QUANTIFYING GEOMETRIC CHANGES IN BIM-GIS CONVERSION

A. H. Liu*, C. Ellul

Dept. of Civil, Environmental & Geomatic Engineering, UCL, Gower Street, London, WC1E 6BT, UK
huaqiu.liu.18@ucl.ac.uk, c.ellul@ucl.ac.uk

Commission IV, WG IV/9

KEY WORDS: BIM, GIS, integration, conversion, data quality, geometry, information loss

ABSTRACT:

A conversion process is often carried out to migrate data during BIM and GIS integration, often from the highly detailed BIM to the less detailed GIS environment. Due to the differences between the two systems, information loss occurs during conversion. While research has been focusing on addressing information loss on the semantics, it is also necessary to quantify geometric changes resulted from converting geometry representations used in the two systems. This paper describes a preliminary study which evaluates the geometric changes during conversion for a list of primitives. The outcome shows that the metrics are useful both to those carrying out the conversion to balance between potential information loss and resulting data complexity, and to end users of the converted information to assess the fitness for purpose and impact of the conversion results.

1. INTRODUCTION

With the increase of availability of geospatial and BIM data, there has been a continuing recognition of the benefits for integrating geospatial information with building information models to bring together the built and natural environments for the Architecture, Engineering and Construction (AEC) industries, for a better understanding of the surrounding space where construction projects reside. As both BIM and GIS model the built environment, the integration of BIM and GIS, termed as GeoBIM, is widely acknowledged in both domains to be crucial and mutually beneficial for the realization of 3D city modelling (Ohori *et al.*, 2018). A number of applications across various domains have been identified which can benefit from the integration, including 3D cadastres, location-based services and navigation, asset management, site selection and planning, and construction coordination of infrastructure projects (Boyes *et al.*, 2017; Liu *et al.*, 2017, 2021; Noardo, Wu, *et al.*, 2020; Moretti *et al.*, 2021). More specific analyses include automatic check of the derived height of a planned building against the maximum height allowed in the development plan (Olsson *et al.*, 2018), parking availability, shadow analysis and other environmental impact of a proposed building in planning (Noardo, Ellul, *et al.*, 2020).

However, due to the difference in the original motivation of their development, GIS focuses primarily on modelling existing man-made or natural features from a building to cities and the world, while BIM focuses from a building down to its individual architectural, structural, and engineering components (Ohori *et al.*, 2018). As such, BIM data covers a limited geographical extent but contains much more engineering detail than GIS data (Ibid.). Other GIS and BIM differences identified include their respective focus on data management (data flows within spatial data infrastructure versus data functionalities in native software and file-based storage with collaboration tools), key players (government dominated versus industry dominated), data sharing, geometric representation and the use of local versus national spatial reference systems (Ellul *et al.*, 2020). These fundamental differences in the two systems introduce barriers and challenges to the integration process technically and non-technically. The absence of software to support both BIM and GIS data and GeoBIM capability was identified as the major technical challenge (Ellul *et al.*, 2018).

These issues mean that creating the integrated GeoBIM environment, with data in the same system, to support the above applications and analyses can be challenging. An extract, transform and load (ETL) process is often carried out during the integration[1] process to convert data from one system to another. In the case of GeoBIM, this process typically migrates data from the highly detailed BIM environment to the less detailed GIS environment.

Within the context of ETL to support integration, research into conversions between the two main interoperable standards for these data sources are common – IFC[2] (Industry Foundation Classes) for BIM and CityGML[3] (City Geography Markup Language) for GIS. To date, the majority of this research focuses on understanding and addressing the information loss on semantics during conversion (Floros *et al.*, 2018; Stouffs *et al.*, 2018; Biljecki *et al.*, 2021; Floros and Ellul, 2021).

However, there is also a necessity to quantify the changes in geometric characteristics. Different geometry representation methods (see Section 2.1) are often used in different file formats, data standards, software kernels, and applications across BIM and GIS. The initial geometric representation method of a model is determined by how the model is created and by its application context, and the model may come from sources including the digitalization from the real physical world, digital creation in

---

* Corresponding author

[1] There are different definitions and approaches in literature for BIM and GIS integration (see Beck *et al.* (2020)). In the context of this paper, integration is realized by conversion, i.e., the translation of the source model to the target model.

[2] https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/

[3] https://www.ogc.org/standards/citygml

https://doi.org/10.5194/isprs-annals-X-4-W2-2022-185-2022

modelling software, or procedural modelling that uses a set of rules describing the generation of a model (Ganovelli *et al.*, 2014). Specifically, the IFC standard represents geometry in various forms and CityGML uses BRep only (see Section 2.2).

Systematically understanding changes and information loss due to geometry conversion will provide a quality control measure for the ETL process, as unwanted changes could lead to incorrect calculations or poor performance when working with the data in the integrated environment. Understanding information loss is also fundamental in terms of increasing trust in the resulting data. For example, in the case of building planning and permits analytical rule checking, a control of the input data quality and the error propagation to the result is required, as any uncertainties in the converted geometry could result in a difference between a compliant and non-compliant building (Olsson *et al.*, 2018).

This paper thus aims to explore the possible means to quantify changes in geometric characteristics that occur during conversion due to different geometric representations and describes a preliminary study to understand their potential impact by testing a list of common primitive objects in different representations. Focusing on converting from implicit modelling approach (CSG and Swept Solids, used in BIM) to boundary representation (used in GIS), the paper will examine the geometric changes resulting from converting CSG to triangle meshes, which are the respective representative method of each category, by comparing changes in the volume, surface area and file sizes before and after conversion. The outcomes will be useful both to those carrying out the conversion – who will be able to optimize the process to find the best balance between potential information loss and resulting data complexity, and to end users of the converted information, who will be able to assess the fitness for purpose, and impact of the conversion on the results obtained through any analysis they carry out.

