This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction

Oscar Wong Chong, S.M.ASCE[1], Jiansong Zhang, Ph.D., A.M.ASCE[2]

1. Introduction

In the past decades, a concerning labor shortage is experienced by the labor-driven Architecture, Engineering, and Construction (AEC) industry [4,5,6]. For instance, in the United States, 80% of the construction companies cannot find trades workers to fill job positions [4]. Just for January of 2020, 267,000 unfilled construction positions were reported in the U.S. [5]. This problem has impacted negatively the productivity in the AEC industry, causing delays and cost overrun in construction projects [6]. In a survey conducted by Construction Labor Market Analyzer, more than 90% of the respondents reported a lower productivity for the years 2014 and 2015 due to the labor shortage [6].

Technologies such as offsite construction and automation allow greater efficiency by automating construction processes, which can be translated into higher productivity in the industry. This increase in productivity can relieve some of the strain imposed by the workforce shortage [20,21,22]. On the one hand, offsite construction provides many benefits over conventional stick-built construction such as working in a controlled environment, the ability to conduct activities in parallel, and improvement of the built quality [15,16,17]. Automation in construction, on the other hand, includes the application of technologies such as Computer Numerical Control (CNC), robotics, and other automation machines that could be easier to implement in a factory setting than in onsite construction. For example, these technologies can be adopted to support the fabrication and assembly of building components by automating otherwise manual operations. Moreover, automation can improve safety by saving workers from dangerous and heavy-duty tasks and/or in hazardous conditions.

However, despite these benefits, there are many challenges with the automation of offsite construction in practice. One main challenge is that offsite construction demands more rigorous

[1] Graduate Student, Automation and Intelligent Construction Lab (AutoIC), School of Construction Management Technology, Purdue University, West Lafayette, IN. 47907; email: owongcho@purdue.edu

[2] Assistant Professor, Automation and Intelligent Construction Lab (AutoIC), School of Construction Management Technology, Purdue University, West Lafayette, IN. 47907; email: zhan3062@purdue.edu

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

design, planning [13], and construction requirements than those of onsite construction. For example, the design, manufacturing, and assembly tolerances for offsite construction are tighter than those for stick built because the assembled components need to appropriately fit the prepared foundation rather than the components sequentially erected onsite, in which the latter option allows more flexibility for local adjustments of the connections between the frames and the foundation. In addition, automation requires detailed and precise information such as building information models, material, and building systems to obtain the desired outcomes [14]. These challenges have impeded the wide adoption of automation in offsite construction.

Building Information Modeling (BIM) has the potential to overcome these limitations and enable automation in construction by providing detailed, precise, and complete information as input for automation technologies in the context of offsite construction [14]. However, the support of BIM for offsite construction is still limited in the current digital workflow. For instance, BIM lacks the capability to represent complex buildings or to plan for automated processes in offsite construction [15]. Therefore, to address this gap, the authors propose a method to facilitate the automation of wood construction by automatically analyzing building design information to obtain construction operational level information from the analysis so that it can be further used to feed into construction automation technologies. The proposed method utilizes a logic enabled approach to extract and infer information from IFC-based BIM models. The method involves the development of: (1) a set of algorithms (using logic rules) for the automated information extraction and properties inference, (2) representation of the IFC-based BIM models into logic facts, and (3) the logic reasoning using the logic rules and facts.

2. Background

2.1 Offsite Construction Automation

Offsite construction refers to the manufacture and preassembly of building components in a controlled environment, which are then transported and assembled on-site [16]. Offsite construction can be classified as non-volumetric, volumetric, or modular buildings based on the type of element and the level of prefabrication on the building [17]. For instance, the volumetric preassembly consists of structurally enclosed units (modules) [18]. In addition, offsite construction can be realized using concrete, steel, or wood, among other materials. Concrete and steel are predominately used in commercial and industrial facilities such as high-rises, warehouses, and bridges. While wood has been primarily used in residential houses to mid-rise buildings, especially in North America and Europe [19]. For instance, in the United States, nine out of ten houses are built of wood [20]. In addition, wood is a more sustainable and energy

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

efficient material than concrete and steel in terms of the level of carbon dioxide emission [21] and embodied energy [22], respectively. These advantages have made wood one of the most commonly used construction materials.

The adoption of prefabrication and digitalization allows for more automation opportunities [23] in the manufacturing and assembly processes of wood construction. Some commonly implemented automation technologies in offsite wood construction include the use of robots, CNC, and other machines. Industrial arms are the most common type of robots used to automate the assembly and material handling operations in production lines or workstations. For instance, in Willmann et al. [24], a robotic system was used to assemble "The Sequential Roof", a timber roof structure that consists of slat elements. More applications of robotics in the automation of wood construction can be found in [25-27]. Moreover, CNC tools are used to remove layers of material (e.g., drilling, milling, and cutting) to shape the pieces according to designs (i.e., subtractive manufacturing) [28]. The implementation of CNC machines in wood manufacturing lines provide automated prefabrication of building elements (e.g., wood pieces and boards) [27]. Furthermore, other machines such as the semiautomated wood framing machine developed by [29], also facilitate wood framing processes. A common property shared by these automation technologies is that they require reliable and precise digital information as input for its successful operations. This requirement can be fulfilled using BIM as the source of information.

Offsite construction can be partially industrialized [30], meaning that the prefabricated components and assembled units can be treated as manufacturing products instead of a construction product. Consequently, offsite construction opens more opportunities for introducing manufacturing technologies, principles, and methods in the prefabrication of houses. Currently, the adoption of tools and technologies from the manufacturing industry for modular construction such as production lines, mechanical engineering and manufacturing CAD packages, and other prefabrication-based technologies, creates time and cost savings compared to traditional on-site construction. However, these technologies lack the capacity to analyze building designs, which limit their applicability in the design development stage [14]. This inability of these automation technology to fully integrate with BIM creates inefficiency caused by a bottleneck of information transfer within and between the design and construction phases.

2.2 Industry Class Foundation (IFC)

BIM is a "modeling technology and associated set of processes to produce, communicate, and analyze building models" [14] and it serves as a "shared knowledge resource for information about a facility forming a reliable basis for decisions during its life-cycle from inception onward" [31]. As such, BIM has the potential to improve the collaboration between designers and

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

contractors and to allow a seamless coordination (e.g., data exchange) between design and construction phases in offsite construction projects. However, this potential is not yet fully realized due to proprietary concerns and interoperability difficulties in the AEC industry [32]. As a result, it creates significant re-work, measurable waste, and has impeded the use of BIM to advance automation [20,25,26]; costing the AEC industry $15.8 billion per year [34]. To improve BIM data exchange, existing approaches focus on data schema standardization and/or term-based semantics of AEC objects [35]. The CIMSteel Integration Standards (CIS/2) for steel construction data and the Industry Foundation Classes (IFC) for building and construction data are two prominent BIM standards. Both standards are defined using the Standard for Exchange of Product (STEP) description methods - ISO 10303 [36].

Industry Foundation Classes (IFC) is a vendor-neutral standard for data exchange developed and maintained by buildingSMART as a solution to the interoperability problem in the AEC industry [44,45]. The IFC schema is written using the EXPRESS data definition language and its data schema architecture contains four conceptual layers, namely resources, core, interoperability, and domain, to describe information such as geometry, material, and relationship of a BIM model [39]. IFC contains an essential set of elements such as beam, column, wall, floor, and roof, to describe a building. Furthermore, each element can be represented using different geometric representations, such as swept solid, Boundary representation (B-rep), or body clipping. Moreover, multiple cross-section profile definitions exist for each geometric representation. For instance, a column modeled as Swept Solid representation, can have a rectangular profile definition (IfcRectangleProfileDef) or an arbitrary closed profile definition (IfcArbitraryClosedProfileDef) to depict its cross section. These variety of representations come from the 3D modelling approaches adopted by the BIM authoring tools. To illustrate the IFC data model, an example of IFC instances with its corresponding entities, relationship, and tracing pattern for the dimensions of column element are shown Figure 1 and Figure 2, respectively.

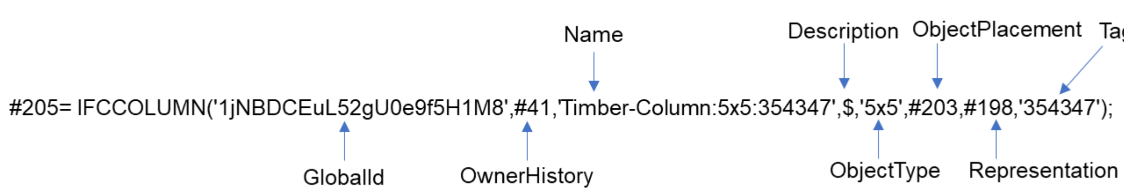


Figure 1. An IFC instance sample.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

XDim YDim Depth Name and ID

```
#128= IFCAXIS2PLACEMENT3D(#6,$,$);
#129= IFCLOCALPLACEMENT(#32,#128);
#144= IFCRECTANGLEPROFILEDEF(.AREA.,'5x5',#143,0.375,0.375);
#148= IFCEXTRUDEDAREASOLID(#144,#147,#19,10.0457511378774);
#158= IFCSHAPEREPRESENTATION(#109,'Body','SweptSolid',(#148));
#161= IFCCARTESIANPOINT((-3.3825575870149,3.34750587084454,-9.));
#163= IFCVECTOR(#19,1.);
#164= IFCLINE(#161,#163);
#165= IFCCARTESIANPOINT((-3.3825575870149,3.34750587084454,0.));
#167= IFCCARTESIANPOINT((-3.3825575870149,3.34750587084454,10.0457511378774));
#169= IFCTRIMMEDCURVE(#164,(#165),(#167),.T.,.CARTESIAN.);
#175= IFCAXIS2PLACEMENT3D(#6,$,$)
#176= IFCREPRESENTATIONMAP(#175,#158);
#172= IFCSHAPEREPRESENTATION(#107,'Axis','Curve3D',(#169));
#178= IFCAXIS2PLACEMENT3D(#6,$,$);
#179= IFCREPRESENTATIONMAP(#178,#172);
#180= IFCCOLUMNTYPE('1jNBDCEuL52gU0g9f5H1M8',#41,'5x5',$,$,$,(#176,#179),'354163','5x5',.COLUMN.);
#189= IFCCARTESIANTRANSFORMATIONOPERATOR3D($,$,#6,1.,$);
#190= IFCMAPPEDITEM(#176,#189);
#192= IFCSHAPEREPRESENTATION(#109,'Body','MappedRepresentation',(#190));
#194= IFCMAPPEDITEM(#179,#189);
#196= IFCSHAPEREPRESENTATION(#107,'Axis','MappedRepresentation',(#194));
#198= IFCPRODUCTDEFINITIONSHAPE($,$,(#192,#196));
#202= IFCAXIS2PLACEMENT3D(#6,$,$);
#203= IFCLOCALPLACEMENT(#129,#202);
#205= IFCCOLUMN('1jNBDCEuL52gU0e9f5H1M8',#41,'Timber-Column:5x5:354347',$,'5x5',#203,#198,'354347');
```

Figure 2. Tracing pattern of the IfcColumn dimensions information.

