatial database capabilities in handling 3D data have also made it possible to facilitate 3D applications. For instance, fast-paced development necessitates efficient

* Corresponding author

https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-117-2023 |

management of 3D building complexes that consists of multiple uses or ownerships. In addition to the 3D representation and storage of 3D objects or parcels, the complex relationships between objects can be described as legal spaces whereby each space adheres to rights, restrictions, and responsibilities.

Several studies have investigated and proposed topology models to comprehensively represent the 3D object with its 3D relationships. Barzegar et al. (2020) implemented a spatial query algorithm to identify horizontal and vertical boundaries between 3D parcels in a 3D BIM model based on 3D topological relationships. This includes legal boundaries of a 3D parcel which consists of interior, exterior and median boundary. Likewise, 3D volumetric parcels for cadastral applications undergoing subdivision processes can also use directional operators to determine topological relationships between faces. The intersection face which subdivides the 3D parcels is determined using directional operators that describe topological relationships of the object components (Jaljolie et al., 2021). An example of the identified boundaries for a 3D parcel based on the algorithm is shown in Figure 1.

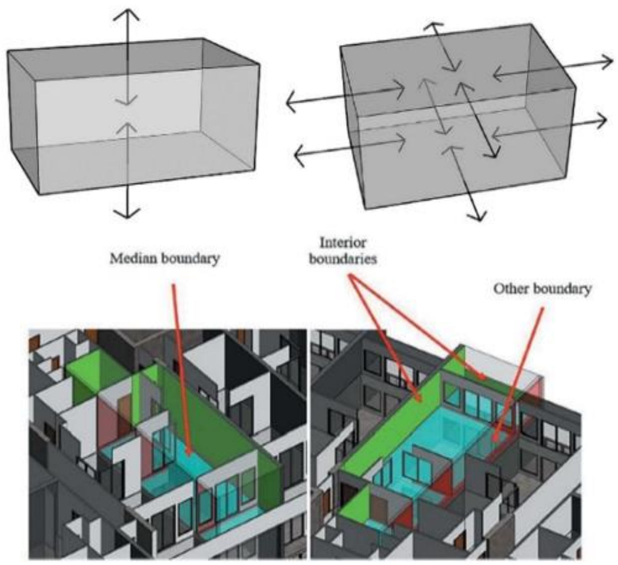


Figure 1. Identification of 3D parcel boundaries based on 3D topological relationships in horizontal and vertical direction (Barzegar et al., 2020)

Similarly, a topological model was developed in a study by Li et al. (2016) to determine topological relationships between 3D objects in CityGML which can be applied for objects in any level-of-detail (LoD). These models act as an extension of city modelling, BIM or LADM-based data models to provide topological support for 3D objects. On the other hand, topological data structures have also been developed and implemented to maintain topological properties for 3D objects stored within a spatial database. Ujang et al. (2019) developed a complex abstract cell complexes (CACC) topological data structure that can be used to preserve connectivity between 3D objects based on the topological primitives. An example of 3D object traversal using the CACC topological data structure is depicted in Figure 2.

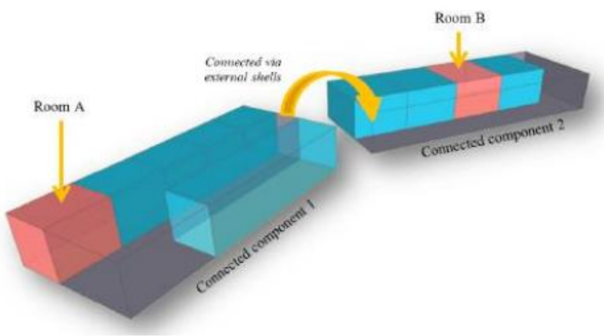


Figure 2. Navigation of 3D objects using the CACC topological data structure (Ujang et al., 2019).