2. GEOMETRY REPRESENTATIONS

2.1 Common Type of Geometry Representations

The geometry of an object can be described in parametric form or implicit form. In the parametric form, points on the object surface are given directly by points in parameter domain mapped to the object space. In the implicit form, the points belong to an object are given indirectly through a point-membership classification function, which defines the relationship of the points to the space where the object is embedded. In both cases, for more complicated geometry, it may not be feasible to find a single function to represent the given shape within the accuracy tolerance, in which case the function domain is split into sub-regions with their individual functions defined, namely a piecewise definition. A common piecewise definition in the parametric form is the segmentation into triangles or quadrangles, and in the implicit form the embedding space is often split into voxels or tetrahedral cells.

As well as being classified as parametric or implicit, the representation methods used in solid modelling are classified by whether the model describes the surface of the object or the solid volume, namely, the boundary representation (BRep) and the volumetric representation, which benefit from the parametric form and the implicit form respectively. Examples of parametric surfaces used in BRep are the non-uniform rational B-splines (NURBS) surfaces and triangular surface meshes, and the advantage in geometric inside/outside queries using implicit form makes it suitable for construction solid geometry (CSG). Parametric and implicit representation have complementary advantages considering specific geometric operations in evaluation (e.g., sampling the surface geometry for rendering), query (e.g., determining point membership classification), or modification (e.g., editing the surface geometry or topology). Parametric surfaces are flexible in representing 3D object surfaces and can be converted to other representations easily. They are typically used in modelling software for manual creation of 3D freeform objects and can be difficult to create automatically (Ganovelli *et al.*, 2014). Implicit surfaces do not have holes given the defining function is continuous, and geometric self-intersections cannot occur (Botsch *et al.*, 2010). The selection of a suitable representation should be considered when considering a specific geometry operation and efficient conversion methods between the two are needed.

2.1.1 BRep: Boundary representation describes a solid using its bounding elements, including both the geometric description of the shape using its location and the geometric entities (points, lines and surfaces), as well as the topological description of the connectivity, orientation and adjacency of the bounding elements using its corresponding topological entities (vertices, edges and faces) (Hoffmann, 1989). BRep can be flexible in representing geometric shapes, however it does not guarantee a closed valid solid, and its topology needs to be validated before performing further operations on the model (Chang, 2014).

Polygonal surface mesh is one type of boundary representation that describes the smooth surface of a solid with a discrete approximation using planar polygons. A triangle mesh is widely used to represent the surface of an object, composed of a network of triangles with shared vertices and edges. Other than triangles, meshes can have quadrilaterals or other polygons as basic elements. Mesh representation has the advantage of being supported and optimized for processing algorithms, applications and graphic hardware as it is easy to convert other representations to polygon meshes for processing, and rendering with triangles is much easier and optimizable comparing to other complex shapes (Ganovelli *et al.*, 2014). However, as a discrete surface representation method, the mesh can only describe curved surfaces in approximation. The representation is not compact, in particular for highly detailed models, which require a large amount of data to capture all the details (Ibid.). Direct editing of a mesh can also be difficult, which also requires a data structure storing topological connectivity (Marschner and Shirley, 2018).

2.1.2 CSG: An object in constructive solid geometry is constructed from standard primitives using a sequence of regularized Boolean operations (Hoffmann, 1989). Typical primitives include rectangular blocks, spheres, cylinders, cones, and torus. Regularized Boolean operations are union, intersection, and difference. The primitives are first instantiated by applying dimension parameters to the generic shapes, followed by translation or rotation if necessary, then combined using the regularized Boolean operations, which ensure the closure of the resulting object interior and eliminate dangling lower dimensional parts, i.e., the planes, lines or points created from intersecting two touching primitives. The type of primitives can be extended to include more general shapes, provided they support CSG operations, for example swept solid shapes resulted from extrusion or revolution. A CSG model is expressed as a binary tree, whose leaves are the primitives and interior nodes are the Boolean operations or rigid body transformations, i.e., translations or rotations.

CSG is guaranteed to define a valid and bounded sets, provided the primitives are valid, and can be easily parameterized

(Mäntylä, 1988). It consists of primitives that are easy to handle and the CSG tree models a solid unambiguously. However, the type of primitives can be limited, and a solid can be represented in more than one way, which means the CSG tree may not be unique. A CSG tree is compact for storage, as it describes the sequence of the operations, not storing the intermediate resulting geometries. Although a CSG tree is concise, it can include redundant primitives that do not contribute to the final solid, and tends to grow when additional information to achieve efficient graphical operations is added to the basic tree (Mäntylä, 1988; Hoffmann, 1989). As CSG do not explicitly carry boundary information, some algorithms, e.g., boundary evaluation to construct the faces of the solid from a CSG tree, can be computationally complicated, and editing of a complex shape can be difficult (Chang, 2014; Ganovelli *et al.*, 2014).

2.2 Geometry Representation in Common File Formats

IFC is an open and vendor-neutral data exchange format developed by buildingSMART to allow sharing of relevant information throughout the lifecycle of any built environment asset among all participants independent from the software and tools used (buildingSMART International, 2021). Any object within an IFC building project is described as a semantic entity, linking to one or more distinct geometric representations, which then allows different geometric representations for different applications, e.g., simple triangulated meshes for model visualization and high quality BRep or CSG models for editing in BIM tools. However, the IFC data model does not address the potential consistency problems between the distinct geometric representations (Borrmann *et al.*, 2018). IFC supports a list of geometric representation methods, for example, constructive solid geometry (CSG), half-space solids, extrusion bodies and boundary representation (BRep). Starting from version IFC4, NURBS surfaces are also supported for describing surfaces in BRep. Interpretating the geometric information embedded in IFC correctly is an essential but complex process for a software tool, as it needs to support all representation methods defined by IFC (Amann et al., 2018).

