

Building Information Model Pre-Processing for Automated Geometric Quality Control

M. Bueno¹ and F. Bosché²

^{1,2}School of Engineering, University of Edinburgh, Scotland

martin.bueno@ed.ac.uk, f.bosche@ed.ac.uk

Abstract -

Geometric quality control (QC) in a construction project is an important but time-consuming and not value-adding task. While significant progress is being made in construction digitalisation, geometric QC processes remain highly manual and inefficient. This manuscript proposes a new methodology to pre-process initial information contained in the as-designed Building Information Model ('BIM model' hereafter) and obtain a full list of the geometric QC tasks that need to be conducted over the duration of the corresponding construction project. The proposed methodology employ a network graph constructed automatically from the BIM model information; a dictionary of types of geometric QC (e.g. dimensions to be checked with tolerances) that apply to the given project; and a QC digital manager that ties both elements together and identifies the list of unique geometric QC tasks that need to be conducted throughout the given project. The workings of the proposed methodology are illustrated with a case study, for some QC specifications extracted from the EN 13670 standard. These demonstrate its usefulness to exhaustively establish and record all geometric QC tasks required over a project.

Keywords -

BIM; IFC; Quality Control; Geometric; Point Clouds; Digital Twin

1 Introduction

During construction, guaranteeing that the built elements meet quality specifications (e.g. geometric tolerances) is critical to ensure they "achieve the intended level of safety and serviceability during their service life" [1]. From a construction delivery viewpoint, this is also critical to ensure that subsequent construction works are not delayed (due to rework), which would also result in additional costs.

Yet, current practice in geometric Quality Control (QC) is human-intensive, prone to error. Besides, quality specifications are recorded across a multitude of documents, including standards, regulations, and bespoke project-specific specifications, which makes it difficult to track all quality specifications and conduct all necessary corre-

sponding QC assessments and measurements.

Digitalisation in the construction industry promises numerous benefits in terms of efficiency and quality improvements [2]. But, despite the current challenges discussed above, Geometric QC processes have seen some, but limited evolution. Automated compliance checking is an active area of R&D for the analysis and validation of design (BIM) models [3, 4]. But this area has not explored compliance checking of actual works, only design (BIM) models. Regarding QC of physical construction, developments have occurred around surveying technologies, with laser scanning (terrestrial and mobile) and photogrammetry offering means to rapidly and effectively collect 3D survey data. Among those, terrestrial laser scanning (TLS) offers high accuracy that is particularly suitable for use in geometric QC, but the analysis of the large point clouds TLS acquires has traditionally remained a manual and tedious process. Digitalisation of this data analysis stage would provide significant benefits to improve the quality and speed of geometric QC activities. To fill this need, works have been exploring the 'scan-vs-BIM' principle to match laser scanned points to components in 3D BIM model in order to (1) recognise those components and subsequently (2) assess their geometric correction [5, 6, 7].

Although it remains an active area of research, the value of the 'scan-vs-BIM' principle to analyse point cloud is now generally accepted. But, the remaining gap in knowledge is in automatically identifying what type of geometric QC needs to be conducted where in the model, and robustly conducting all these geometric QC checks [8].

In this paper, we propose a methodology to automatically analyse 3D BIM models to identify where geometric specifications apply and thus need to be quality-controlled. The methodology includes the development of (1) a simplified dictionary to store the different geometric specifications, or rules; (2) an algorithm to extract from the design 3D BIM models all components and component relationships relevant to the given specifications, stored in a graph structure; (3) an algorithm to process that graph to systematically establish where each specification is applied in the given model. The output of that last step is a list of QC checks that need to be conducted during construction.

The rest of the manuscript is organised as follows. In

Section 2, the GeometricQC tool is briefly introduced and the proposed method for establishing geometric QC tasks given the design BIM model is detailed. Preliminary results obtained with a case study is reported in Section 3. Section 4 concludes the manuscript.

2 Proposed methodology

2.1 GeometricQC tool overview

The work presented in this manuscript is part of a highly automated geometric QC tool, called GeometricQC tool, developed as part of the COGITO project [9].