Currently, 2D topology is fully supported in spatial databases, however, the support for 3D topology remains limited. Most spatial databases implement a set of 2D topology rules that are based on the Dimensionally Extended Nine Intersection Model (DE-9IM) which results in 2D topological relationships such as “meets”, “disjoint”, “overlaps”, “contains” and “within”. A more comprehensive set of 2D topology rules such as the 32 ArcGIS geodatabase topology rules describe valid interactions between 2D areas, 1D lines, and 0D points as shown in Figure 3.

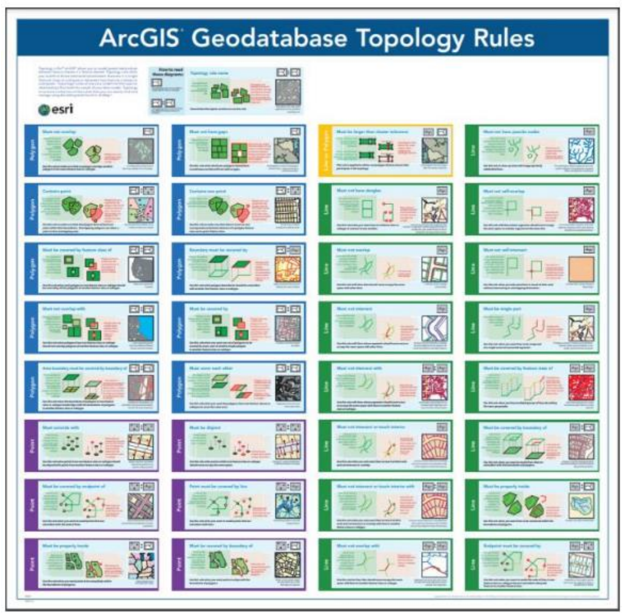
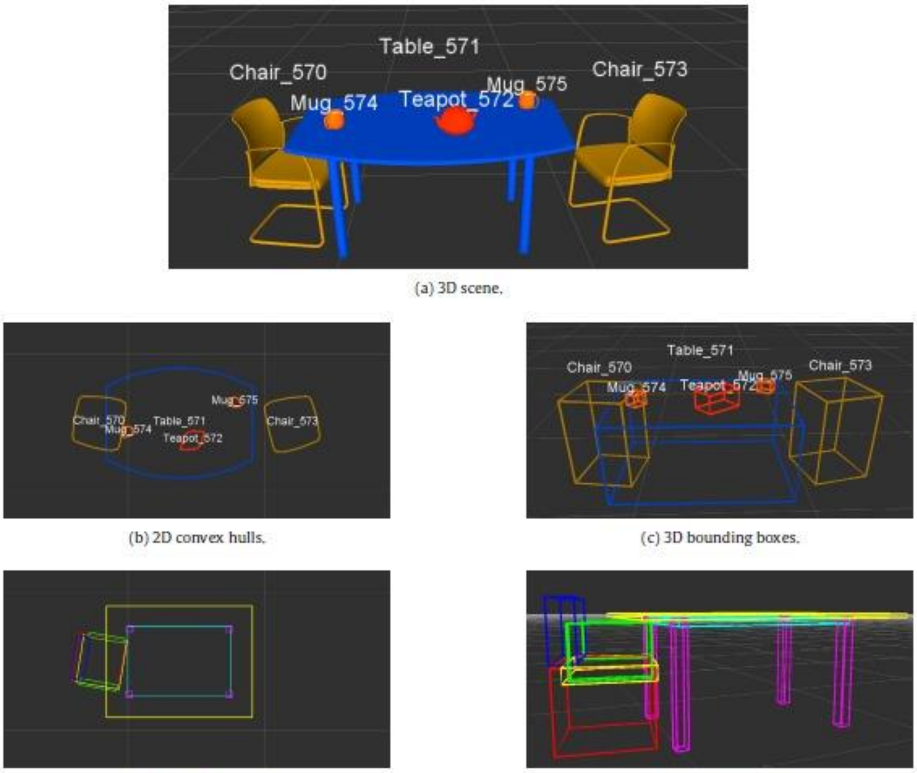


Figure 3. ArcGIS Geodatabase Topology Rules for 0D to 2D objects (ESRI, 2010).