To model objects in BRep, *IfcFacetedBrep* is used for flat surfaces only, while *IfcAdvancedBrep* can describe surfaces with curved edges, e.g., NURBS surfaces. If the object has cavities and holes, classes *IfcAdvancedBrepWithVoids* and *IfcFacetedBrepWithVoids* are used instead. To model objects in CSG, *IfcCsgPrimitive3D* provides primitives including blocks, spheres, cylinders, cones, and rectangular pyramids. *IfcBooleanResult* is the resulting model from the Boolean operations, which possesses an attribute for union, intersection, or difference operator and two operands that can be *IfcCsgPrimitive3D*, *IfcSolidModel*, *IfcHalfSpaceSolid*, or recursively, *IfcBooleanResult*. *IfcSweptAreaSolid* and *IfcSweptDiskSolid* model a 3D solid that is a rotation or extrusion of a 2D profile or a circular disc.

CityGML is an open data model developed by Open Geospatial Consortium (OGC) for storing and exchanging 3D city models, which models the geometry, semantics, topology, and appearance of objects within virtual 3D city and landscape models (Open Geospatial Consortium, 2021). Thematically it is structured into eleven modules to model different type of features, for example, buildings, bridges, tunnels, water body. Feature geometries are represented using the geometry classes defined in ISO 19107 Geographic information – Spatial schema. CityGML supports primitive geometries including points, curves, surfaces, and solids as well as aggregation (*MultiPoint*, *MultiCurve*, *MultiSurface*, *MultiSolid*) and composites (*CompositeCurve*, *CompositeSurface*, *CompositeSolid*) of the primitive types. CityGML 2.0 allowed a subset of ISO 19107 geometry types, restricting curves to straight lines and surfaces to planar polygons. CityGML 3.0 does not restrict the usage of any specific geometry type as defined in ISO 19107, allowing 3D surfaces to be represented as polygonal meshes or NURBS surfaces, unless a type is explicitly disallowed in the encoding. However, the volumetric solid can only be represented as BRep.

Esri's native file formats shapefile and geodatabase store 3D objects as multipatch geometry, which is a non-topological BRep data structure developed by Esri (ESRI, 2008). A multipatch is a collection of surface patches that consist of a combination of one or more of its primitive geometries, including triangles, triangle fans, triangle strips and rings (an area bounded by one closed path). To be a valid representation of a solid, a multipatch feature needs to be closed and orientable as other types of BRep.

Some other formats are also used for 3D models mainly for visualization purposes. The STL (Standard Triangle Language) format stores triangle meshes as separate triangles by their vertices with a unit normal for each face. It does not store any scale, colour, texture information. The structure has lots of redundancy but no connectivity information, making it prone to geometry errors, such as gaps or overlaps between faces (Marschner and Shirley, 2018). The Wavefront's OBJ format and the Object File Format (OFF) store a triangle mesh using triangle-to-vertex references, which requires approximately half of the storage of the previous structure and has further advantage in storage when attributes are stored with the vertices (Ibid.).

2.3 Representation Conversion and Geometric Challenges

Each representation method possesses its own advantages and disadvantages depending on the application in which they are being used. BIM software uses both the boundary representation (BRep) or the procedural modelling approach (e.g., CSG) to model the 3D geometry, often as a hybrid approach with the system documenting individual modelling steps of the construction while taking snapshots of the resulting explicit geometry to reduce computational load and improve display speed (Borrmann and Berkhahn, 2018).

One of the common conversions for BIM and GIS integration concerns with converting IFC to CityGML. As IFC supports more representation methods for solids, e.g., CSG, BRep and swept solids, the representation conversion from IFC to CityGML is often converting from implicit methods to BRep, and discretizing curves and curved surfaces into linearized lines and polygonal meshes if required in the target representation. BRep are easier to interpret as all geometric information is stored explicitly within the data model, e.g., coordinates of all vertices are available without the need of any evaluation. Implicit modelling methods, on the other hand, require the further evaluation on some geometric operations, such as the Boolean operations for constructing a solid in CSG. The evaluation can be complex for IFC as the operands can provide arbitrary complex solids to the Boolean operations, e.g., a triangulated surface body, instead of just geometric primitives.

Shapefiles are also used as a destination format to store 3D data converted from IFC, as it is a widely supported format among GIS software. Zhu, Wang, Wang, *et al.* (2019) implemented an algorithm to convert IFC swept solid into multipatch. The converted geometry was checked for validity, i.e., being topologically correct and closed. The algorithm was enhanced to accommodate more representation types from IFC, and a translation from IFC BRep to multipatch BRep was also needed, as the definitions of a closed ring are different where multipatch requires a closed ring repeating the first point at the end of the sequence (Zhu, Wang, Chen, *et al.*, 2019). The assessment of the output quality was based on the quantity of converted objects to indicate no geometric loss, however the potential geometric changes in individual objects were not assessed.

Ohori *et al.* (2017) converted IFC objects in implicit representation to polyhedrons in BRep that were supported by CGAL[4] kernel, and identified a few geometric errors emerged from conversion, including self-intersection errors existing in IFC geometries. An observation was made by the authors that different linearized lines or polygonal meshes would be generated from the implicit representations if the discretization method and its parameters were chosen differently. This supports the necessity of assessing the geometric changes occurred during conversion as the process can yield different results.