The GeometricQC tool is designed to conduct automatic geometric QC by comparing the as-designed geometry, contained in the BIM model, and the as-built data, captured in the form of point clouds acquired using TLS. The tool essentially follows an Automated Rule Checking (ARC) approach [10, 11]. The GeometricQC tool first defines what geometric QC needs to be conducted, where and when during the project geometric QC. After components requiring geometric QC are constructed, the surveyor acquires the relevant site-referenced (or geo-referenced) laser scans. At this point, the GeometricQC tool automatically matches the as-built TLS points to the 3D geometry of the components of interest stored in the as-designed BIM model, following a ‘scan-vs-BIM’ process, and performs the scheduled geometrical control. This GeometricQC tool workflow is summarised in Figure 1.

This paper focuses on the first part of the above process that occurs during the planning phase of a construction project. This part aims to define ‘what’ geometric QC needs to be conducted, ‘where’ and ‘when’ during the project geometric QC’.

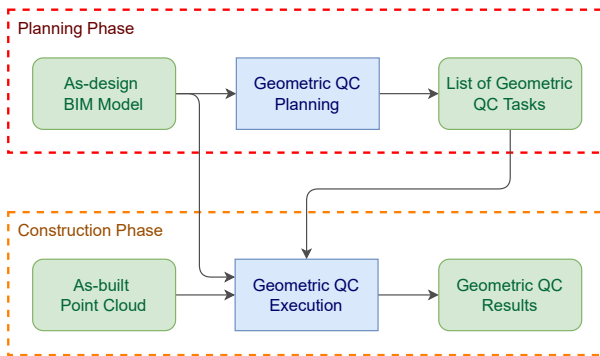


Figure 1. GeometricQC tool workflow. This paper focuses on the planning phase of the process.

For the ‘what’, digital rules first need to be defined. In construction, as discussed earlier, geometric QC specifications and tolerances can be defined in various ways, but standard specifications also exist. For example, EN

13670:2009 [1] and EN 1090-2:2018 [12] provide construction and erection geometric specifications for the execution of concrete and steel structures, respectively. These include all dimensions (or geometry) that need to be checked and tolerances for each of them. For example, EN 13670:2009 [1] details in section 10.4 and annex G the different geometrical tolerances that need to be verified for each of the structural components and the type of deviations. Figure 2 shows two representations of the specifications in EN 13670:2009 [1], for the inclination of a structural column or wall, and for the alignment of stacked structural columns or walls.

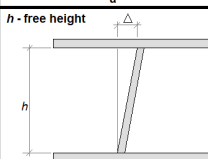
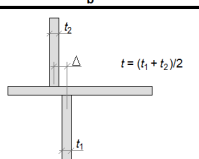
	a	b
Type of deviation		
Description	Inclination of a column/wall at any level in a single- or a multi-storey building	Deviation between centres
Permitted deviation Δ	The larger of $h \leq 10 \text{ m} \Rightarrow 15 \text{ mm or } h/400$ $h > 10 \text{ m} \Rightarrow 25 \text{ mm or } h/600$	The larger of $\#30 \text{ or } 15 \text{ mm}$ but not more than 30 mm

Figure 2. EN 13670 [1] Walls and Columns geometrical tolerances example. (a) Inclination/verticality of a single wall/column, (b) Deviation between centres of stacked walls/columns.

In this paper, we present a solution to digitise such geometric specifications as digital rules, thereby creating a dictionary of all geometric specifications (i.e. all rules) that apply to a given project. We focus on 15 rules defined in EN 13670:2009 [1] for the execution of concrete structures.

For the ‘where’, the solution must identify where the above rules apply in the given project. For this, we developed an algorithm that processes the as-design BIM model, in IFC format [13], and detects where the geometric QC rules contained in the pre-defined dictionary apply. The output of this process is a network graph where nodes are the components in the as-design BIM model requiring geometric QC and edges are the component relationships that are relevant to the geometric specifications to be applied to the project.

Finally, for the ‘when’, we take into account the information contained in the project 4D model that links the as-design BIM model components to construction schedule activities, so that the defined list of geometric QC tasks can be laid over time, in line with the construction schedule.

The above three steps are detailed in the following subsections.