The current IFC format presents several limitations. One limitation is the existence of a user defined properties set, which make its use unpredictable. Another limitation is the multiple ways in which a building element can be represented. For example, a wall might be represented as a wall, a thick slab, or an upstand beam [40]. This common misuse introduce subjectivity in the IFC standard. Other limitations of IFC include the needs of improved information description, a more robust way to represent elements, and information and precision loss/inadequacy [41]. These limitations make the standardization of IFC challenging.

Even with the current limitations, the IFC schema is widely accepted as the most promising solution to the interoperability issue faced by the AEC industry [42]. Qualities such as openness, impartiality, and file format simplicity, have made the IFC standard a major focus of BIM research and industry applications. Currently, a growing number of BIM platforms (94) are being certified as IFC compliant; making them compatible with IFC [43].

2.3 BIM to Support Offsite Construction Automation

Many research efforts have used BIM-based approaches to improve offsite construction processes at different lifecycle stages. This section presents some of the applications of BIM workflows in offsite construction.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

In the design and planning stages of offsite construction, Liu et al. [44] developed a rule-based algorithm and automated BIM-based approach to minimize waste in the design and planning of light-frame boarding (sheathing and drywall), taking into consideration contractors' practical knowledge. Their method helps reduce errors and time consumption in manual modeling of a construction-centric model and it is implemented as a Revit add-on. As an attempt to reduce the design cost, improve the layout accuracy and productivity, Alwisy et al. [13] proposed a framework to automate the design and drafting of wood frame panel modules. The input to the BIM model and the corresponding shop drawings for the manufacturing process is generated from 2D CAD layout drawings. Their methodology was implemented using Visual Basic embedded into AutoCAD. In addition, Abushwereb et al. [45] proposed a platform (FrameX) as a Revit add-on to automate the analysis, modelling, and design of light-frame wood structure for offsite construction. Their platform was developed using a rule-based approach that arguably improve time efficiency and accuracy in the early design stage of a project.

Besides design and planning, other research works have focused on the optimization of offsite construction. For instance, to optimize offsite construction designs for meeting client expectations, Isaac et al. [46] used a graph-based methodology based on BIM models to reduce delays and to avoid incurring additional cost and labors in the prefabrication of modular house modules. Another optimization approach is the integration of manufacturing CAD into the workflow (e.g., SolidWorks and TactonWorks Studio) to automate the selection of module configurations during design. This integration method is proposed by [47] to reduce conflicts between stakeholders and variability in the downstream process of the prefabrication building components. Furthermore, Mekawy and Petzold [48] proposed a method to explore and optimize design alternatives of Box Prefabricates using their developed Autodesk Dynamo for Revit package called Box Module Generator. In addition, to reduce waste in the construction phase, Gbadamosi et al. [49] presented a framework for the assessment and optimization of design (BIM models) options based on lean principles and design for manufacturing concepts such as ease of assembly, ease of handling, speed of assembling, and assembly waste. Their assembly assessment framework was implemented using Revit, Dynamo Studio, and Microsoft Excel for the design optimization of exterior insulation finish systems.

The use of BIM in other modular construction processes include quantity take-off and off-site manufacturing. In the first case, Wang et al. [50] proposed a method to reduce manual work in the automated quantity take-off process of wall and floor components using a SQL database and BIM. In the second case, Root et al. [51] proposed a method to better understand the offsite construction process of exterior insulated finishing systems manufacturing using Revit and the Metal Wood Framing plug-in from Strucsoft.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Although many research efforts have contributed to the advancement of BIM and offsite construction, most of the existing efforts only focused on workflows using proprietary BIM platforms and processes in the design and planning stages. The dependency on proprietary BIM applications prevents the realization of a truly seamless BIM interoperability. In addition, the lack of BIM research focused on the construction phase make the implementation of automation in offsite construction a challenge. To address this limitation, this paper proposes a new methodology for BIM analysis to support wood construction automation, using a logic-based approach and IFC standard.

2.4 Logic Representation

Logic has been used for computer design and computer programs reasoning [52]. From a programming language perspective, the direct use of logic is called logic programming [52]. Logic programs consist of rules that establish relations between objects [52]. Formal logic such as predicate logic, allows the representation of knowledge and the derivation of correct conclusions from that knowledge [53]. First-order logic (FOL), a subset of predicate logic, is the most common type of logic representation [54].

2.4.1 *First order logic (FOL)*

FOL can be used to represent IFC-based BIM models information, in the form of logic clauses, which in turn consists of predicates. The relations between the predicates are logically expressed using quantifiers and logic connectives. In FOL, the universal (∀ or for all) and existential (∃ or there exists) quantifiers are used to make assertions about variables in statements [55]. Likewise, the logic connectives: conjunction AND (∧), disjunction OR (∨), negation NOT (¬), and implication (→), are used to make logical connections between predicates [55].

Furthermore, logic clauses can be expressed using Horn clauses (HCs). A HC is a conjunction of logic clauses of which at most one literal is positive [56]. HC can be expressed as $H \leftarrow (C_1 \wedge C_2 \wedge C_3 \cdots \wedge C_n)$, where H is the head of the clause and $C_1, C_2, C_3 .. C_n$ are goals. This expression implies that the conclusion H holds if all the goals are met. The structure of a HC is simple and sufficiently expressive to represent all types of computations that allows programs to be general and efficient [57]. There are three types of HCs based on the representation structure: facts, rules, and queries [58].

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

2.4.2 *Basic elements of logic programs*

A HC can be classified as facts, queries, or rules, based on its structure. In a HC, when only the head *H* is present, it is a fact. Logic facts are "statement that describe object properties or relations between objects" [54]. It consists of predicates that are composed of a predicate name and one or more arguments. The number of arguments in a predicate is called arity and the smallest unit of a logic fact is an atom, which consists of only one argument. In the case when a HC only contains a set of goals $(C_1 \wedge C_2 \wedge C_3 \cdots \wedge C_n)$, it is called a query. Queries are used to retrieve information from a logic program and it will reach a conclusion whether relations between objects hold (conjunction of goals). Lastly, a HC is classified as a rule when it contains both the head *H* and the body (i.e., a set of goals to be evaluated according to the logic facts). If all the goals are met, then the rule evaluates to true. On the contrary, if any of the goals cannot be achieved, the rule evaluates to false.

2.4.3 *Second order logic (SOL)*

An extension of FOL in terms of expressive power is the second-order logic (SOL). Unlike FOL, whose domain of quantification is the range of individuals, SOL can quantify subsets of individuals with certain properties or relations over the entire domain [59]. Therefore, SOL is particularly useful to find all instances of building components/elements with certain properties from building design logic facts. The application of SOL in logic programming is referred as second-order programming, which is represented by the "find all-solutions" predicates such as findall(Term,Goal,List) [52]. To illustrate this, the clause *column_material :- findall((Column,Materialname),(relassociatesmaterial(Relassociatesmaterial),material(Material),has_relatingmaterial(Relassociatesmaterial,Material),column(Column),has_relatedobjects(Relassociatesmaterial,Column),has_name(Material,Materialname),Materialname == 'lumber'), L)* will function as finding all the column instances from the existing logic facts with the material property 'lumber' and store them in the list *L*.

2.5 Logic Reasoning

The use of logic representation and reasoning facilitates the analysis of building design information. Once the logic representation of the IFC data is enabled and the logic rules are defined, the logic reasoning is performed automatically. The essence of logic reasoning relies on the unification function and three deduction rules: 1) identity, 2) generalization, and 3) instantiation [52].

According to Sterling and Shapiro [52], the identity rule consists of the search of logic facts based on queries to determine logical consequences. The second deduction rule is generalization, which relates a logical consequence to an instance of an existential quantified variable for any substitutions. Lastly, the instantiation rule can be used to deduce any instance of a logic fact from a universally quantified fact. The automated deduction in logic reasoning is possible through the unification algorithm [52]. Unification provides efficient pattern matching and variable binding functions [60] to allow the reasoning based on logic rules and facts.

2.5.1 *Prolog language*

A partial, yet powerful realization of logic programming is through prolog. Prolog (stands for programming in logic) language is a programming formalism based on the concept of logic programming created in the early 1970s by Alain Colmerauer [52]. The use of prolog has been proven to be successful in applications such as artificial intelligence (AI), language processing, and expert systems [59,60]. Prolog differentiates from other programming languages through a different programming paradigm: it is declarative as opposed to the more conventional procedural and objected-oriented approaches.

2.5.2 *Reasoning using a closed-world assumption*

By default, prolog adopts a closed-world assumption for the logic reasoning. A closed-world assumption considers any unproven assumptions to be false. Therefore, any missing information will be considered as false as well. This implies that the information to be reasoned about needs to be complete and logical.

2.6 Logic-Based Representation for Automated Reasoning

Logic-based representation has been widely and extensively used for automated reasoning. Automated reasoning is the ability to make inferences automatically through computing systems and it has been applied to solve many challenging problems in domains of computer science, mathematics, software and hardware verifications, among others [63]. Several disciplines (e.g., sensing, natural language processing, robotics) along with automated reasoning, formed the fundamental building blocks in the conception and rise of AI in the modern era, allowing the creation of intelligent entities that can perceive the environment and perform actions autonomously [64]. In the AEC domain, the application of AI methods (e.g., neural networks, genetic algorithms, and machine learning) have increased exponentially since the early 2000s (e.g., more than 41,827 existing related bibliographic records in the Scopus database) [65]. Although some computing paradigms such as machine learning and other statistical-based computing method are trending, logic-based representation and reasoning through Logic programming is a no-less powerful computing paradigm due to its rigor, expressiveness, and ability to draw logical conclusions. Yet, comparatively speaking, it is significantly underexplored. Therefore, the authors are testing the use of logic-based representation and reasoning for automated inferences to unlock the benefits of logic-based automation in the construction domain.

2.7 Logic-Based Representation and Reasoning in the AEC Domain

Previous studies have explored the use of FOL in the AEC domain. One of the early applications of FOL is in the area of structural engineering design [61,62]. More recently, FOL has been used in the representation and reasoning of building design and regulatory information in the area of code compliance checking [54,63,64].

2.8 Comparison to ifcOWL Ontology

In addition to logic-based approach, semantic modeling can be used to support logical inferences. The most commonly used form of semantic modeling is ontology and it uses sematic web technology [i.e., Web Ontology Language (OWL)] for representation of things and their relations [67]. In an ontology, knowledge is represented in concept hierarchies and the relationships between the concepts, and axioms [68].