Another common geometric challenge results from the flexibility of IFC models allowing many ways to model an object. For example, the different ways of modelling and connecting walls make it difficult to implement a general method to identify external walls automatically (Olsson *et al.*, 2018). It is also a challenge for a software to support all types of representations, as observed by Ohori *et al.* (2017) the CGAL kernel could not support modelling all types of complex IFC features, which is the challenge for many other geometric kernels.

As BIM data often contains more detail than GIS (Ohori *et al.*, 2018), the complex geometry from the conversion can cause issue in the GIS system. In the process of integrating BIM and GIS for condition assessment for asset management, it was noted that geometry from BIM needed to be simplified before being visualized in GIS platform (Moretti *et al.*, 2021).

3. SOFTWARE, DATA AND METRICS

Given the independent development of BIM and GIS (Ohori *et al.*, 2018), common commercial software, such as Autodesk Revit (for BIM) or Esri ArcGIS, does not allow in depth comparison of converted data in both B-Rep and CSG. Thus, it is necessary to identify appropriate (combination of) software that will manage both representations. Although CSG is a main modelling method used in CAD and BIM software, other than IFC files, the common file formats do not explicitly store the model as CSG tree structure. Proprietary geometric kernels used in modelling software, for example Autodesk Revit, do not provide access to geometry representation information of the objects stored in their native format, therefore those objects created in implicit modelling method may not be retrieved as implicit representation. As popular open-source geometric kernels, such as Open CASCADE[5] or CGAL, are developed for BRep, this paper uses BRL-CAD[6] which is primarily based on CSG with basic support of BRep, to be able to create and interrogate objects in CSG as a starting point. Additionally, as noted in Section 2.3, previous conversion processes have highlighted potential issues resulted from discretization methods, i.e., converting from implicit representation to polygonal meshes, this paper focuses on testing primitives with curved surfaces, e.g., spheres, cylinders, cones, to understand the geometric changes before and after conversion. In addition to the metrics of vertex, edge and face count used by Wong and Ellul (2016) to validate converted BReps for 3D city models, volume, surface area and file sizes are also compared between the two representations.

3.1 Software: BRL-CAD as CSG Kernel

BRL-CAD is an open-source modelling system developed by U.S. Army Research Laboratory for military and industrial applications, which comprises of libraries, utilities, and tools to support interactive geometry editing, ray tracing and geometric processing for CSG models. The libraries are mainly developed with C/C++, allowing customization from the existing functionalities. Command line tools are also available handling common file format and geometry representation conversions.

The native file format for storing CSG model in BRL-CAD is a *.g* database which is organized as a directed acyclic graph to store a CSG tree with its primitive objects, Boolean operators and transformation matrices. The primitive objects supported in BRL-CAD include common types of prisms, spheres, cylinders, cones, torus, as well as special types such as elliptical hyperboloids and parabolic cylinders. When initiating the common shapes, spheres are created as special cases of ellipsoid and stored as ellipsoid type. Similarly, cylinders and cones are special cases of truncated general cones.

3.2 Data

The testing objects include five standard CSG primitive objects: one cube, one sphere, one cylinder, one cone and one torus. The primitives are instantiated by setting associated parameters as one meter, with reference to the size of the testing objects used in the GeoBIM Benchmark project (Noardo *et al.*, 2019). The parameters of the sphere, cylinder and cone are then scaled in both directions by a factor of two to create a series of the primitives varying in sizes, within the range of the object sizes commonly found in BIM models, e.g., from bolts, chairs, to windows, columns, and slabs.

The dimensions of the objects are as specified below:

A. One cube with sides of 1m
B. Nine spheres with radius of 0.0625m, 0.125m, 0.25m, 0.5m, 1m, 2m, 4m, 8m, 16m
C. Nine cylinders with radii of 0.0625m, 0.125m, 0.25m, 0.5m, 1m, 2m, 4m, 8m, 16m, at fixed height of 1m
D. Nine cylinders with heights of 0.0625m, 0.125m, 0.25m, 0.5m, 1m, 2m, 4m, 8m, 16m, at fixed radius of 1m
E. Nine cones with radii of 0.0625m, 0.125m, 0.25m, 0.5m, 1m, 2m, 4m, 8m, 16m, at fixed height of 1m
F. Nine cones with heights of 0.0625m, 0.125m, 0.25m, 0.5m, 1m, 2m, 4m, 8m, 16m, at fixed radius of 1m
G. One torus with major radius of 1m and minor radius of 0.5m

All of the primitive objects are created as CSG in BRL-CAD and stored as individual *.g* databases. Additionally, equivalent shape representations of cylinders and cones are created as swept solids, where CSG primitives are not directly supported to enable metric evaluation with additional software. Group C, D, E, F are created in Autodesk Revit as swept solids. Group C and D are created directly in IFC as swept solids (*IfcSweptAreaSolid*).

---

[4] https://www.cgal.org/

[5] https://dev.opencascade.org/

[6] https://brlcad.org/

https://doi.org/10.5194/isprs-annals-X-4-W2-2022-185-2022

3.3 Representation Conversion and Metrics Calculation

3.3.1 CSG in BRL-CAD: The BRL-CAD command line tools are used to convert CSG models to triangle meshes in OBJ and STL formats. The binary *.g* databases are also exported to ASCII format for file size comparison to the OBJ and STL files.