In an effort to provide 3D topological support, extensive research has been conducted to develop custom extensions for handling 3D objects within spatial databases. Directional operators show direction vectors from one vertex to another. In terms of topology, directional operators can preserve positional information of objects in relation to each other (Deeken et al., 2018). Figure 4 depicts how this information was used as a semantic mapping framework (SEMAP) for PostgreSQL to support robotic applications where the positions of 3D objects in an indoor environment are detrimental to navigation (Deeken et al., 2018).



(a) 3D scene.

(b) 2D convex hulls.

(c) 3D bounding boxes.

(d) 2D decomposition.

(e) 3D decomposition.

Spatial Relation	Geometry	Target	Reference	Result
is-in	CH2D	Teapot572	Table571	True
is-in	BB3D	Teapot572	Table571	False
is-on	BB3D	Teapot572	Table571	True
intersects	$GEO3D_{body}$	Chair570	Table571	False
intersects	$BB3D_{body}$	Chair570	Table571	True
intersects	$BB3D_{links}$	Chair570	Table571	False

(f) Spatial queries.

Figure 4. Spatial relationships of objects within 3D scene (Deeken et al., 2018).

External topology engines were also developed as additional support for spatial databases. For example, the Jaspa topology engine was developed for the validation of datasets based on the DE-9IM topology rules (Martinez-Llario et al., 2017). Similarly, a Spatial Query Engine for Incomplete Information (I-SQE) was developed as a spatial reasoning approach to determine topological relationships in facilitating efficient spatial queries from incomplete spatial information (Cuzzocrea et al., 2011).

3. METHODOLOGY

Two types of spatial database were investigated consisting of a desktop GIS software and an RDBMS or Relational Database Management System. Based on previous studies (Salleh et al., 2021; Xu and Zlatanova, 2013), ArcGIS (ArcScene) and Oracle RDBMS had datatype capabilities to determine topological relationships between 3D objects. An experiment was carried out between two 3D objects that have a “meet (touches)” topological relationship using the topology functions and operators within the spatial database. The 3D objects with “meets (touches)” topological relationships used as test data is illustrated in Figure 5.

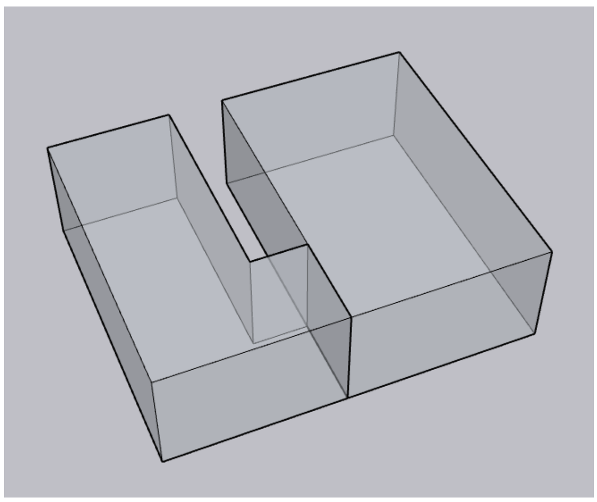


Figure 5. 3D objects that “meets (touches)”.

3.1 Existing Spatial Database Topology Experimentation

In order to determine topological relationships in ArcGIS, the “Select by Location” tool which consists of various spatial

https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-117-2023

operators based on the ArcGIS Geodatabase Topology Rules. Similarly, the 9IM was also utilised as topology rules within the Oracle RDBMS "SDO_Relate" utility. Besides that, a topology model can also be used in Oracle RDBMS which creates a topology table based on the geometries of objects stored in the database tables. The topology is built using topological primitives such as nodes, edges, and faces. The nodes are connected to form edges and in turn edges are connected to form faces.

3.2 36IM Topology Rules Experimentation

Another approach demonstrated is the implementation of 3D topology rules in the Oracle RDBMS as a separate PL/SQL script based on a 36-intersection model (36IM). Intersections between objects were tested from 0D intersections to 3D intersections whereby only the highest intersection dimension was preserved in the intersection matrix. The flowchart for the 36IM topology rules implementation is shown in Figure 6.