Both, semantic representation and logic representation could be utilized to support the reasoning process and facilitate human interpretation and understandability of the formal representation. However, they differ in their application intent fundamentally. Semantic modeling was originally conceived based on description logic to mainly model knowledge and represent the semantic of a specific domain. As a result, semantic representation (i.e., OWL ontology) relies on rule-based languages such as Semantic Web Rule Language (SWRL) to perform automated reasoning on top it because it does not permit representation of if-then statements directly [69]. In contrast to semantic modeling, a logic-based representation has been used extensively in automated reasoning. The use of logic-based representation in automated theorem provers have proved many famous scientific problems such as the problem of Robbins conjecture in 1966 [70]. In addition, logic-based representation has been successfully used to verify the correctness of computer programs and to assist in the construction of mathematical proofs [71]. Despite these achievements, it is still underexplored in the automation of architecture, engineering, and construction projects, compared with other popular approaches such as statistical machine learning and semantic modeling.

In an effort to link IFC standard and semantic web technologies to support flexibility, interoperability, and reusability of data and data exchange [72], and also to support logic-based reasoning, the ifcOWL ontology was created. The ifcOWL ontology and IFC schema are more similar in structure [73] compared to that between logic facts and IFC. In addition, the

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

conversion from the IFC EXPRESS schema to ifcOWL ontology seems to be more direct and easier than the transformation of the IFC schema to logic facts from the logic-based approach. However, despite these advantages, there are challenges in its implementation for logic inference; for example: 1) the size of an ontology is considerably large and complex to load and use due to its numerous classes, objects and data properties, and logical axioms, and 2) the time efficiency is comparatively low due to the large number of assertions in the loading (to reasoning engine) and reasoning processes [69,70]. Moreover, ifcOWL ontology needs separate and additional rule representations (e.g., SWRL) in the reasoning process, which are also limited in terms of expressivity, simplicity, and computational logic reasoning performance compared to the use of first-order logic and second-order approach. For example, SWRL is not as effective as FOL in decidability (i.e., true or false decision problems). In addition, it is easy to find all instances of building components/elements with certain properties from the building design using SOL, which in contrast is not feasible with SWRL.

3. Proposed Method

In order to extract and analyze building design information (e.g., geometric and physical properties) from BIM to support automation in offsite wood construction, the authors proposed a novel data-driven method to address this challenge. The proposed method is based on the use of logic representation and reasoning of BIM information in an automated fashion. Moreover, the proposed method allows performance improvement by iteratively refining and extending the refinement algorithm and logic rules, respectively. The proposed method consists of six main steps (Figure 3): (1) Logic Facts Generation, (2) Logic Rules Development, (3) Logic Reasoning Execution, (4) Result and Error Evaluation, (5) Logic Facts Refinement, (6) Logic Rules Extension. Following, detailed explanations of the six steps along with some implementation examples to illustrate the specifics of the proposed method are provided.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

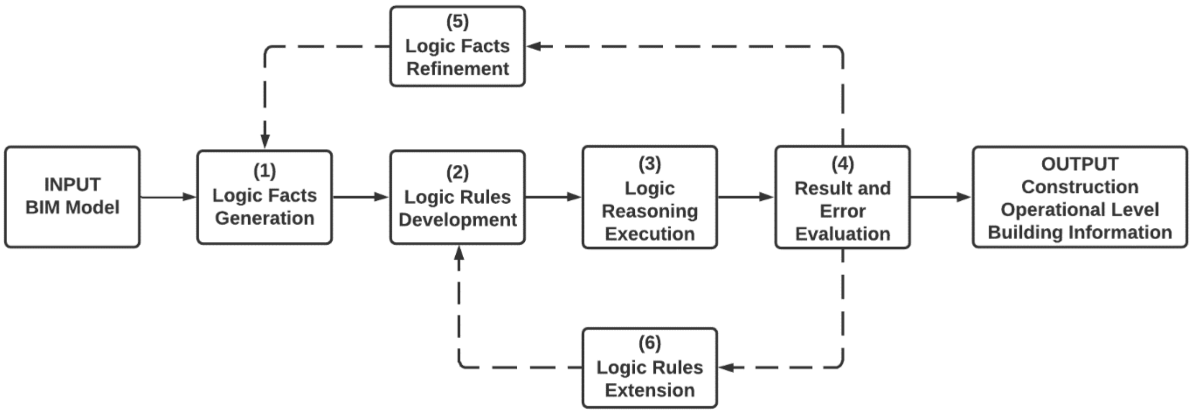


Figure 3. Proposed method.

3.1 Logic Facts Generation

The generation of logic facts consists of three sub-steps: sub-step (1) BIM to IFC export, sub-step (2) IFC to logic facts conversion, and sub-step (3) logic facts refinement.

3.1.1 *BIM to IFC export*

The first sub-step consists of exporting the BIM model from the BIM authoring tool to IFC format. Many BIM authoring tools such as Autodesk Revit, Graphisoft ArchiCAD, and Trimble SketchUp are IFC certified and therefore have built-in functions for exporting their respective BIM models to IFC format [43]. In addition, different parameters can be adjusted such as the specific IFC schema version and model view definitions when exporting to IFC format. For the implementation of the proposed method, the IFC 2x3 and MVD CV2.0 were selected.

3.1.2 *IFC to logic facts conversion*

In this sub-step, the IFC data is represented using logic facts. The conversion of the IFC model into logic facts is performed using the open source application ZE_BIM_FOL_Converter2.0, developed by [75].

The prototype system ZE_BIM_FOL_Converter2.0 implements an information extraction (IE) and information transformation (ITr) algorithms of the building design information (IFC) [75]. In Figure 4, an example of the IFC data conversion into logic facts by applying the IE and ITr algorithms in four steps is provided.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Lines in IFC-Based BIM

```
#148= IFCEXTRUDEDAREASOLID(#144,#147,#19,10.0457511378774);
#158= IFCSHAPEREPRESENTATION(#109,'Body','SweptSolid',(#148));
#176= IFCREPRESENTATIONMAP(#175,#158);
#190= IFCMAPPEDITEM(#176,#189);
#192= IFCSHAPEREPRESENTATION(#109,'Body','MappedRepresentation',(#190));
#198= IFCPRODUCTDEFINITIONSHAPE($,$,(#192,#196));
#205= IFCCOLUMN('1jNBDCEuL52gU0e9f5H1M8',#41,'Timber-Column:5x5:354347',$,'5x5',#203,#198,'354347');
```

Logic Facts as Output of ITr

```
column(column205).
productdefinitionshape(productdefinitionshape198).
has_representation(column205,productdefinitionshape198).
shaperepresentation(shaperepresentation192).
has_representations(productdefinitionshape198,shaperepresentation192).
mappeditem(mappeditem190).
has_items(shaperepresentation192,mappeditem190).
representationmap(representationmap176).
has_mappingsource(mappeditem190,representationmap176).
shaperepresentation(shaperepresentation158)
has_mappedrepresentation(representationmap176,shaperepresentation158).
extrudedareasolid(extrudedareasolid148).
has_items(shaperepresentation158,extrudedareasolid148).
has_depth(extrudedareasolid148,11.0457511378774).
```

Figure 4. Sample of IFC transformation into logic facts.

3.1.3 *Logic facts refinement*

The last sub-step refines the raw logic facts converted from the previous sub-step (IFC to logic facts conversion), to make them suitable for the logic reasoning and compliant with the prolog syntax requirements. Two types of logic facts refinement include: 1) conversion of unit symbols to their corresponding unit name, and 2) addition of single quote symbol ('*string*') to enclose strings in logic fact arguments such as constant, numbers, and some IFC instances.

The unit of measurements in the logic facts are expressed in symbol forms, which are inherited from the original BIM authoring software. For instance, the unit for a window dimensions, are commonly expressed as 24" x 48" in the original BIM software. Accordingly, the predicate *has_objecttype(window1724,24"_x_48")* also contains the symbol for inch (") in its second argument. By refining it, the unit symbols are changed to its equivalent unit name. For example, the window dimensions *24"_x_48"* is changed to *24_inches_x_48_inches*.

Arguments that start with a number followed by strings are not compliant with the prolog syntax requirements. In this case, single quote is used to enclose the string arguments ('*string*') of the logic facts to make the logic facts compliant with the prolog syntax. For instance, without string arguments enclosure, the second argument of the predicate *has_globalid(window1724,2pbnvhtcp1kavcyplsrmwh)*, which is an instance of the global ID for

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

the *window1724*, yielded a syntax error because the argument starts with a numeric value which is not allowed in prolog syntax. By enclosing the arguments with single quote, prolog treats the argument as a string.

To this end, a refinement algorithm was developed using the regular expression (re) module in Python (version 3.7) to refine the logic facts information (Figure 5).

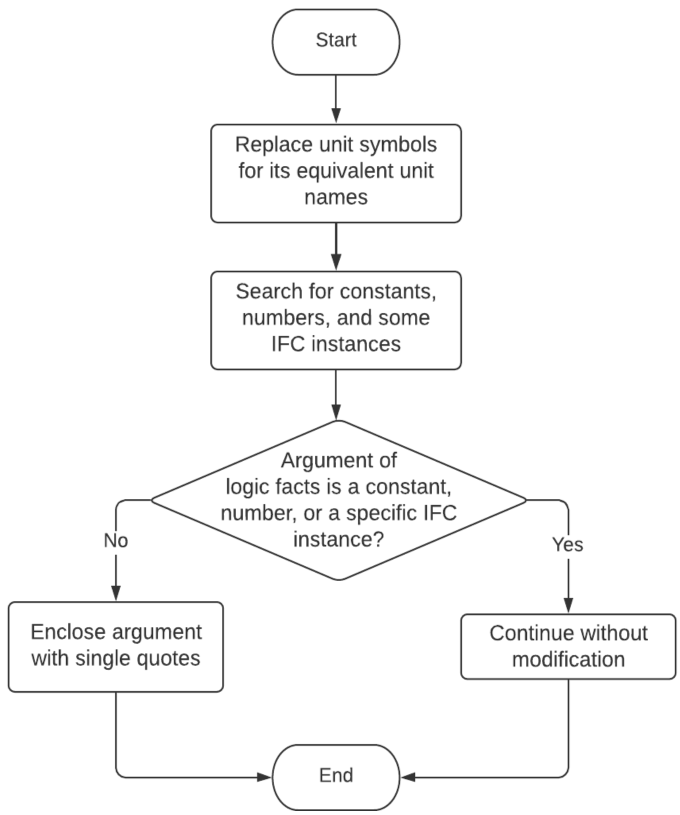


Figure 5. Refinement algorithm flow chart.