The volume of an object is compared between the theoretical value, the volume calculated for the CSG representation, and the volume calculated for OBJ/STL. The theoretical value is calculated from the shape dimensions. The volume in CSG is calculated by BRL-CAD using a quantitative geometry analysis function, which shoots grids of rays from the three axis-aligned directions and progressively refine the grids until the results from all three directions converge within a tolerance. The volume calculation for CSG is a discretized sampling method, whose accuracy depends on the spacing between the rays. A set of spacing distance is first tested on the sphere with radius of 1m, compared to the theoretical volume, to find the optimal spacing distance balancing the precision and processing speed. For objects with parameter set to 1m, the spacing distance for raytracing is 0.5mm. This value is scaled proportionally with the change of object sizes, unless the convergence cannot be achieved, for example for objects with a very large height-to-base ratio. The volume in OBJ/STL format is calculated using FME[7].

Similarly, the surface area of an object is compared between the calculated theoretical value and surface area measure from OBJ/STL using FME. Additionally, the number of vertices, edges and faces in the OBJ/STL meshes are calculated.

3.3.2 Swept Solids in Revit: The cylinders and cones created in Revit are exported into IFC in four levels of details, defined by Revit, as high, medium, low, and extra low. The volume and surface area of the swept solids are calculated from the shape dimensions, and the same metrics for the converted IFC shapes are calculated with FME along with the number of vertices, edges, and faces.

3.3.3 Swept Solids in IFC: The cylinders are converted to OBJ with FME in two modes, with generic polygonal faces or fully triangulated. The volume and surface area of the swept solids are calculated from the shape dimensions, and the same metrics of the OBJs are calculated with FME along with the number of vertices, edges, and faces.

Table 1 below summarizes the conversion and metrics calculation described above.

Primitives	Conversion	Software/Tools	Metrics
Sphere	N/A	BRL-CAD	Raytracing spacing
All	CSG to OBJ/STL	BRL-CAD, FME	Volume, surface area, file size, number of vertices, edges, and faces
Group C, D, E, F	Swept solid to triangle mesh	Revit, FME	Volume, surface area, number of vertices, edges, and faces
Group C, D	IFC Swept solid to OBJ	FME	Volume, surface area, number of vertices, edges, and faces

Table 1. Summary of primitives, conversion, and metrics

4. RESULTS

4.1 CSG in BRL-CAD

The volumes of all primitive objects were measured by raytracing using BRL-CAD and were compared to the theoretical value calculated from the shape dimensions. Of all fifty objects, the percentage variation between the measured and the theoretical volume ranges from -0.02% to 0.07%, with an average of 0.02%. Only the measured volume of the cube remained the same as its theoretical value. The larger variations occurred on cylinders and cones whose height-to-base ratio are highest or lowest.

For the conversion to OBJ/STL, the spheres were all triangulated into a mesh with 146 vertices, 432 edges and 288 faces. As expected, the larger the size of the sphere gets, the larger the differences in both the volume and surface area are. The percentage loss in volume and surface area was independent of the size of the spheres at 4.02% and 2.18% respectively. Although the same tolerance was set for triangulation, the cylinders/cones were not all triangulated the same way. The volume and surface area of the cube did not change after triangulation, while the percentage loss in volume and surface area of the torus were 10.05% and 4.00% respectively.

The average file size of the CSG exported in ASCII is 131 bytes. The sizes of OBJ and STL depend on the complexity of the meshes. For the same triangle mesh, STL takes on average six times more storage than OBJ. For a sphere stored as CSG, the ASCII is 109 bytes. Its corresponding triangle mesh with 146 vertices takes 8722 bytes as OBJ and 63601 bytes as STL.

4.2 Swept Solids in Revit and IFC

Revit allows exporting models in four levels of details. For cylinders and cones created as swept solids, the objects were exported as BRep and triangulated, by Revit's internal default setting, instead of being kept as implicit representations. The number of vertices and faces for different levels of detail are determined by Revit, and the numbers may or may not be consistent for the same primitives in different sizes (see **Table 2** and **Table 3** for the vertex and face count of the cylinders and cones). **Table 4** shows the percentage volume and surface area loss for cylinders in different sizes and levels of detail. Except for Extra Low, the high, medium, and low levels on average have a geometry loss in less than 1%. The changes in cones are in the same range and follow the same trend, although the meshes are much more complex. It has been observed from meshes both exported from Revit and converted with FME that the percentage changes of volumes of cylinders remain constant if triangulated the same way regardless of the size variations, while the percentage changes of surface area are size dependent.

5. DISCUSSION

This paper conducted a preliminary study on the geometric changes on conversion from CSG to triangle meshes, which are two representative representations of implicit and explicit modelling methods, by comparing changes in the volume, surface area and file sizes before and after the conversion on a list of primitive objects of various shapes and sizes.

Among the various sizes of spheres, the generated triangle meshes were tessellated the same way, i.e., with the same number

---

[7] https://www.safe.com/

of vertices and faces. However, it is worth noting that the percentage of change in volume and surface area remains constant, which results from the geometric characteristic of a sphere that it is a uniform shape in all directions. In the cases of cylinders and cones, while the volume change remains constant (when looking at meshes tessellated in the same way) among different sizes of shapes, the changes in the surface area are shape dependent. The similar the shape is to a sphere, the smaller changes there are in the surface area. In another way, the change in surface area is minimal when the height-to-ratio is one.