2.2 What: Establishing the Geometric QC Dictionary

The geometric QC dictionary encompasses all the geometric QC ‘rules’ that need to be checked during the project. Each rule defines the context within which it must be applied. In the case of structures, that context is defined by: the types of components involved in the rule (e.g. wall, column, slab), their material types (e.g. concrete or steel), and their geometric relationships (e.g. stacked, connected). Note that some rules refer to single components (e.g. column inclination), in which case no geometric relationship applies to define the rule’s context. At each location in the as-design BIM model where the defined context is encountered, the corresponding rule must be checked. Since we consider the as-design BIM model to be encoded in IFC format, it is important that the dictionary rules be encoded by employing standard (or pre-agreed) IFC classes and properties, such as *IfcWall* or *IfcColumn* for component types, and material category contained in the *IfcMaterial* field of a component description.

The dictionary rules then contain additional description fields, capturing where the rule comes from (e.g. title of original document and section or specification number with it), and a description of the rule. Finally, the rule is associated to software code that encodes the actual geometric specification rule is to be applied to each instance where the rule context is detected in the BIM model.

In summary, the list of fields for each of the dictionary entries is as follows:

- **Rule Description:**
 - **Source document:** norm or regulation number, or specific ID to identify the original document
 - **Source section:** an ID value to identify the geometrical tolerance within the original document
 - **Description:** geometrical tolerance brief description
- **Rule Context:**
 - **Component Type:** the type of components the entry needs to be applied to (i.e. walls, slabs, etc)
 - **Material Type:** the material type of the structural components that the entry needs to be applied to (i.e. concrete, steel)
 - **Relationship Type:** the geometric relationship the different components need to be connected with that the entry needs to be applied to (i.e. above, below, same level adjacency, etc). Important remark here that in case the tolerance only involves a single element, this field can be left empty.

In our implementation, the dictionary is stored using the JSON open format, which is easy to read by users and most development tools.

Table 1 illustrates an example for a pair of the entries from EN 13670:2009 [1], where several cases of the keywords are represented for clarity.

Table 1. Quality control dictionary entries examples from EN 13670:2009

QC_1	
SourceDocument	EN 13670-2009
SourceSection	ColumnsAndWalls_a
Description	Inclination of element
ComponentType	[COLUMN, WALL]
MaterialType	[CONCRETE]
RelationshipType	[]
QC_2	
SourceDocument	EN 13670-2009
SourceSection	ColumnsAndWalls_b
Description	Deviation between centres
ComponentType	[COLUMN, WALL]
MaterialType	[CONCRETE]
RelationshipType	[ABOVE, BELOW]

2.3 Where: defining all instances of geometric QC to be conducted in a given project

Given the geometric QC dictionary, the BIM model is now analysed to identify all instances where each geometric QC rule applies, i.e. where geometric QC must be conducted. This is done by finding in the BIM model all instances where the ‘context’ defined in each geometric QC rule is found. It is performed in two steps. First, this requires the input as-design BIM model be interpreted to identify the component types and relationships of interest for the given geometric QC rules; this step is detailed in section 2.3.1. The output of the first step can then be further interpreted to identify all individual instances where the geometric specifications need to be controlled; this step is detailed in section 2.3.2.

2.3.1 BIM Model Interpretation

Here, we assume that component types and material types are explicitly encoded in the IFC file. However, component relationships are rarely provided, especially all the ones necessary for the geometric specifications we consider in the paper. As a result, specific algorithms had to be developed to detect such relationships in the model. It is proposed here to represent the outcome of this BIM model interpretation process using a network graph where each node represents a BIM model component (3D component in our case) and each edge represents a relationship

between pairs of components. The nodes and edges have properties that are defined for the intended purpose.

In the context of geometric QC of concrete or steel structure works with the above-defined geometric QC dictionary, the following properties are defined. The nodes contain the following properties: structural component **unique ID**, the **type of component** (e.g. wall, beam, column, slabs), the **material type** (e.g. concrete, steel), and a **label** (if provided in the BIM model) to facilitate the component identification for the users.

The edges contain the following properties: **source UID**, **target UID** (indicating the source and target structural components in the BIM model the edge is connected to), and the **geometric relationship type**. For the latter, we currently consider the following relationship types: *above*, *below*, *adjacent storey level*, *same storey adjacency* or *physical connection*. These relationships are those employed in the different geometric QC specifications defined in EN 13670:2009 [1].