3.2 Logic Rules Development

For the development of the logic rules, a knowledge engineering approach is applied. Knowledge engineering is concerned with "data and information representation and encoding methodologies" to meet the needs of the user. Particularly, two heuristics from knowledge engineering were applied: 1) heuristic of specific situations and 2) heuristic of situation comparison [76]. The first heuristic uses specific cases to acquire the target knowledge. Relating to the logic rules development, specific predicates corresponding to building components (e.g., wall and floor) and building elements (e.g., column and beam) were used to develop the extraction and inference rules. For example, the predicate *has_representation(column2118,productdefinitionshape2112)*, which translates to column2118

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

has representation productdefinitonshape2112, is transformed to the general form *has_representation(Column,Productdefinitionshape)* by replacing the specific arguments with variables, that will serve as a template to capture the predicate *has_representation* for any columns and *productdefinitionshapes* instances of the logic facts. The second heuristic applies when comparing predicates that are shared by different building components. For instance, IfcSlab can be used to model roofs and floors. Therefore, to clarify if the predicate *slab(Slab)*, is a roof or slab, additional predicates such as *has_predefinedtype(Slab,floor)* and *has_predefinedtype(Slab,roof)* allow its correct classification, respectively.

Logic rules are developed to allow information extraction and filtering, as well as inference of new information from the building design logic facts. Logic rules in this paper can be divided into three types, based on the purpose (Figure 6): information extraction (IE) type (type I), information inference type (type II), and information filtering type (type III). The type I logic rules are used to extract information such as quantities, dimensions, and materials of the building components/elements (i.e., wall, roof, column) from the building design. Therefore, to capture the various IFC entities representation in a BIM model [i.e., building components/elements (e.g., wall, column), geometric representation (e.g., Brep, SweptSolid), and cross-sectional profile definition (e.g., rectangular, circular)], the corresponding logic rules need to be created to account for such combinations. The type II logic rules are used to infer physical properties (e.g., area, volume, and weight) from the building components/elements. These properties are derived from information extracted using type I logic rules and a supporting material module. The supporting material module contains material and density information for wood materials, which are encoded as logic facts. Lastly, the type III logic rules are used to filter out building components/elements of other materials (e.g., concrete and steel) from those of wood materials, and filter out unrelated object types such as recess and corner board.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

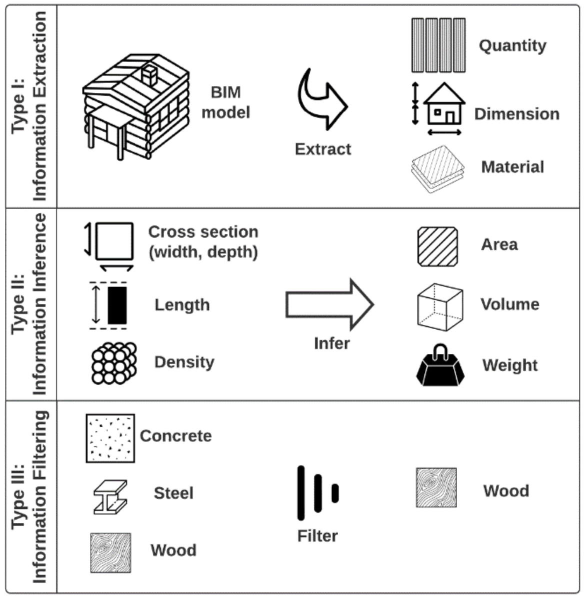


Figure 6. Type of logic rules with some application examples.

The development process for logic rules consists of four sub-steps: 1) relevant logic facts identification, 2) terms replacement, 3) logic facts connection, and 4) SOL application. In the first sub-step, the relevant logic facts are identified and serve as (a) constraints in the unification process with only the relevant logic facts. For example, the set of relevant logic clauses [i.e., *relassociatesmaterial(relassociatesmaterial4670* to *has_name(material265,'softwood,_lumber')]* as shown in Figure 7, are used to extract the material information for column instances; and (b) sub goals that need to be met for the logic rules to succeed. In the terms replacement sub-step, constants and numbers are replaced by variables (first letter upper-cased). The third sub-step consists of joining the logic facts using the logical conjunction (,). Finally, in the last sub-step, the all-solution predicates (i.e., findall, bagof, and setof) are applied to extend the rule for all building component/element instances. To illustrate the process, the development of a logic rule to extract the material, and to infer the volume and weight for columns with swept solid geometric representation and rectangular profile definition are summarized in Figure 7 and Figure 8, respectively. Similarly, different logic rules are developed for columns with other types of geometric representations (e.g., B-rep) and profile definitions.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

Sub-step 1

```
relassociatesmaterial(relassociatesmaterial4670).
material(material265).
has_relatingmaterial(relassociatesmaterial4670,material265).
column(column2153).
has_relatedobjects(relassociatesmaterial4670,column2153).
has_name(material265,'softwood,_lumber').
```

Sub-step 2

```
relassociatesmaterial(Relassociatesmaterial).
material(Material).
has_relatingmaterial(Relassociatesmaterial,Material).
column(Column).
has_relatedobjects(Relassociatesmaterial,Column).
has_name(Material,Materialname).
```

Sub-steps 3-4

```
rule3_column_material:-
    L=[H|T],
    findall((Column,Materialname),
    (relassociatesmaterial(Relassociatesmaterial),
    material(Material),
    has_relatingmaterial(Relassociatesmaterial,Material),
    column(Column),
    has_relatedobjects(Relassociatesmaterial,Column),
    has_name(Material,Materialname)),L),
    foreach((Column,Materialname) in L,
    asserta(has_material_list(Column,Materialname))).
```

Figure 7. Logic rule development for extracting the material information for columns.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Figure 8. Type II logic rule sample: a) material module, and rule to derive b) volume and c) weight for columns.

3.3 Logic Reasoning Execution

In addition to the logic facts and logic rules, supporting modules were developed to provide reasoning support: a) unit conversion, which are logic clauses that convert any length unit to feet unit length; and b) material properties, which contains the material type and density information. Once, all the components and information are ready, the logic reasoning is performed in an automated way. First, the logic facts are loaded into the logic reasoner. Then, the logic rules are executed internally along with the supporting modules by the logic reasoner. The reasoner applies the unification and substitution functions to prove the goals defined in the logic rules. Finally, the results are returned in terms of logic rules' success or failure.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

3.4 Result and Error Evaluation

The performances of the extraction algorithms are evaluated using the precision, recall, and F1-measure. The precision performance metric measures how good the extraction result is and it is defined as the ratio of the number of correctly extracted logic fact instances in the total number of logic fact instances extracted (Equation 1). Correspondently, recall is a measurement of the degree of coverage in the target information and the definition is the ratio of the number of correctly extracted logic fact instances in the total number of existing logic fact instances in the source file (Equation 2). The use of precision or recall alone does not provide a full-picture evaluation, therefore a third performance metric call F1-measure is used to provide an overall performance based on the precision and recall. The F1-measure is defined as the harmonic mean between the precision and recall (Equation 3).

$$Precision\ (P) = \frac{Correctly\ extracted\ LF}{Total\ LF\ extracted} \qquad \text{Equation 1}$$

$$Recall\ (R) = \frac{Correctly\ extracted\ LF}{Total\ exiting\ LF} \qquad \text{Equation 2}$$

$$F1\ measure = \frac{2PR}{P + R} \qquad \text{Equation 3}$$

3.5 Logic Facts Refinement

This step applies when additional logic predicates that are not suitable for the logic reasoning are detected and were not captured in the logic facts refinement sub-step of Step 1. Those predicates make the related logic rules yield unexpected results. Therefore, the refinement algorithm needs to be extended to modify those predicates, making them suitable for the logic reasoning. For example, the numbering of the predicate *has_segments#(Term1, Term2)*, makes the logic reasoning ineffective because each numbered predicate (i.e., *has_segments1*,..,*has_segments#*) are treated as independent predicates and therefore unable to unify all the instances related to that predicate.

3.6 Logic Rules Extension

At the beginning, the proposed method contains only the set of logic rules created based on the development BIM model. For example, the proposed method can be used to analyze columns with swept solid geometric representation (used in the development model) and not for columns with geometric representations that were not present in the development model (e.g., B-rep). To improve the performance, supplementary logic rules are gradually added to the existing rule set for those cases in which the current logic rules fail to capture. Once the newly added logic rules are incorporated to the set of rules, the method will be able to analyze effectively the previously incorrectly identified geometric representations, as well as materials information. This process improves the performance and overall effectiveness of the proposed method by iteratively and accumulatively increasing the number of logic rules in the set.

4. Experimental Implementation and Validation

In this section, the experimental implementation of the proposed method along with the results are presented. Five BIM models were used for this purpose, one for the development and four for the validation. Additionally, the time efficiency of the automated analysis was also measured.

4.1 Implementation Software

The prolog programming language implementation used for the proposed method is B-Prolog. B-Prolog uses HC representation and it was selected because of its inherent reasoning capabilities, compatibility with C and Java programming languages, and is based on classic Prolog [55].

4.2 Test Cases

The five BIM models used for the experiments are described here. The proposed method was initially implemented using the development model (Dev), a wood frame structure modeled in Autodesk Revit software (version 2019) as shown in Figure 9. This Dev model serves as a gold standard.

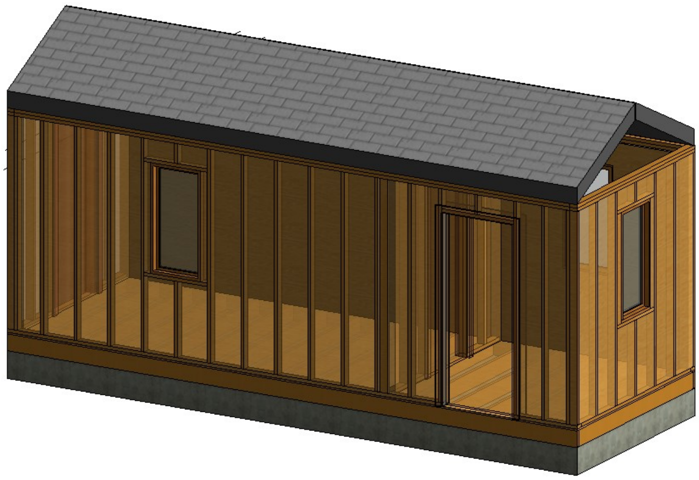


Figure 9. BIM Dev model for algorithm development.

For the testing and validation, four additional BIM models, namely T1, T2, T3, and T4, were used (Figure 10) to: (1) evaluate the robustness of the proposed method, and (2) to test the effectiveness of the developed logic rules on unseen models. The first three models (T1-T3) consisted of wood residential projects and T4 is a three stories commercial wood building project. The four BIM models are created in Autodesk Revit to a LOD 300 (T1-T3) and LOD 400 BIM (T4), respectively.