		62.5		**125**		**250**		**500**		**1000**		**2000**		**4000**		**8000**		**16000**	
Fixed height		**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**
	High	336	170	320	162	336	170	320	162	448	226	320	162	448	226	632	318	896	450
	Medium	224	114	320	162	336	170	320	162	448	226	320	162	448	226	632	318	896	450
	Low	84	58	80	42	112	58	160	82	224	114	320	162	448	226	632	318	896	450
	Extra Low	30	24	36	26	36	26	40	22	48	26	72	38	96	50	128	66	184	94
Fixed radius		**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**
	High	448	226	448	226	448	226	448	226	448	226	448	226	448	226	448	226	448	226
	Medium	448	226	448	226	448	226	448	226	448	226	448	226	448	226	448	226	448	226
	Low	224	114	224	114	224	114	224	114	224	114	224	114	224	114	224	114	224	114
	Extra Low	48	26	48	26	48	26	48	26	48	26	48	26	48	26	48	26	48	26

Table 2. Face and Vertex Count for Cylinders

		62.5		**125**		**250**		**500**		**1000**		**2000**		**4000**		**8000**		**16000**	
Fixed height		**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**
	High	9008	4560	4256	2179	2130	1079	1078	546	1330	676	945	488	1914	965	3414	1718	7354	3703
	Medium	4192	2141	4256	2179	2130	1079	1078	546	1330	676	945	488	1914	965	3414	1718	7354	3703
	Low	460	252	318	169	242	128	268	138	364	186	552	278	918	464	1870	944	3746	1887
	Extra Low	36	26	18	14	18	14	22	14	24	14	36	20	48	26	64	34	180	93
Fixed radius		**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**	**# F**	**# V**
	High	882	447	900	456	960	486	1024	519	1330	676	2211	1115	3986	2018	7926	4043	15700	8004
	Medium	882	447	900	456	960	486	1024	519	1330	676	2211	1115	3986	2018	7926	4043	15700	8004
	Low	220	112	224	114	224	114	260	132	364	186	476	242	938	479	1848	963	3604	1880
	Extra Low	24	14	24	14	24	14	24	14	24	14	24	14	44	25	84	50	202	120

Table 3. Face and Vertex Count for Cones

		62.5		**125**		**250**		**500**		**1000**		**2000**		**4000**		**8000**		**16000**	
Fixed height		**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**
	High	0.09%	0.03%	0.10%	0.03%	0.09%	0.04%	0.10%	0.05%	0.05%	0.03%	0.10%	0.08%	0.05%	0.04%	0.03%	0.02%	0.01%	0.01%
	Medium	0.21%	0.06%	0.10%	0.03%	0.09%	0.04%	0.10%	0.05%	0.05%	0.03%	0.10%	0.08%	0.05%	0.04%	0.03%	0.02%	0.01%	0.01%
	Low	3.32%	0.98%	1.64%	0.55%	0.84%	0.34%	0.41%	0.21%	0.21%	0.13%	0.10%	0.08%	0.05%	0.04%	0.03%	0.02%	0.01%	0.01%
	Extra Low	37.93%	12.15%	18.22%	6.27%	17.76%	7.26%	6.45%	3.24%	4.64%	2.90%	2.09%	1.57%	1.17%	1.00%	0.65%	0.59%	0.31%	0.30%
Fixed radius		**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**	**dV/V**	**dA/A**
	High	0.05%	0.05%	0.05%	0.05%	0.05%	0.04%	0.05%	0.04%	0.05%	0.03%	0.05%	0.03%	0.05%	0.02%	0.05%	0.02%	0.05%	0.02%
	Medium	0.05%	0.05%	0.05%	0.05%	0.05%	0.04%	0.05%	0.04%	0.05%	0.03%	0.05%	0.03%	0.05%	0.02%	0.05%	0.02%	0.05%	0.02%
	Low	0.21%	0.20%	0.21%	0.19%	0.21%	0.18%	0.21%	0.16%	0.21%	0.13%	0.21%	0.10%	0.21%	0.08%	0.21%	0.07%	0.21%	0.06%
	Extra Low	4.64%	4.43%	4.64%	4.25%	4.64%	3.94%	4.64%	3.48%	4.64%	2.90%	4.64%	2.33%	4.64%	1.87%	4.64%	1.56%	4.64%	1.38%

Table 4. Percentage Loss in Volume and Surface Area for Cylinders

From the file size comparison, CSG is a more compact format than BRep. Between the two formats storing triangle meshes, the size of STL, which stores individual vertices for each triangle, on average is six times larger than the size of OBJ, which stores vertices coordinates and indices of the vertices compromising the triangles. As expected, the more vertices and faces in a mesh, the larger the file gets.

The use of BRL-CAD gives the opportunity to create, access and interrogate objects in their CSG form. Due to the nature of implicit modelling approach, measuring volume and surface area of CSG objects are not straightforward. The option to measure surface area for CSG is still missing for the preliminary study, but BRL-CAD provides a way of measuring CSG volume by raytracing. The advantage of evaluating the common primitive shapes is that the theoretical value of the volume and surface area can be calculated from the shape dimensions, allowing a quality assessment on the accuracy of raytracing. From the results comparing the raytracing volume to the theoretical value, it shows the raytracing function in general offers reasonable estimates on primitive shapes, providing the spacing distance between the rays are carefully selected with the consideration of processing speed. This gives the opportunity for using the function to measure CSG volume for more complex shapes, when the theoretical values are difficult to calculate, although testing on more complex shapes still need to be conducted to verify the ability of raytracing handling irregular shapes.

From the results of the volume comparison of different sizes of cylinders and cones, the raytracing measurements are less accurate for shapes with very large or very small height-to-base ratio, e.g., the pointy cones or the nearly flat cylinders. It is also challenging to select an appropriate spacing distance for the rays in these cases, as the spacing is set uniformly in all three directions, which is difficult to balance the number of rays when the shape is highly disproportional. In the case of pointy cones, the rays coming from the top of the cone need small spacing distance to have a reasonable amount of hit/miss calculation, however, this would result in a great increase of numbers of rays in the other two directions.