These geometric relationships are computed by analysing each structural component (nodes), requiring various levels of computation effort. However, the details of these computations are beyond the scope of this paper (due to space constraints).

The use of the network graph has multiple advantages: First, it is easy to read. Beside, meaningful statistical information can be obtained from it overall (e.g. graph density) or for each of its nodes (e.g. centrality analysis), which can subsequently help analyse the ‘tightness’ of the geometric specifications and the ‘criticality’ of certain components.

The network graph only needs to be computed once, since the structural components are not going to change once construction is initiated (if they are the network could naturally be recomputed). Figure 3 illustrates a very basic graph example with a couple of nodes and their relationships.

Naturally, the graph structure with its nodes and edges properties can be extended to include other component types and/or properties that can be useful to satisfy other types of geometric QC (and more broadly, other use cases). The different properties that are depicted in this manuscript are the ones that were identified as minimal requirements to obtain the geometric QC of concrete structures according to the EN 13670-2009 along the duration of the project.

2.3.2 QC instances

The second step produces all individual instances of geometric QC tasks that need to be conducted for the given construction project.

For each dictionary rule, the network graph is queried to provide all unique instances of the rule ‘context’, i.e. the structural components with the same *Component Types*

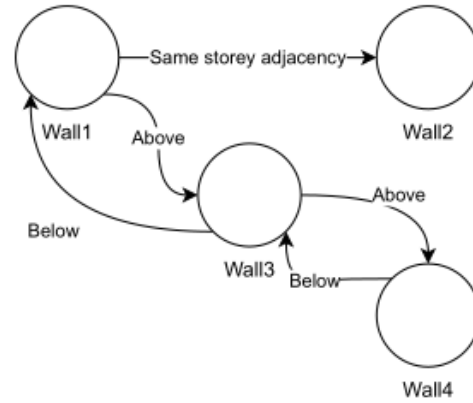


Figure 3. Example network graph of structural components and their relationships.

and *Material Types*, and, in the case the entry requires multiple components, the related components with the *Relationship Type*.

Table 2 shows a small example of the result of the search for the dictionary rules defined in Table 1 (rules QC_1 and QC_2) in the network graph of Figure 3. Table 2 shows the list of QC task associated to ‘Wall1’ only.

Table 2. Geometrical tolerances list example

QC0		
	QC_UID	Project_example_QC0
	Dictionary_ID	QC_1
	SourceDocument	EN 13670-2009
	SourceSection	ColumnsAndWalls_a
	Description	Inclination of element
	Result	
	Involved_Components	[Wall1]
	TimestampSchedule	01/01/2022
	TimestampPerformed	
	Unit	[,]
ScalarResult	[]	
ToleranceReference	[]	
AuxiliaryOutputFile		
QC1		
	QC_UID	Project_example_QC1
	Dictionary_ID	QC_2
	SourceDocument	EN 13670-2009
	SourceSection	ColumnsAndWalls_b
	Description	Deviation between centres
	Result	
	Involved_Components	[Wall1, Wall3]
	TimestampSchedule	01/01/2022
	TimestampPerformed	
	Unit	[,]
ScalarResult	[]	
ToleranceReference	[]	
AuxiliaryOutputFile		

2.3.3 When: scheduling of the QC instances

Beyond the as-designed 3D BIM model, projects increasingly develop 4D model that link the design 3D model with the construction schedule. In a 4D model, the different physical components making up the design are linked to the construction activities that construct them. These activities are defined by start date and duration (and therefore finish date). The QC Manager also queries this information from the input (4D) BIM model, so that the overall list of unique geometric QC tasks that is automatically extracted from the 3D BIM model can be organised along the project timeline. This way, once a construction task is completed (e.g. building columns ground floor), then the project manager and QC manager know the exact list of all geometric QC tasks that need to be performed at that specific point, i.e. and thus where surveying needs to be conducted. Here, we refer the reader back to the "Construction Phase" box in Figure 1.

3 Case study: Revit Sample Project Technical School

The proposed methodology has been tested in the Revit Sample Project Technical School [14]. This model is a sample model provided by Autodesk. We focus on the structural model (Figure 4), since the rules considered in these tests focus on structural works.