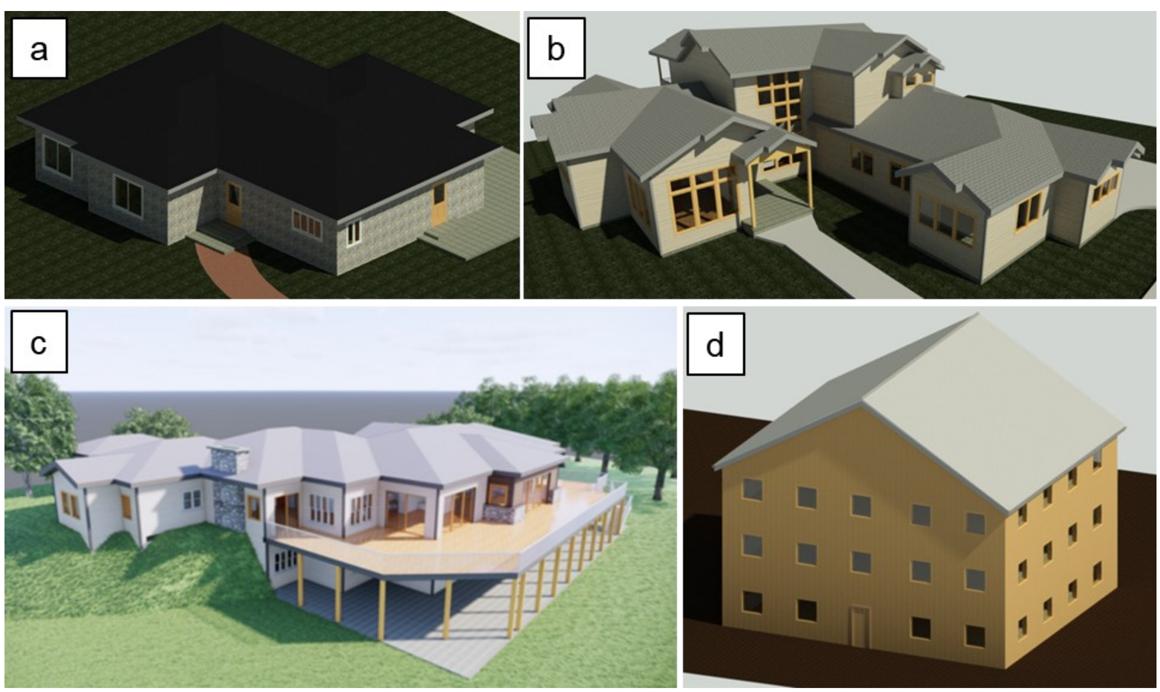


Figure 10. Test case models: a) T1, b) T2, c) T3, and d) T4.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

4.3 From BIM model to Logic Facts (Step 1)

First, each BIM model was exported from Revit to IFC format. Next, the IFC models were converted to logic facts using the ZE_BIM_FOL_Converter2.0 prototype as shown in Figure 11. The top right of Figure 11 shows some IFC instances from an IFC file; while the bottom right of Figure 11 shows corresponding logic facts successfully converted. Then the logic facts are preprocessed using the Python algorithm (Figure 5). Some sample logic facts before and after the preprocessing are shown in Figure 12.

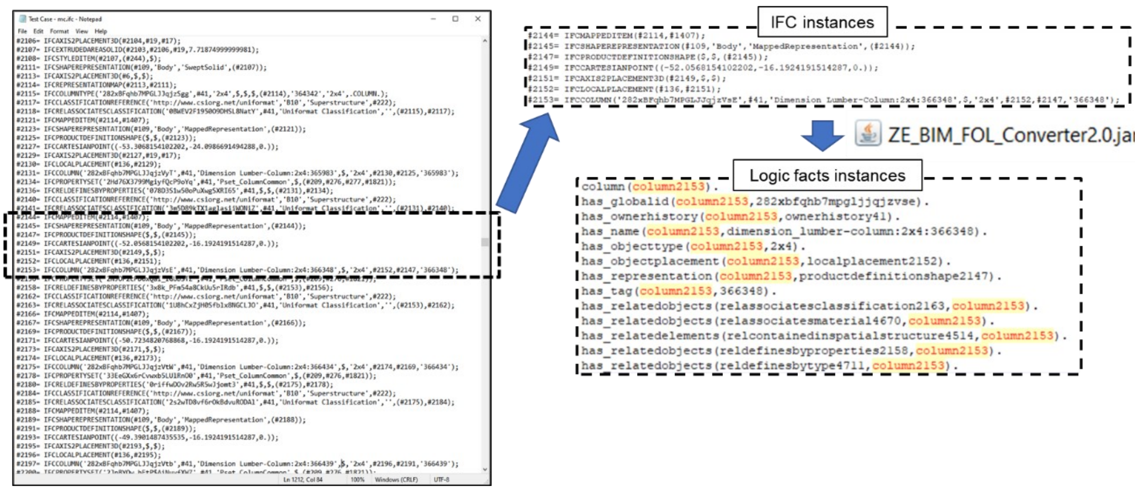


Figure 11. Sample conversion of IFC to logic facts.

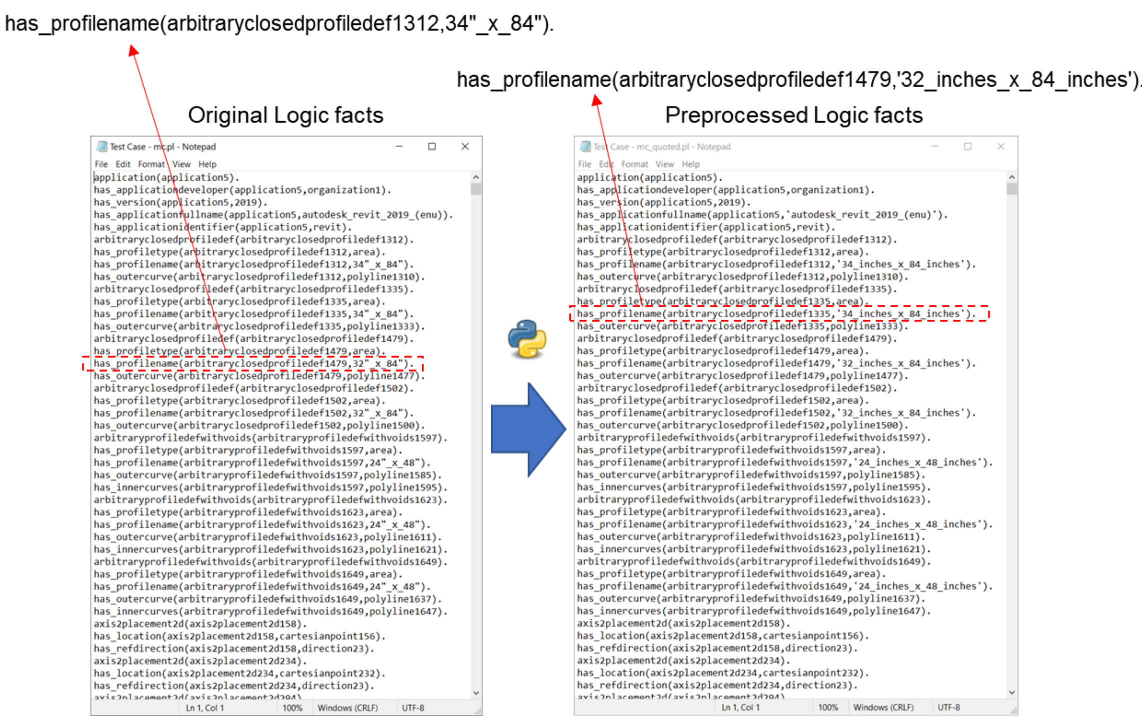


Figure 12. Sample of preprocessed logic facts.

4.4 Logic Rules Development and Logic Reasoning Execution (Step 2-3)

The logic rules algorithms to extract and infer information from BIM models were developed using a randomly selected subset from the Dev model logic facts. A total of 44 logic rules were developed initially, in which 26 correspond to extraction rules, 17 correspond to inference rules, and 1 filtering rule (Table 1). After the logic rules development, the logic rules were tested using the Dev model in the logic reasoning step. These steps were further applied to the four test models (T1-T4) in a similar way.

Table 1. List of rules developed using the Dev model.

Rule	Rule Type*	Component /element	Source IFC Object	Geometric Representation	Profile Definition	Target information
1	I	Interior Wall	IfcWallStandardCase	Swept solid	Rectangle	Dimensions
2	I					Quantity
3	I					Material
4	I	Opening (Int. Wall)	IfcOpeningElement			Quantity
5	I			Swept solid	Rectangle	Dimensions
6	II					Area
7	II					Total Area
8	II	Interior Wall	IfcWallStandardCase			Area
9	II					Net area
10	I	Exterior Wall		Swept solid	Rectangle	Dimensions
11	I					Quantity
12	I					Material
13	I	Opening (Ext. Wall)	IfcOpeningElement			Quantity
14	I			Swept solid	Rectangle	Dimensions
15	II					Area
16	II					Total Area
17	II	Exterior Wall	IfcWallStandardCase			Area
18	II					Net area
19	I	Floor	IfcSlab	Swept solid	Rectangle	Dimensions
20	I					Quantity
21	I					Material
22	I	Opening (Floor)	IfcOpeningElement			Quantity
23	I			Swept solid	Rectangle	Dimensions
24	II					Area
25	II					Total Area
26	II	Floor	IfcSlab			Area

Rule	Rule Type*	Component /element	Source IFC Object	Geometric Representation	Profile Definition	Target information
27	II					Net area
28	I	Roof		Swept solid	Rectangle	Dimensions
29	I					Quantity
30	I					Material
31	II					Area
32	I	Column	IfcColumn	Swept solid	Rectangle	Dimensions
33	I					Quantity
34	I					Material
35	II					Volume
36	II					Weight
37	I	Beam	IfcBeam	Swept solid	Rectangle	Dimensions
38	I					Quantity
39	I					Material
40	II					Volume
41	II					Weight
42	I	Opening	IfcOpeningElement			Quantity
43	I			Swept solid	Rectangle	Dimensions
44	III	Floor	IfcSlab			Material

*Type I: extraction; Type II: inference; Type III: filtering

4.5 Result and Error Evaluation (Step 4)

In this section, the experimental results for the testing of the five models based on the performance metrics (precision, recall, and F1 measure) were presented in Step 4. First, the detailed results for the development model Dev are shown in Table 2. Following, the detailed result for the validation tests of the proposed method using the four BIM models (T1-T4) are presented in Table 3 through Table 6, respectively. Each table contains the building component/element (column 1) and the target information (column 2) defined. Column 3 to 5 show the total number of logic clauses present in each converted model, the number of correctly extracted logic clauses, and the number of extracted logic clauses, respectively. The last three columns present the precision P, recall R, and F1 measure, respectively.