In this preliminary study, using different software/tools for conversion shows a common challenge in controlling the triangulation algorithm, where the constraints can be set for distance/angle tolerance, but not the number of vertices/faces. This caused difficulties in comparing changes of volume and surface area for different shapes tessellated in the same way, where the impact on geometric changes by certain structures could have been better studied, e.g., triangle fans in cones, and triangle strips in cylinders.

An interesting observation is while using FME calculating the volume and surface area from the OBJ and STL files, although the coordinates are stored to the same precision in the files, there is a slight numerical difference between the calculations. The largest difference is less than 0.02 parts per million, which is assumed due to accumulated rounding errors.

This preliminary study quantified geometric changes, i.e., volume and surface area, resulted from converting primitive objects in CSG to triangle meshes. While in the conversion for BIM and GIS integration, objects are usually in more complex shapes and contain more details. Even though calculating volume and surface area for BRep is straightforward, it is not a trivial task for CSG or other implicit modelling methods. Many native formats in commercial software do not provide models in the form of CSG or other implicit approach, even though the implicit format may be a more compact option to transfer and store the original model. The geometric changes occur during conversion especially when approximating smooth surfaces with discretized ones, and the amount of change relates to the shape and size of the object. While Revit provides four levels of detail for exporting their models, there are no metrics to guide the user in the selection. From the test results, the percentage changes of volume and surface area are very close in the high, medium, and low levels, however the resulted number of vertices and faces vary significantly. Without a guiding metric, the user may have to choose unnecessarily higher level of detail, resulting in overcomplex geometry without gaining better accuracy.

As different geometric representations are used in different file formats, software and applications, and different methods have different advantages for specific geometric operations, conversion between the representations occur frequently, intentionally or in the backend. However, the represented objects are not always equivalent, e.g., polygonal surface meshes can only approximate a smooth surface. Some changes may be neglectable, for their insignificant sizes or the converted objects are intended for visualization purpose only, which does not require the same level of controlled accuracy as objects used for engineering analysis. However, as BIM and GIS are integrated for purposes beyond visualization, it is essential to be able to quantify the geometric changes occurred during conversion as part of the assessment of fitness for purpose. In the context of BIM and GIS integrated for the enabling work of a railway construction project (Liu *et al.*, 2021), to estimate the amount of material removal occurred during the piling process and to plan a safe disposal of the construction waste, the estimation is based on the converted geometry in the GIS system, which provides the site, ground and soil information. Based on the percentage loss calculated from cylinders in this preliminary study, for a total of 2127 piles planned in the project (HS2, 2021), the total volume loss, estimated with pile sizes of 525mm diameter and 25m depth (Skanska, 2009), is 11m^3 for mesh export in high and medium levels of detail, 96m^3 for low level and 2044m^3 for extra low level, where the resulted differences depend on the selected level of detail by the user, usually without any metric guidance.

6. CONCLUSION AND FUTURE WORK

This preliminary study was carried out to understand the geometric information loss occurred during representation conversions, by measuring volume, surface area and file size changes while converting from CSG to triangle meshes, which represent implicit modelling and BRep approaches respectively. By analyzing a list of primitive objects and objects varying in sizes, the preliminary results indicate that the geometric changes relate to the shape and size of an object. The use of open-source geometric kernel allowed the creation, access, and evaluation of CSG models directly and results show that in certain situations there can be significant information variation between the two formats depending on the tessellation algorithm, which is commonly not controlled by the user.

The conversion for BIM and GIS integration usually concerns with more complex shapes, where quantitative measure of the geometric changes is challenging, however an understanding of the geometric changes is necessary for quality control purpose. Future work will apply this method to examine more complex and irregular shapes, extending from the basic primitives to common objects used in BIM, evaluate with other geometric kernels, and to look for metrics that can quantify geometric changes locally, in addition to measure total volume and surface area, e.g., the shift of critical vertices, edges or faces.

ACKNOWLEDGEMENTS

This work was supported by the Engineering and Physical Sciences Research Council [grant numbers EP/N509577/1, EP/T517793/1].

REFERENCES

Amann, J., Preidel, C., Tauscher, E. and Borrmann, A. (2018) ‘BIM Programming’, in *Building Information Modeling: Technology Foundations and Industry Practice*. Springer.

Beck, F., Borrmann, A. and Kolbe, T.H. (2020) ‘The need for a differentiation between heterogeneous information integration approaches in the field of “BIM-GIS Integration”: a literature review’, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 6, pp. 21–28.

Biljecki, F. *et al.* (2021) ‘Extending CityGML for IFC-sourced 3D city models’, *Automation in Construction*, 121, p. 103440.

Borrmann, A., Beetz, J., Koch, C., Liebich, T. and Muhic, S. (2018) ‘Industry Foundation Classes: A Standardized Data Model for the Vendor-Neutral Exchange of Digital Building Models’, in *Building Information Modeling : Technology Foundations and Industry Practice*. Switzerland: Springer.

Borrmann, A. and Berkhahn, V. (2018) ‘Principles of Geometric Modeling’, in *Building Information Modeling : Technology Foundations and Industry Practice*. Switzerland: Springer.

Botsch, M., Kobbelt, L., Pauly, M., Alliez, P. and Lévy, B. (2010) ‘Surface Representations’, in *Polygon mesh processing*. CRC.

Boyes, G.A., Ellul, C. and Irwin, D. (2017) ‘Exploring BIM for operational integrated asset management-a preliminary study utilising real-world infrastructure data’, in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 49–56.

buildingSMART International (2021) *Industry Foundation Classes (IFC) - An Introduction*. Available at: https://technical.buildingsmart.org/standards/ifc/ (Accessed: 5 April 2021).