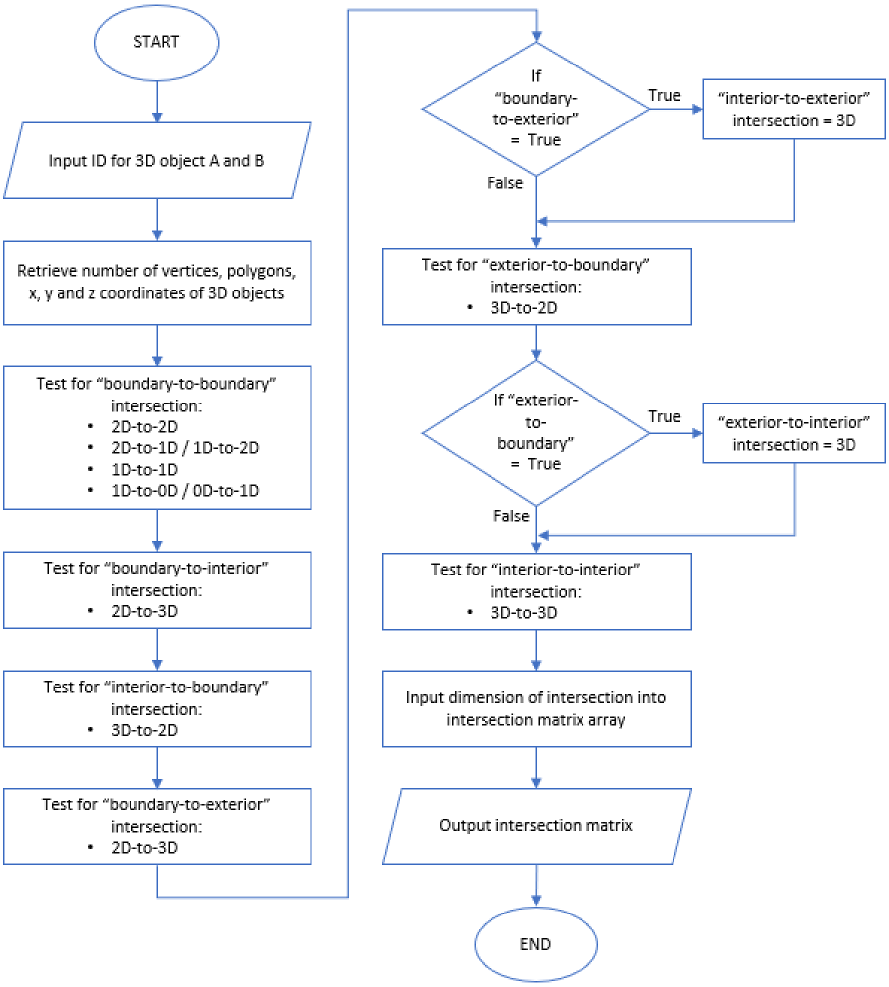


Figure 6. Flowchart of 36IM topology rules implementation.

The resulting intersection matrix describes nine intersections between 3D objects that are interior-to-interior, interior-to-boundary, interior-to-exterior, boundary-to-interior, boundary-to-boundary, boundary-to-exterior, exterior-to-interior, exterior-to-boundary, and exterior-to-exterior. The possible dimension of intersections for 3D objects based on the 36IM are described in Table 1.

Table 1. Possible dimension of intersections between 3D objects based on 36IM.

Object B / Object A	Interior B	Boundary B	Exterior B
Interior A	3D	2D	3D
Boundary A	2D	0D, 1D, 2D	2D
Exterior A	3D	2D	*(non-empty)

4. RESULTS AND DISCUSSION

The existing topological mechanisms and implementation of 3D topology rules were tested on two 3D objects with a "meets (touches)" topological relationship.