Table 2. Results of the Dev model (gold standard).

Component/ Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted
Ext wall	quantity	4	4	4

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

Component/ Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted
Ext wall	dimensions (length, height, width)	12	12	12
Ext wall	material	12	12	12
Ext wall	opening quantity	5	5	5
Ext wall	opening dimensions (length, height, width)	15	15	15
Int wall	quantity	1	1	1
Int wall	dimensions (length, height, width)	3	3	3
Int wall	material	3	3	3
Int wall	opening quantity	1	1	1
Int wall	opening dimensions (length, height, width)	3	3	3
Floor	quantity	1	1	1
Floor	dimensions (length, height, width)	3	3	3
Floor	material	1	1	1
Floor	opening quantity	-	-	-
Floor	opening dimensions (length, height, width)	-	-	-
Roof	quantity	2	2	2
Roof	dimensions (length, height, width)	6	6	6
Roof	material	6	6	6
Roof	opening quantity	-	-	-
Roof	opening dimensions (length, height, width)	-	-	-
Column	quantity	60	60	60
Column	dimensions (length, height, width)	180	180	180
Column	material	60	60	60
Beam	quantity	50	50	50
Beam	dimensions (length, height, width)	150	150	150
Beam	material	50	50	50
Total/Average		**628**	**628**	**628**

Table 3. Result for the model T1.

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Ext wall	quantity	12	12	30	40	100	57.1
Ext wall	dimensions (length, height, width)	36	36	90	40	100	57.1
Ext wall	material	60	60	78	76.9	100	87
Ext wall	opening quantity	14	14	14	100	100	100
Ext wall	opening dimensions (length, height, width)	42	42	42	100	100	100
Int wall	quantity	30	30	30	100	100	100
Int wall	dimensions (length, height, width)	90	81	81	100	90	94.7
Int wall	material	90	90	90	100	100	100
Int wall	opening quantity	17	17	17	100	100	100
Int wall	opening dimensions (length, height, width)	51	45	45	100	88.2	93.8
Floor	quantity	1	1	1	100	100	100
Floor	dimensions (length, height, width)	3	0	0	0	0	0
Floor	material	3	3	4	75	100	85.7
Floor	opening quantity	-	-	-	-	-	-
Floor	opening dimensions (length, height, width)	-	-	-	-	-	-
Roof	quantity	11	11	11	100	100	100
Roof	dimensions (length, height, width)	33	0	0	0	0	0
Roof	material	33	33	33	100	100	100
Roof	opening quantity	-	-	-	-	-	-
Roof	opening dimensions (length, height, width)	-	-	-	-	-	-
Column	quantity	-	-	-	-	-	-
Column	dimensions (length, height, width)	-	-	-	-	-	-
Column	material	-	-	-	-	-	-
Beam	quantity	-	-	-	-	-	-
Beam	dimensions (length, height, width)	-	-	-	-	-	-
Beam	material	-	-	-	-	-	-
	Total/Average	**526**	**475**	**566**	**83.9**	**90.3**	**87.0**

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

Table 4. Result for the model T2.

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Ext wall	quantity	69	69	69	100	100	100
Ext wall	dimensions (length, height, width)	207	99	99	100	47.8	64.7
Ext wall	material	276	276	276	100	100	100
Ext wall	opening quantity	90	90	90	100	100	100
Ext wall	opening dimensions (length, height, width)	270	264	264	100	97.8	98.9
Int wall	quantity	63	62	62	100	98.4	99.2
Int wall	dimensions (length, height, width)	189	183	183	100	96.8	98.4
Int wall	material	189	186	186	100	98.4	99.2
Int wall	opening quantity	30	30	30	100	100	100
Int wall	opening dimensions (length, height, width)	90	90	90	100	100	100
Floor	quantity	4	4	4	100	100	100
Floor	dimensions (length, height, width)	12	12	12	100	100	100
Floor	material	12	12	12	100	100	100
Floor	opening quantity	-	-	-	-	-	-
Floor	opening dimensions (length, height, width)	-	-	-	-	-	-
Roof	quantity	16	9	9	100	56.3	72
Roof	dimensions (length, height, width)	48	27	27	100	56.3	72
Roof	material	34	27	27	100	79.4	88.5
Roof	opening quantity	-	-	-	-	-	-
Roof	opening dimensions (length, height, width)	-	-	-	-	-	-
Column	quantity	8	8	8	100	100	100
Column	dimensions (length, height, width)	24	24	24	100	100	100
Column	material	8	8	8	100	100	100
Beam	quantity	9	9	9	100	100	100
Beam	dimensions (length, height, width)	27	27	27	100	100	100
Beam	material	9	9	9	100	100	100

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Total/Average		**1684**	**1525**	**1525**	**100**	**90.6**	**95.0**

Table 5. Result for the model T3.

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Ext wall	quantity	97	97	97	100	100	100
Ext wall	dimensions (length, height, width)	291	291	291	100	100	100
Ext wall	material	427	427	427	100	100	100
Ext wall	opening quantity	58	58	58	100	100	100
Ext wall	opening dimensions (length, height, width)	174	174	174	100	100	100
Int wall	quantity	115	115	149	77.2	100	87.1
Int wall	dimensions (length, height, width)	345	336	336	100	97.4	98.7
Int wall	material	345	345	345	100	100	100
Int wall	opening quantity	33	33	33	100	100	100
Int wall	opening dimensions (length, height, width)	99	99	99	100	100	100
Floor	quantity	12	12	12	100	100	100
Floor	dimensions (length, height, width)	36	36	36	100	100	100
Floor	material	41	41	41	100	100	100
Floor	opening quantity	-	-	-	-	-	-
Floor	opening dimensions (length, height, width)	-	-	-	-	-	-
Roof	quantity	4	4	4	100	100	100
Roof	dimensions (length, height, width)	12	12	12	100	100	100
Roof	material	4	4	4	100	100	100
Roof	opening quantity	-	-	-	-	-	-
Roof	opening dimensions (length, height, width)	-	-	-	-	-	-
Column	quantity	17	17	26	65.4	100	79.1
Column	dimensions (length, height, width)	51	51	57	89.5	100	94.4
Column	material	17	17	26	65.4	100	79.1
Beam	quantity	-	-	-	-	-	-

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Beam	dimensions (length, height, width)	-	-	-	-	-	-
Beam	material	-	-	-	-	-	-
Total/Average		**2178**	**2169**	**2227**	**97.4**	**99.6**	**98.5**

533

534 Table 6. Result for the model T4.

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Ext wall	quantity	12	12	18	66.7	100	80
Ext wall	dimensions (length, height, width)	36	36	54	66.7	100	80
Ext wall	material	36	36	42	85.7	100	92.3
Ext wall	opening quantity	39	39	609	6.4	100	12.0
Ext wall	opening dimensions (length, height, width)	117	117	2295	5.1	100	9.7
Int wall	quantity	12	12	12	100	100	100
Int wall	dimensions (length, height, width)	36	36	36	100	100	100
Int wall	material	60	60	60	100	100	100
Int wall	opening quantity	41	41	41	100	100	100
Int wall	opening dimensions (length, height, width)	123	123	123	100	100	100
Floor	quantity	3	3	8	37.5	100	54.5
Floor	dimensions (length, height, width)	9	9	24	37.5	100	54.5
Floor	material	3	3	8	37.5	100	54.5
Floor	opening quantity	-	-	-	-	-	-
Floor	opening dimensions (length, height, width)	-	-	-	-	-	-
Roof	quantity	2	2	2	100	100	100
Roof	dimensions (length, height, width)	6	6	6	100	100	100
Roof	material	4	4	4	100	100	100
Roof	opening quantity	-	-	-	-	-	-
Roof	opening dimensions (length, height, width)	-	-	-	-	-	-
Column	quantity	538	538	538	100	100	100

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

Component /Element	Item	No. of relevant LC	No. of LC correctly extracted	No. of LC extracted	P (%)	R (%)	F (%)
Column	dimensions (length, height, width)	1614	1614	1614	100	100	100
Column	material	538	538	538	100	100	100
Beam	quantity	490	490	607	80.7	100	89.3
Beam	dimensions (length, height, width)	1470	1005	1356	74.1	68.4	71.1
Beam	material	490	490	607	80.7	100	89.3
Total/Average		**5679**	**5214**	**8602**	**60.6**	**91.8**	**73.0**

A summary of the performance result in terms of precision, recall, and F1-measure for the development model and the four test cases is shown in Table 7. The output is a report in a comma-separated values file (.csv) that contains the result of the analysis.

Table 7. Summary of the testing results.

#	Model	P (%)	R (%)	F1 (%)
1	Dev*	100	100	100
2	T1	83.92	90.30	87.00
3	T2	100.00	90.56	95.05
4	T3	97.40	99.59	98.48
5	T4	60.61	91.81	73.02

*Development model

4.6 Performance Improvement: Algorithm Refinement and Logic Rules Extension (Step 5-6)

After the evaluation step of each test model, the refinement algorithm and logic rules were extended to improve the reasoning performance. For the refinement algorithm, two specific predicates *has_cfsfaces#* and *has_segments#* used in the B-rep and swept solid geometric representations, respectively, were modified to facilitate the implementation of the inherent unification and SOL functions of prolog. This is done by extending the refinement algorithm in Step 5, which eliminates the numbering sequence of the predicates (*has_cfsfaces* and *has_segments*). Then, in Step 6, 28 new logic rules (18 extraction rules and 10 filtering rules) were added to the initial set of logic rules, to improve the reasoning capability of the ruleset (Table 8). Examples of geometric representation and profile definition variations encountered for walls, roofs, and floors information, include B-rep and arbitrary closed profile, respectively.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021).
https://www.sciencedirect.com/science/article/pii/S0926580521002077

After application of Step 5 and 6, the proposed method was able to extract and infer all the information from the BIM models.

Table 8. List of new rules developed from the testing models (T1-T4).