Chang, K. (2014) ‘Solid Modeling’, in *Product Design Modeling Using CAD/CAE*. Elsevier, pp. 125–167.

Ellul, C. *et al.* (2018) ‘Investigating the state of play of GeoBIM across Europe’, in *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, pp. 19–26.

Ellul, C., Noardo, F., Harrie, L. and Stoter, J. (2020) ‘The EuroSDR GeoBIM Project - Developing Case Studies for the Use of GeoBIM in Practice, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 44(4/W1), pp. 33–40.

ESRI (2008) *The Multipatch Geometry Type - An ESRI White Paper*.

Floros, G., Ellul, C.D. and Dimopoulou, E. (2018) ‘Investigating interoperability capabilities between IFC and CityGML LOD 4--retaining semantic information’, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences-ISPRS Archives*, 42(4/W10), pp. 33–40.

Floros, G.S. and Ellul, C. (2021) ‘Loss of information during design and construction for highways asset management: A geobim perspective’, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 8, pp. 83–90.

Ganovelli, F., Corsini, M., Pattanaik, S. and Di Benedetto, M. (2014) ‘How a 3D Model Is Represented’, in *Introduction to Computer Graphics*. 1st edition. Chapman and Hall/CRC.

Hoffmann, C.M. (1989) *Geometric and Solid Modeling, Geometric and Solid Modeling*. CA: Morgan Kaufmann.

HS2 (2021) *Euston Throat – Granby Terrace to Hampstead Road: Construction update*. Available at: https://s3-eu-west-2.amazonaws.com/commonplace-customer-assets/hs2ineuston/Euston Throat construction update_Commonplace.pdf (Accessed: 12 May 2022).

Liu, A.H., Ellul, C. and Swiderska, M. (2021) ‘Decision Making in the 4th Dimension—Exploring Use Cases and Technical Options for the Integration of 4D BIM and GIS during Construction’, *ISPRS International Journal of Geo-Information*, 10(4).

Liu, X. *et al.* (2017) ‘A state-of-the-art review on the integration of Building Information Modeling (BIM) and Geographic Information System (GIS)’, *ISPRS International Journal of Geo-Information*, 6(2), p. 53.

Mäntylä, M. (1988) ‘Constructive Models’, in *An Introduction to Solid Modeling*. Rockville, MD: Computer Science Press.

Marschner, S. and Shirley, P. (2018) ‘Data Structures for Graphics’, in *Fundamentals of Computer Graphics*. 4th edn. A K Peters/CRC Press.

Moretti, N., Ellul, C., Re Cecconi, F., Papapesios, N. and Dejaco, M.C. (2021) ‘GeoBIM for built environment condition assessment supporting asset management decision making’, *Automation in Construction*, 130, p. 103859.

Noardo, F. *et al.* (2019) ‘GeoBIM benchmark 2019: Design and initial results’, in *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, pp. 1339–1346.

Noardo, F., Wu, T., *et al.* (2020) ‘GeoBIM for digital building permit process: learning from a case study in Rotterdam’, *ISPRS Annals of Photogrammetry. Remote Sensing and Spatial Information Sciences, Copernicus GmbH*, pp. 151–158.

Noardo, F., Ellul, C., *et al.* (2020) ‘Opportunities and challenges for GeoBIM in Europe: developing a building permits use-case to raise awareness and examine technical interoperability challenges’, *Journal of Spatial Science*, 65(2), pp. 209–233.

Ohori, K.A. *et al.* (2017) ‘Towards an integration of GIS and BIM data: What are the geometric and topological issues’, *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4(W5), pp. 1–8.

Ohori, K.A., Biljecki, F., Kumar, K., Ledoux, H. and Stoter, J. (2018) ‘Modeling Cities and Landscapes in 3D with CityGML’, in *Building Information Modeling: Technology Foundations and Industry Practice*. Switzerland: Springer.

Olsson, P.-O., Axelsson, J., Hooper, M. and Harrie, L. (2018) ‘Automation of building permission by integration of BIM and geospatial data’, *ISPRS International Journal of Geo-Information*, 7(8), p. 307.

Open Geospatial Consortium (2021) *OGC City Geography Markup Language (CityGML) Part 1: Conceptual Model Standard*. Available at: https://docs.ogc.org/is/20-010/20-010.html (Accessed: 13 September 2021).

Skanska (2009) *Driven Piling*. Available at: https://www.skanska.co.uk/4afe5f/siteassets/expertise/construction/piling-and-foundations/case-studies/driven-piles-cementation-skanska-data-sheet.pdf (Accessed: 12 May 2022).

Stouffs, R., Tauscher, H. and Biljecki, F. (2018) ‘Achieving complete and near-lossless conversion from IFC to CityGML’, *ISPRS International Journal of Geo-Information*, 7(9), p. 355.

Wong, K. and Ellul, C. (2016) ‘Using geometry-based metrics as part of fitness-for-purpose evaluations of 3D city models’, in *ISPRS Annals*, pp. 129–136.

Zhu, J., Wang, X., Chen, M., Wu, P. and Kim, M.J. (2019) ‘Integration of BIM and GIS: IFC geometry transformation to shapefile using enhanced open-source approach’, *Automation in Construction*, 106.

Zhu, J., Wang, X., Wang, P., Wu, Z. and Kim, M.J. (2019) ‘Integration of BIM and GIS: Geometry from IFC to shapefile using open-source technology’, *Automation in Construction*, 102, pp. 105–119.