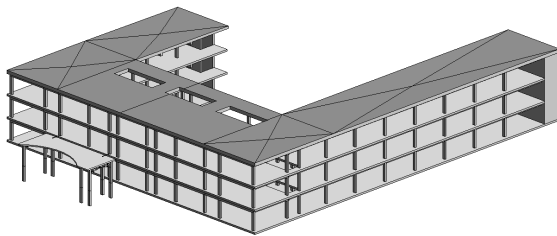


Figure 4. Technical School sample model

The (preliminary) version of the dictionary used in this test contains a total of 15 rules, all of them selected from the EN 13670-2009 document [1]. The selected rules involve four types of structural components: (*slabs, columns, walls, beams*) and five types of geometrical relationships: *above, below, adjacent storey level, same storey adjacency, and physical connection*. The rules include (reference to document section between brackets):

- Inclination of a column/wall (10.4 Columns and Walls No a)
- Deviation between centres (10.4 Columns and Walls No b)
- Curvature of a column/wall between adjacent storey levels (10.4 Columns and Walls No c)

- Location of a column/wall at any storey level w.r.t. base level (10.4 Columns and Walls No d)
- Location of a beam-to-column connection measured relative to the column (10.5 Beams and Slabs No a)
- Position of bearing axis of support (10.5 Beams and Slabs No b)
- Cross-sectional dimensions (10.6 Sections No a)
- Lap-joints (10.6 Sections No c)
- Free space between adjacent columns/walls (Annex G - G.10.4 Columns and Walls No c)
- Horizontal straightness of beams (Annex G - G.10.5 Beams and Slabs No a)
- Distance between adjacent beams (Annex G - G.10.5 Beams and Slabs No b)
- Inclination of a beam/slab (Annex G - G.10.5 Beams and Slabs No c)
- Level of adjacent beams (Annex G - G.10.5 Beams and Slabs No d)
- Level of adjacent floors at supports (Annex G - G.10.5 Beams and Slabs No e)
- Orthogonality of a cross-section (Annex G - G.10.6 Sections No a)

The test consisted of loading the different structural components from the IFC file, compute their geometric relationships and construct the network graph. The graph could then be analysed to generate the list of geometric QC tasks that would need to be conducted over the duration of the project.

The model contains a total of 589 structural components. Our algorithm reads the IFC file (exported from Revit) and generated the network graph containing those 589 components as nodes. Beside, the algorithms found 6,677 relevant geometrical relationships converted into 6,677 corresponding edges in the graph. Those nodes and relationships are broken down as follows:

• Components:

- 6 Walls
- 5 Slabs
- 203 Columns
- 375 Beams

• Relationships:

- 126 Above
- 126 Below
- 5 Adjacent storey level
- 6,013 Same storey adjacency
- 407 Physical connection

Figure 5 shows the graph structure distribution and its colour reference legend, where we can visualise all the structural components contained in the BIM model represented by the coloured nodes, and the geometric relationships connecting them represented by the coloured arrows

between the nodes. We can also identify the high amount of columns (orange nodes) and beams (blue nodes) that is contained in the BIM model, representing more than the 95% of the structural components, and the different relationships between them. Each node has a size proportional to its *degree centrality* (i.e. the number of edges connected to it).

Approximately 90% of the edges belong to the same storey adjacency category, hence the graph is full of those edges making it difficult to visualise the other relationships. Using the graph visualisation and analysis tool Gephi [15], the graph is explored and analysed, extracting interesting statistics and exploring different arrangements. For example, Figure 5 shows a *Force Atlas 2* layout, where we can easily depict the different storeys by the way the columns form three distinct clusters. Degree centrality can also be further investigated. Nodes with high degree centrality can be considered as potentially more critical from a geometric QC viewpoint, so the building team may be interested to know which ones they are and the reasons for their criticality. For example, Figure 6 shows a highlighted component (on the top) with high degree centrality, indicating it can be a critical structural component from a geometrical tolerance compliance viewpoint, while on below, we can see the same component identified in the as-design BIM model, where it may be easier for the contractor to understand its criticality.