Rules	Rule Type*	Component /element	Source IFC Object	Geometric Representation	Profile Definition	Target information
1	I	Interior Wall	IfcWallStandardCase	sweptsolid	arbitrary closed	Dimensions
2	I			clipping	Rectangle	
3	I			clipping	arbitrary closed	
4	I		IfcWall	clipping	Rectangle	
5	I					Material
6	I	Exterior Wall	IfcWallStandardCase	clipping	Rectangle	Dimensions
7	I			clipping	arbitrary closed	
8	III					Material
9	III					
10	III	Exterior Wall	IfcWall			Object type
11	III	Opening	IfcOpeningElement			
12	I	Floor	IfcSlab	sweptsolid	arbitrary closed	Dimensions
13	III					Material
14	III					
15	III					
16	I	Roof	IfcRoof	Facetedbrep		Dimensions
17	I		IfcSlab	sweptsolid	arbitrary closed	
18	I		IfcSlab			Material
19	I	Opening	IfcOpeningElement	sweptsolid	arbitrary closed	Dimensions
20	I			Facetedbrep		
21	III	Column	IfcColumn			Material
22	III					
23	I	Beam	IfcBeam	sweptsolid	Rectangle (assembly)	Dimensions
24	I			sweptsolid	arbitrary closed	
25	I			clipping	arbitrary closed	
26	I			clipping	Rectangle	
27	I			Facetedbrep		
28	III					Material

*Type I: extraction; Type II: inference; Type III: filtering

4.7 Time Performance Testing

A time performance testing of the proposed method implementation was empirically performed. The results show that the processing time increases with the total number of logic

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

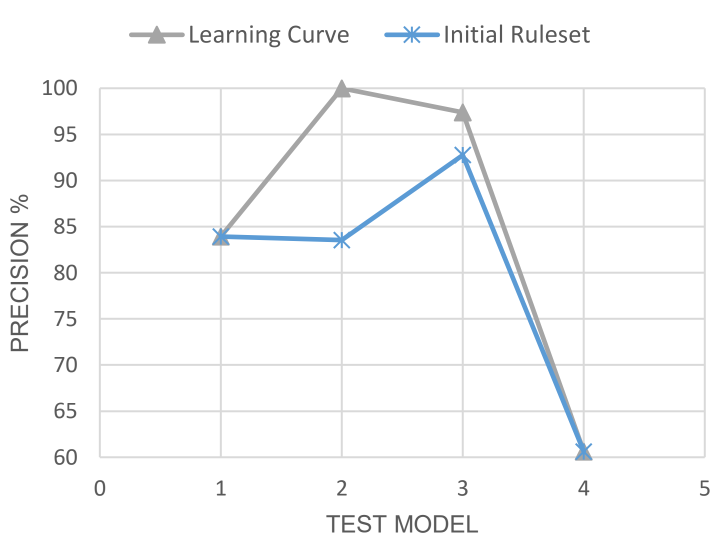
clauses and the number of relevant logic clauses (Table 9). The total number of logic clauses correspond to the loading time of the logic facts into the prolog system and the number of relevant logic clauses are related to the searching and matching of the logic clauses according to the relationship from the logic rules. The longest analysis time corresponds to the T4 model and it took around a minute to complete, despite having to load 524,561 logic clauses and to analyze 5,679 relevant clauses. The experiments were conducted using a laptop with a random access memory (RAM) of 15.7 gigabytes and an Intel(R) Core(TM) i7-9750H processor with 2.60 gigahertz (GHz) of central processing unit (CPU) speed.

Table 9. Time performance results.

Model	Total Number of Logic Clause	Number of Relevant Logic Clause	Time (s)
Dev	11,647	628	0.11
T1	62,877	526	0.82
T2	212,787	1,684	6.49
T3	548,053	2,178	39.88
T4	524,561	5,679	78.24

4.8 Discussion

A closer comparison on the performances between the implementation of the method using a learning curve and the implementation with only the initially developed logic rules are shown in Figure 13, Figure 14, and Figure 15. The results showed an overall increase in precision, recall, and F1 measure for the test models (T1-T4), except for the model T4 because it contained 2,295 recesses that are mapped as openings in the IFC model, causing the overall performance to drop.



This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Figure 13. Learning curve for precision.

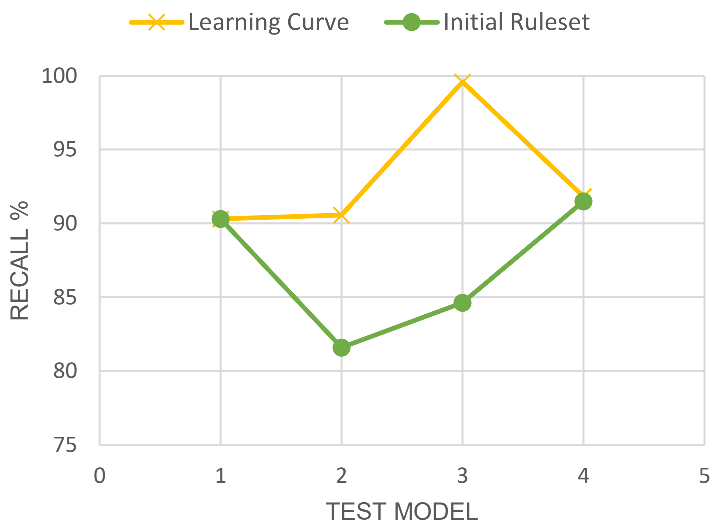


Figure 14. Learning curve for recall.

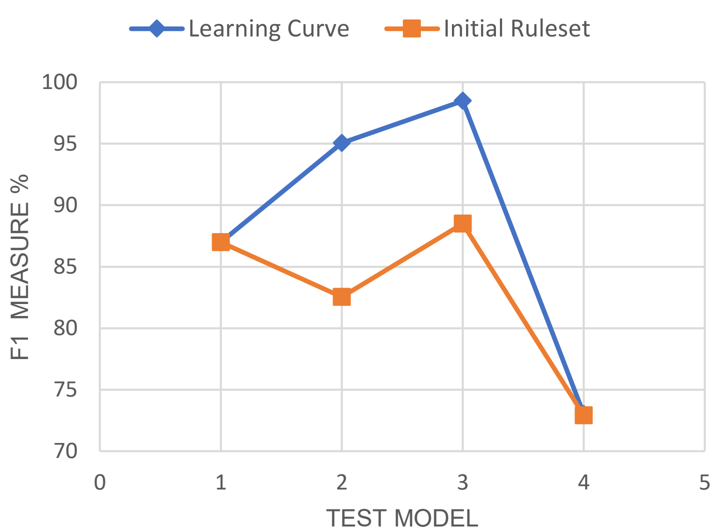


Figure 15. Learning curve for F1 measure.

Besides the information extraction and inference, the algorithm was also able to classify all components' properties and relations correctly. Properties and relations include the function of walls (i.e., internal or external) and slabs (i.e., roof or floor), material properties set, relationship between walls and openings, and others. Lastly, although the derived information is presented in a general file format (.csv), which can be used for quantity takeoff and fabrication purposes (e.g., cutting), it can be easily extended to other file formats (e.g., .nc for CNC code and .py for robotic controllers) based on the requirements of the downstream applications.

5. Contributions

The proposed method contributes to the body of knowledge in four main ways. First, it provides a promising way, in terms of recall, precision, and time efficiency, to analyze BIM models and infer construction level information to facilitate wood construction automation, by providing logic representation and reasoning based on powerful but underexplored FOL and SOL. Furthermore, the generated information (e.g., length of the wood elements) can be further transferred as input to downstream applications (e.g., CNC or robots) to perform operations (e.g., cutting) in a continuous and digital workflow. Second, this research confirmed the robustness of logic representation and reasoning to represent and reason about IFC schema information in the AEC domain, such a new application area is demonstrated in addition to the existing automated code compliance checking research [69]. Third, this research leveraged SOL for efficient information extraction from IFC-based BIM models by extending the binary nature of the FOL (i.e., satisfy or fail) into set quantification (i.e., all instances of a query). Fourth, unlike existing BIM Application Programming Interface (APIs) that are integrated in proprietary software to achieve data extractions from the software's native formats, the proposed approach utilizes IFC as the BIM input data representation, which contributes to the standardization and interoperability of information exchange in the AEC domain.

6. Conclusions, Limitations, and Future Direction

The objective of this research was to analyze building design information using logic representation and reasoning and to infer construction level information for supporting wood construction automation during preconstruction. To accomplish this, a novel data-driven method was proposed that utilizes FOL and SOL approaches to reason about logic enabled BIM models. More specifically, the proposed method can be used for automated information extraction (e.g., quantities and dimensions) and information inference (e.g., area and weight) from the logic enabled IFC representation of building objects. The proposed method was successfully implemented using B-Prolog and a development BIM model. In addition, the proposed method was tested and validated in four unseen BIM models. The experimental results have shown that the proposed method can achieve high performance in precision, recall, and F1 measure. These experiments confirmed that the iterative nature of the proposed method serves to continuously improve the results until a satisfactory performance is achieved. By adding more supplementary logic rules to the current set of rules, the analysis capabilities of the proposed method become more robust every time rules are created and added based on previously unseen cases of IFC object instances. The research has also empirically shown that the use of logic-enabled algorithm is time efficient.

The impact of this work in the AEC domain could be far-reaching. First, the proposed method opens the door to greater automation in decision making and computational tasks in the AEC domain by providing a ready-for-reasoning representation of building information. Second, the application of this work could be extended to support automation in the construction domain in general such as in robotic automation, lean construction, or lifecycle analysis, among others, by customizing reasoning rules. Third, the proposed method builds on the open standard IFC and logic representation, which favor the interoperability and human readability (as long as the predicate names are meaningful), respectively.

One limitation of this study is the use of bounding box to determine the dimensions of irregular shaped BIM objects. In many cases, wood in construction such as studs are parallelepiped and the dimensions of the bounding box matched with the real ones, however, in some cases, the use of the bounding box dimensions can be inaccurate, especially for irregular shape elements. This limitation becomes important when computing the volume and weight of irregular shape elements because the magnitude of the error increases with the degree of irregularity. Another limitation is that the quality and quantity of the extracted information depend on the level of development of the BIM model. For example, in a LOD 300 BIM model, the columns (studs) information would not be modeled, therefore the method would not be able to capture that information. Lastly, despite that the proposed method is applicable to any level of prefabrications of offsite construction, the test cases focused on were wood construction mainly. Despite these limitations, the proposed method has shown to be promising and reliable for automated BIM model information analysis.

Further work is needed to improve the comprehensiveness of the proposed implementation for irregular shape objects. Additionally, the logic-enabled BIM model information is lacking in terms of the semantic for offsite construction and the level of details of BIM models. Therefore, to leverage these two aspects, in future work, the authors will propose a knowledge model that would further help with the analysis of wood construction. In terms of the level of details, one direction is to analyze information from BIM models with LOD 400, which provide a richer content information for fabrication and/or construction phase. Another alternative is to complement the current method with shop drawings and/or specifications information. This way will provide additional information that are missing from LOD 300-350 BIM models.