4.1 Existing Spatial Database Topology Results

The Geodatabase Topology Rules within ArcGIS were able to determine topological relationships between objects as shown in Figure 7.

https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-117-2023

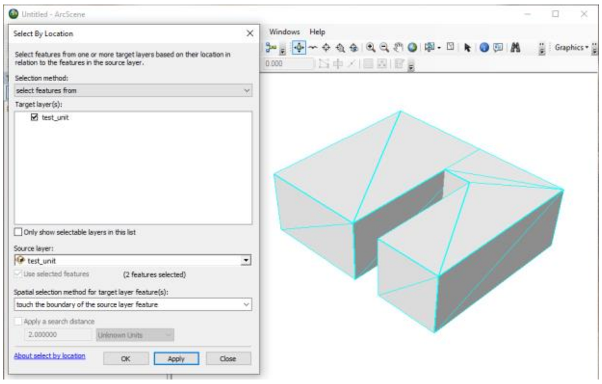


Figure 7. ArcGIS results using "touch" spatial operator.

Although it is a set of 2D topology rules, ArcGIS could determine topological relationships between objects as a result of storing 3D objects as a "MultiPatch" feature composed of 2D surfaces. An example of the decomposition from 3D object to "MultiPatch" feature is shown in Figure 8. Consequently, the topological information appears to be in 3D, but is limited to 2D topological connectivity. Nonetheless, it is still a practical method to determine topological relationships between 3D objects.

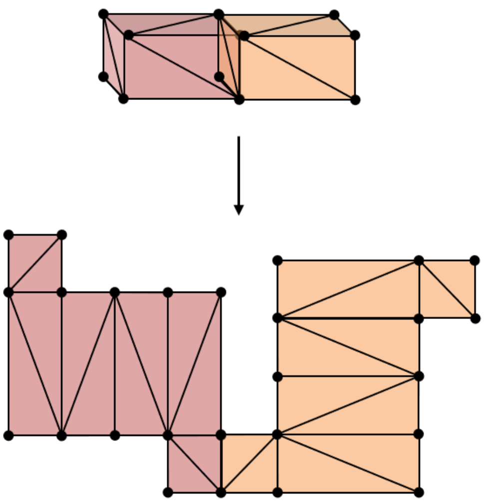
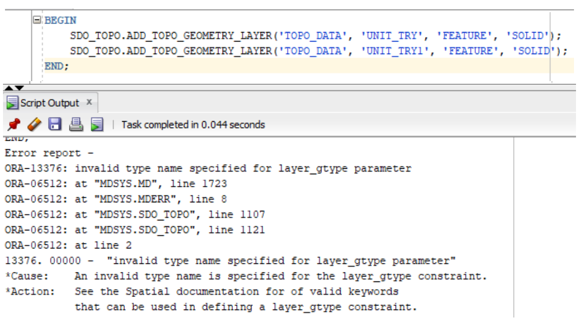


Figure 8. 3D object decomposition into "MultiPatch" feature.

The topology model within Oracle RDBMS was also unable to build a topology from 3D solid geometry objects, as shown in Figure 9.



```
BEGIN
    SDO_TOPO.ADD_TOPO_GEOMETRY_LAYER('TOPO_DATA', 'UNIT_TRY', 'FEATURE', 'SOLID');
    SDO_TOPO.ADD_TOPO_GEOMETRY_LAYER('TOPO_DATA', 'UNIT_TRY1', 'FEATURE', 'SOLID');
END;
```

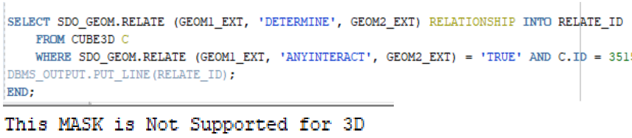
Script Output ×
Task completed in 0.044 seconds

```
Error report -
ORA-13376: invalid type name specified for layer_gtype parameter
ORA-06512: at "MDSYS.MD", line 1723
ORA-06512: at "MDSYS.MDERR", line 8
ORA-06512: at "MDSYS.SDO_TOPO", line 1107
ORA-06512: at "MDSYS.SDO_TOPO", line 1121
ORA-06512: at line 2
13376. 00000 -  "invalid type name specified for layer_gtype parameter"
*Cause:    An invalid type name is specified for the layer_gtype constraint.
*Action:   See the Spatial documentation for of valid keywords
           that can be used in defining a layer_gtype constraint.
```

Figure 9. Geometry type error in the build topology within Oracle.