After generating the network graph, it can be analysed according to the quality control dictionary to output a full list of geometric QC checks that need to be conducted. In this example, the list contains 9,750 unique geometric QC checks. These unique entries in the list demonstrate the total number of quality control actions that should be carried out during the entire project's execution and that should all output a positive result to certify the quality of execution. Table 3 shows a snapshot of the generated JSON file containing all the project's geometrical tolerances.

It is important to remark that despite the limited number of geometric QC rules in the dictionary, the total number of unique geometric QC checks is still significant. This numbers suggests that all checks are not systematically conducted in practice, with many results potentially implied from some others. The rest of the GeometricQC tool aims to automate these control and will ensure they are systematically conducted.

4 Conclusion and future work

The tool proposed in this paper is meant to be employed during project planning stage and has two steps: First, it automatically extracts all the necessary information from the design BIM model (IFC file) and stores it in a convenient data structure, a network graph. Then, the tool automatically detects within the network graph all

Table 3. Technical School geometrical tolerances list example

QC0		
	QC_UID	Rst_advanced_sample_project_QC0
	Dictionary_ID	QC_1
	SourceDocument	EN 13670-2009
	SourceSection	ColumnsAndWalls_a
	Description	Inclination of element
	Result	
	Involved_Components	[01SfNHv5nEReC9M9Bzo7D_]
	TimestampSchedule	17/01/2022
	TimestampPerformed	
	Unit	[,]
	ScalarResult	[]
	ToleranceReference	[]
	AuxiliaryOutputFile	
QC1		
	QC_UID	Rst_advanced_sample_project_QC1
	Dictionary_ID	QC_15
	SourceDocument	EN 13670-2009
	SourceSection	Sections_annex_a
	Description	Orthogonality of a cross-section
	Result	
	Involved_Components	[01SfNHv5nEReC9M9Bzo7D_]
	TimestampSchedule	17/01/2022
	TimestampPerformed	
	Unit	[,]
	ScalarResult	[]
	ToleranceReference	[]
	AuxiliaryOutputFile	
QC100		
	QC_UID	Rst_advanced_sample_project_QC100
	Dictionary_ID	QC_9
	SourceDocument	EN 13670-2009
	SourceSection	ColumnsAndWalls_annex_c
	Description	Free space between adjacent elements
	Result	
	Involved_Components	[15kM\$yjmD0FxFxNxtEUi7rG8, 15kM\$yjmD0FxFxNxtEUi7rIs]
	TimestampSchedule	17/01/2022
	TimestampPerformed	
	Unit	[,]
	ScalarResult	[]
	ToleranceReference	[]
	AuxiliaryOutputFile	
QC8872		
	QC_UID	Rst_advanced_sample_project_QC8872
	Dictionary_ID	QC_7
	SourceDocument	EN 13670-2009
	SourceSection	Sections_a
	Description	Cross-sectional dimensions
	Result	
	Involved_Components	[3KQkoT3ZD758EdANtsCisU]
	TimestampSchedule	17/01/2022
	TimestampPerformed	
	Unit	[,]
	ScalarResult	[]
	ToleranceReference	[]
	AuxiliaryOutputFile	

unique instances of the contexts within which each geometric specification applies and QC must be conducted. These geometric specifications are stored as rules in a QC