This is due to the topology model being limited to edges that construct faces and the absence of a 3D solid topological primitive. Therefore, the 3D solid geometries could not be decomposed to populate the topology table with connected nodes, edges, and faces. Besides that, the objects tested had to be stored in different tables in order to successfully build the topology network. This hinders topology checks or relationships between objects within a table; for example, a table containing 3D sub-units of a high-rise building.

On the other hand, the 9IM topology rules within Oracle RDBMS was also found to be incapable of supporting topological relationships between 3D objects, as shown in Figure 10.



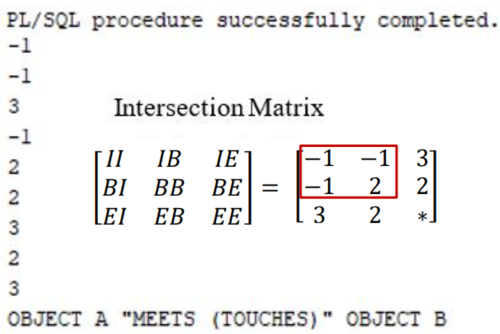
```
SELECT SDO_GEOM.RELATE (GEOM1_EXT, 'DETERMINE', GEOM2_EXT) RELATIONSHIP INTO RELATE_ID
    FROM CUBE3D C
    WHERE SDO_GEOM.RELATE (GEOM1_EXT, 'ANYINTERACT', GEOM2_EXT) = 'TRUE' AND C.ID = 351
DBMS_OUTPUT.PUT_LINE(RELATE_ID);
END;
```

This MASK is Not Supported for 3D

Figure 10. Oracle 9IM topology rule error in handling 3D objects.

4.2 36IM Topology Rules Results

The implementation of 3D topology rules in Oracle RDBMS was found to be able to determine the topological relationships between 3D objects, as shown in Figure 11.



$$\begin{bmatrix} II & IB & IE \\ BI & BB & BE \\ EI & EB & EE \end{bmatrix} = \begin{bmatrix} -1 & -1 & 3 \\ -1 & 2 & 2 \\ 3 & 2 & * \end{bmatrix}$$

Figure 11. 36IM topology rules implementation in Oracle results.

Based on the intersection matrix in Figure 11, the significant intersections for a "meets (touches)" topological relationship between 3D objects is that there are no intersections (-1) for interior-to-interior (II), interior-to-boundary (IB), and boundary-to-interior (BI). Apart from that, the intersection for boundary-to-boundary (BB) can be of value 0, 1, or 2. Therefore, the intersections between each dimension of geometries within the 3D object could be tested without decomposition of the 3D object into lower dimension geometries. This in turn preserves connectivity information between the 3D objects and yields accurate 3D topological relationship results.

5. CONCLUSION

Storage and query runtime are expensive in the usage of spatial databases in storing 3D data as well as executing spatial analysis. In terms of topological information, topological relationships are the simplest ways to describe the connectivity between objects. In this paper, existing topology rules and topology model within ArcGIS and Oracle spatial database were tested for topological relationships between 3D objects. The 3D objects are stored as multi-surface objects in ArcGIS which can visualise the objects as 3D but remain a 2D datatype. This allows the existing 2D topology rules to determine the topological relationships between the objects. However, the topological relationships remain in 2D, which does not maintain 3D intersections.