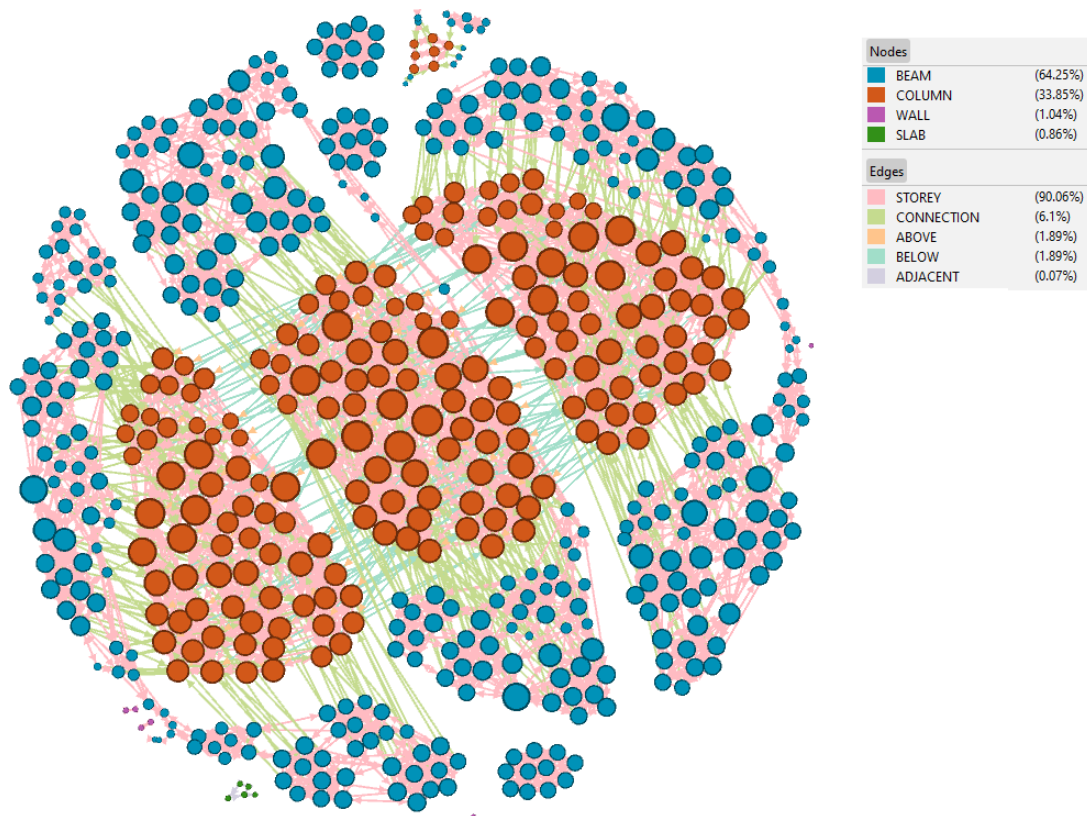


Figure 5. Graph representation with a Force Atlas 2 layout

dictionary, with the overall geometric QC methodology employed by the proposed GeometricQC tool akin to Automated Rule Checking (ARC). The value of the proposed tool to identify all necessary geometric QC tasks within a given project is demonstrated with a realistic mid-size project (school building).

In future works, the planning stage methodology will be extended to include the rest of the structural deviations presented in EN 13670-2009 [1] and EN 1090-2:2018 [12], in addition to the rest of the geometrical relationships, and a fully integrated schedule from the 4D BIM model. Then, the full pipeline of the GeometricQC tool will be delivered that will conduct the automatic geometric QC with point clouds acquired on site. Finally, as part of the large COGITO project [9], the tool will be integrated within a Construction Project Digital Twinning ecosystem enabling construction activities, including QC, to be effectively orchestrated and their output recorded in a structured way with the project Digital Twin.

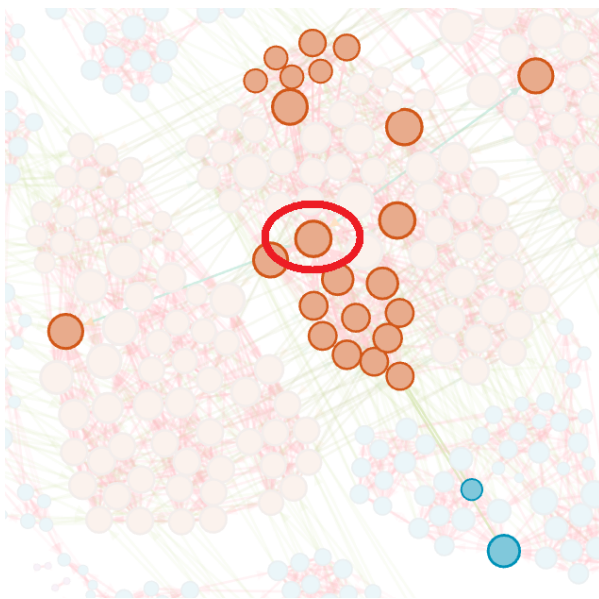
Acknowledgements

The research leading to these results has received funding from the European Union's Horizon 2020 research and

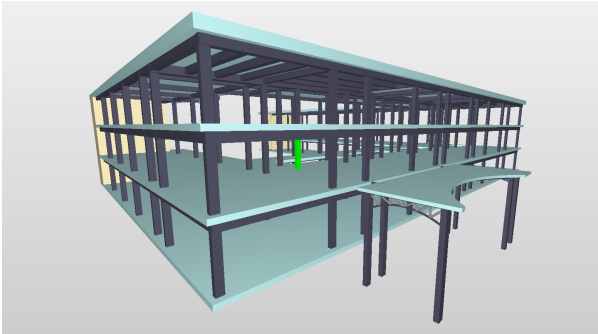
innovation programme under grant agreement No 958310. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

References

- [1] European Committee for Standardization. EN 13670:2009 - Execution of concrete structures. Standard, CEN, 2010.
- [2] Agarwal R., S. Chandrasekaran, and S. Sridhar. Imagining construction's digital future. Technical report, KcKinsey & Company, 2016.
- [3] Aimi Sara Ismail, Kherun Nita Ali, and Noorminshah A. Iahad. A review on bim-based automated code compliance checking system. In *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*, pages 1–6, 2017. doi:10.1109/ICRIIS.2017.8002486.
- [4] Robert Amor and Johannes Dimyadi. The promise of automated compliance checking. *Developments*



(a) Column node selection with all its geometric relationships



(b) Column selection in BIM model

Figure 6. Column component inspection in (a) graph and (b) model.

in the Built Environment, 5:100039, 2021. ISSN 2666-1659. doi:10.1016/j.dibe.2020.100039.

- [5] Frédéric Bosché. Automated recognition of 3d cad model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction. *Advanced Engineering Informatics*, 24(1):107–118, 2010. ISSN 1474-0346. doi:10.1016/j.aei.2009.08.006. Informatics for cognitive robots.
- [6] M.-K. Kim, Q. Wang, J.-W. Park, J.C.P. Cheng, H. Sohn, and C.-C. Chang. Automated dimensional quality assurance of full-scale precast concrete elements using laser scanning and bim. *Automation in Construction*, 72:102–114, 2016. doi:10.1016/j.autcon.2016.08.035.
- [7] Changmin Kim, Hyojoo Son, and Changwan Kim.

Automated construction progress measurement using a 4d building information model and 3d data. *Automation in Construction*, 31:75–82, 2013. ISSN 0926-5805. doi:10.1016/j.autcon.2012.11.041.

- [8] Maarten Bassier, Stan Vincke, Heinder De Winter, and Maarten Vergauwen. Drift invariant metric quality control of construction sites using bim and point cloud data. *ISPRS International Journal of Geo-Information*, 9(9), 2020. ISSN 2220-9964. doi:10.3390/ijgi9090545. URL <https://www.mdpi.com/2220-9964/9/9/545>.
- [9] Construction-phase digital twin model: Cogito. On-line: <https://cogito-project.eu/>, Accessed: 09/02/2022.
- [10] J. Zhang and N. M. El-Gohary. *Automated Extraction of Information from Building Information Models into a Semantic Logic-Based Representation*, pages 173–180. ASCE, 2015. doi:10.1061/9780784479247.022.
- [11] Jiansong Zhang and Nora M. El-Gohary. Semantic-based logic representation and reasoning for automated regulatory compliance checking. *Journal of Computing in Civil Engineering*, 31(1):04016037, 2017. doi:10.1061/(ASCE)CP.1943-5487.0000583.
- [12] European Committee for Standardization. EN 1090-2 - Execution of steel structures and aluminium structures. Technical requirements for steel structures. Standard, CEN, 2018.
- [13] buildingSMART. Industry foundation classes (ifc). On-line: <https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/>, Accessed: 17/02/2022.
- [14] Autodesk. Revit sample project files. On-line: <https://knowledge.autodesk.com/support/revit/getting-started/caas/CloudHelp/cloudhelp/2022/ENU/Revit-GetStarted/files/GUID-61EF2F22-3A1F-4317-B925-1E85F138BE88-htm.html>, Accessed: 12/02/2022.
- [15] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An open source software for exploring and manipulating networks. In *International AAAI Conference on Weblogs and Social Media*, 2009. URL <http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154>.