On the other hand, 3D objects stored in Oracle as a 3D datatype could not utilise existing 2D topology rules and topological models. An additional implementation of 3D topology rules was also tested within Oracle based on 36IM. The 3D topology rules could support 3D datatypes by determining 3D topological relationships that describe intersections from 0D to 3D without any object decomposition. In conclusion, the simplest and most effective way to maintain topological information is to use 3D topology rules whereby the topological relationships to be determined without requiring additional storage. The topological relationships can also be determined without any object decomposition, which safeguards the integrity of the 3D object, reduces query runtime, and accurately represents 3D connectivity between objects.

These findings highlight the dependence of 3D topology support on the availability of appropriate 3D datatypes within spatial databases. Most databases that support 3D datatypes currently have limited support for 3D topology, often requiring the decomposition of 3D objects and a new way of data acquisition (Azri et al., 2018; Azri et al., 2020; Hairuddin et al., 2019). The use of 3D topology rules presents a promising approach for maintaining topological information while avoiding the breakdown of 3D objects into lower-dimensional components.

ACKNOWLEDGEMENTS

This research was supported by Ministry of Education (MOE) through Fundamental Research Grant Scheme (FRGS/1/2021/WAB07/UTM/02/2).

REFERENCES

Azri, S., Ujang, U., Abdul Rahman, A., 2018. Dendrogram Clustering for 3D Data Analytics in Smart City. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* XLII-4/W9, 247-253.

Azri, S., Ujang, U., Abdul Rahman, A., 2020. Voronoi classified and clustered data constellation: A new 3D data structure for geomarketing strategies. *ISPRS Journal of Photogrammetry and Remote Sensing*. 162, 1-16.

Barzegar, M., Rajabifard, A., Kalantari, M., Atazadeh, B., 2020. 3D BIM-enabled spatial query for retrieving property boundaries: a case study in Victoria, Australia. *International Journal of Geographical Information Science*, 34(2), 251-271. doi:10.1080/13658816.2019.1658877.

ESRI, 2010. ArcGIS Geodatabase Topology Rules. https://desktop.arcgis.com/en/arcmap/latest/managedata/topologies/topology-basics.htm

Ghawana, T., Zlatanova, S., 2012. Increasing significance of 3D topology for modelling of urban structures. *Proceedings of Geospatial World Forum 2012*. 23-27 April. Amsterdam, The Netherlands.

Hairuddin, A., Azri, S., Ujang, U., Cuétara, M.G., Retortillo, G.M., Mohd Salleh, S., 2019. Development of 3D City Model Using Videogrammetry Technique. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* XLII-4/W16, 221-228.

Li, L., Luo, F., Zhu, H., Ying, S., Zhao, Z., 2016. A two-level topological model for 3D features in CityGML. *Computers, Environment and Urban Systems*, 59, 11-24. doi:https://doi.org/10.1016/j.compenvurbsys.2016.04.007

Martinez-Llario, J., Coll, E., Núñez-Andrés, M., Femenia-Ribera, C., 2017. Rule-based topology system for spatial databases to validate complex geographic datasets. *Computers & Geosciences*, 103, 122-132.

Mohd, Ujang, U., Choon, T.L., 2017. Heritage House Maintenance 3D City Model Application Domain Extension Approach. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*. 42.

Ohori, K.A., Ledoux, H., Stoter, J., 2015. An evaluation and classification of n-D topological data structures for the representation of objects in a higher-dimensional GIS. *International Journal of Geographical Information Science*, 29(5), 825-849.

Salleh, S., Ujang, U., Azri, S., 2021. 3D Topological Support in Spatial Databases: An Overview. *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLVI-4/W5-2021, pp. 473-478.

Salleh, S., Ujang, U., Azri, S., 2021. Virtual 3D Campus for Universiti Teknologi Malaysia (UTM). *ISPRS International Journal of Geo-Information*, 10(6), 356.

Ujang, U., Anton Castro, F., Azri, S. 2019. Abstract Topological Data Structure for 3D Spatial Objects. *ISPRS International Journal of Geoinformation*, 8(3), 102.

Xu, D., Zlatanova, S. 2013. An approach to develop 3D Geo-DBMS topological operators by re-using existing 2D operators. *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. II-2/W1, pp. 291-298.