ACKNOWLEDGMENTS

The authors would like to thank the National Science Foundation (NSF). This material is based on work supported by the NSF under Grant No. 1827733. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the NSF. Also, the authors would like to thank Professor Clark Cory and Temitope Akanbi for sharing the BIM models (T1-T3) and T4, respectively, that contributed in the experimental section of this research. The first author was supported by an Education Support Leave from the Technological University of Panama.

7. Reference

[1] D. Castro-Lacouture, Construction automation, in: Springer Handb. Autom., Springer, 2009: pp. 1063–1078 ISBN: 978-3-540-78831-7.

[2] C.Z. Li, R.Y. Zhong, F. Xue, G. Xu, K. Chen, G.G. Huang, G.Q. Shen, Integrating RFID and BIM technologies for mitigating risks and improving schedule performance of prefabricated house construction, Journal of Cleaner Production. 165 (2017) pp. 1048–1062. https://doi.org/10.1016/j.jclepro.2017.07.156.

[3] F. Craveiro, J.P. Duarte, H. Bartolo, P.J. Bartolo, Additive manufacturing as an enabling technology for digital construction: A perspective on Construction 4.0, Automation in Construction. 103 (2019) pp. 251–267. https://doi.org/10.1016/j.autcon.2019.03.011.

[4] Associated General Contractors, Eighty percent of contractors report difficulty finding qualified craft workers to hire as firms give low marks to quality of new worker pipeline, (2019). https://www.agc.org/news/2019/08/27/eighty-percent-contractors-report-difficulty-finding-qualified-craft-workers-hire-0 (accessed February 3, 2021).

[5] U.S. Bureau of Labor Statistics, Job Openings and Labor Turnover, Economic News Release. (2020). https://www.bls.gov/news.release/jolts.a.htm (accessed August 8, 2020).

[6] Construction Industry Resources LLC, Construction Productivity in an Imbalanced Labor Market, 2016. www.myCLMA.com.

[7] McGraw-Hill Construction, The business value of BIM for construction in major global markets: How contractors around the world are driving innovation with Building Information Modeling, Bedford, MA, 2014. http://static.autodesk.net/dc/content/dam/autodesk/www/solutions/building-information-modeling/construction/business-value-of-bim-for-construction-in-global-markets.pdf.

[8] S.H. Ghaffar, J. Corker, M. Fan, Additive manufacturing technology and its implementation in construction as an eco-innovative solution, Automation in Construction. 93 (2018) pp. 1–11. https://doi.org/10.1016/j.autcon.2018.05.005.

[9] M. Pan, T. Linner, W. Pan, H. Cheng, T. Bock, A framework of indicators for assessing construction automation and robotics in the sustainability context, Journal of Cleaner Production. 182 (2018) pp. 82–95. https://doi.org/10.1016/j.jclepro.2018.02.053.

[10] R.M. Lawson, R.G. Ogden, Sustainability and Process Benefits of Modular Construction, in: 8th CIB World Build. Congr., Salford, United Kingdom, 2010: pp. 38–51. http://www.irbnet.de/daten/iconda/CIB18783.pdf.

[11] McGraw-Hill Construction, Prefabrication and Modularization, McGraw Hill Construction, Bedford, 2011.

[12] KPMG, Smart Construction: How offsite manufacturing can transform our industry, (2016). kpmg.com/uk.

[13] A. Alwisy, S. Bu Hamdan, B. Barkokebas, A. Bouferguene, M. Al-Hussein, A BIM-based automation of design and drafting for manufacturing of wood panels for modular residential buildings, International Journal of Construction Management. 19 (2019) pp. 187–205. https://doi.org/10.1080/15623599.2017.1411458.

[14] R. Sacks, C. Eastman, G. Lee, P. Teicholz, BIM handbook: a guide to building information modeling for owners, managers, designers, engineers and contractors, 3rd ed., Hoboken, New Jersey Wiley, 2018. https://doi.org/10.1016/S0926-5805(02)00090-0.

[15] S. Meschini, K. Iturralde, T. Linner, T. Bock, Novel applications offered by integration of robotic tools in BIM-based design workflow for automation in construction processes, CIB* IAARC W119 CIC 2016 Workshop. (2016).

[16] C. Goodier, A. Gibb, Future opportunities for offsite in the UK, Construction Management and Economics. 25 (2007) pp. 585–595. https://doi.org/10.1080/01446190601071821.

[17] A. Gibb, Standardization and pre-assembly - distinguishing myth from reality using case study research, Construction Management and Economics. 19 (2001) pp. 307–315. https://doi.org/10.1080/01446190010020435.

[18] N. Lu, T. Korman, Implementation of Building Information Modeling (BIM) in Modular Construction: Benefits and Challenges, in: Constr. Res. Congr. 2010, 2010: pp. 1136–1145.

[19] R.M. Lawson, R.G. Ogden, R. Bergin, Application of modular construction in high-rise buildings, Journal of Architectural Engineering. 18 (2012) pp. 148–154. https://doi.org/10.1061/(ASCE)AE.1943-5568.0000057.

[20] S.W. Nunnally, Construction Methods and Management, Seventh, 2007 ISBN: 0135000793.

[21] J. Nässén, F. Hedenus, S. Karlsson, J. Holmberg, Concrete vs. wood in buildings - An energy system approach, Building and Environment. 51 (2012) pp. 361–369. https://doi.org/10.1016/j.buildenv.2011.11.011.

[22] J. Glover, D.O. White, T.A.G. Langrish, Wood versus Concrete and Steel in House

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

Construction, Journal of Forestry. (2002).

[23] M. Chui, J. Mischke, The impact and opportunities of automation in construction, Voices on Infraestructure. (2019) pp. 7. https://www.mckinsey.com/~/media/McKinsey/Industries/Capital Projects and Infrastructure/Our Insights/The impact and opportunities of automation in construction/The-impact-and-opportunities-of-automation-in-construction.pdf?shouldIndex=false.

[24] J. Willmann, M. Knauss, T. Bonwetsch, A.A. Apolinarska, F. Gramazio, M. Kohler, Robotic timber construction - Expanding additive fabrication to new dimensions, Automation in Construction. 61 (2016) pp. 16–23. https://doi.org/10.1016/j.autcon.2015.09.011.

[25] H. Hasan, A. Reddy, A. Tsayjacobs, Robotic Fabrication of Nail Laminated Timber, Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC). (2019). https://doi.org/10.22260/isarc2019/0162.

[26] S. Asbjørn, A. Oded, E. Phillip, P. Luka, S. Florin, G. Fabio, K. Matthias, Topology Optimization and Robotic Fabrication of Advanced Timber Space-Frame Structures, in: Robot. Fabr. Archit. Art Des. 2016, 2016: pp. 190–203. https://doi.org/10.1007/978-3-319-04663-1.

[27] P. Eversmann, F. Gramazio, M. Kohler, Robotic prefabrication of timber structures: towards automated large-scale spatial assembly, Construction Robotics. 1 (2017) pp. 49–60. https://doi.org/10.1007/s41693-017-0006-2.

[28] Thomas Staff Writer, Understanding CNC Machining, Articles. (2021). https://www.thomasnet.com/articles/custom-manufacturing-fabricating/understanding-cnc-machining/ (accessed February 2, 2021).

[29] P. Martinez, M. Livojevic, P. Jajal, D.R. Aldrich, M. Al-Hussein, R. Ahmad, Simulation-Driven Design of Wood Framing Support Systems for Off-Site Construction Machinery, Journal of Construction Engineering and Management. 146 (2020) pp. 04020075. https://doi.org/10.1061/(asce)co.1943-7862.0001853.

[30] I.J. Ramaji, A.M. Memari, Product Architecture Model for Multistory Modular Buildings, Journal of Construction Engineering and Management. 142 (2016) pp. 1–14. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001159.

[31] NIMBS Committe, National Building Information Modeling Standard, Nbim. (2007) pp. 180. https://doi.org/10.1017/CBO9781107415324.004.

[32] B. Becerik-Gerber, K. Kensek, Building Information Modeling in Architecture, Engineering, and Construction: Emerging Research Directions and Trends, Journal of Professional Issues in Engineering Education and Practice. 136 (2010) pp. 139–147. https://doi.org/10.1061/(asce)ei.1943-5541.0000023.

[33] C. Eastman, P. Teicholz, R. Sacks, K. Liston, BIM handbook: a guide to building information modeling for owners, managers, designers, engineers and contractors, 2nd ed.,

Hoboken, NJ : Wiley, 2011.

[34] M.P. Gallaher, A.C. O'Connor, J.L. Dettbarn, L.T. Gilday, Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry. National Institute of Standards and Technology (NIST), (2004). http://www.bentleyuser.dk/sites/default/files/nist_report.pdf.

[35] E.A. Poirier, D. Forgues, S. Staub-French, Dimensions of interoperability in the AEC industry, in: Constr. Res. Congr. 2014, 2014: pp. 1987–1996. https://doi.org/10.1061/9780784413517.176.

[36] U. Isikdag, G. Aouad, J. Underwood, S. Wu, Building information models: a review on storage and exchange mechanisms, Proceedings of the CIB W78's 24th International Conference on IT in Construction. (2007) pp. 135–143.

[37] buildingSMART, Industry Foundation Classes (IFC), (2019). https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/ (accessed July 28, 2019).

[38] K. Afsari, C. Eastman, D. Shelden, Cloud-Based BIM Data Transmission: Current Status and Challenges, in: Proc. 33rd Int. Symp. Autom. Robot. Constr., 2016. https://doi.org/10.22260/isarc2016/0129.

[39] Y. Arayici, T. Fernando, V. Munoz, M. Bassanino, Interoperability specification development for integrated BIM use in performance based design, Automation in Construction. 85 (2018) pp. 167–181. https://doi.org/10.1016/j.autcon.2017.10.018.

[40] J. Steel, R. Drogemuller, B. Toth, Model interoperability in building information modelling, Software and Systems Modeling. 11 (2012) pp. 99–109. https://doi.org/10.1007/s10270-010-0178-4.

[41] R. Ren, J. Zhang, H.N. Dib, BIM Interoperability for Structure Analysis, in: Proc., Constr. Res. Congr., ASCE, Reston, VA, 2018: pp. 470–479.

[42] J. Wu, J. Zhang, New Automated BIM Object Classification Method to Support BIM Interoperability, Journal of Computing in Civil Engineering. 33 (2019) pp. 04019033 1–17. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000858.

[43] BuiildingSMART, Certified Software, Software Certification. (2020). https://www.buildingsmart.org/compliance/software-certification/certified-software/ (accessed June 8, 2020).

[44] H. Liu, G. Singh, M. Lu, A. Bouferguene, M. Al-Hussein, BIM-based automated design and planning for boarding of light-frame residential buildings, Automation in Construction. 89 (2018) pp. 235–249. https://doi.org/10.1016/j.autcon.2018.02.001.

[45] M. Abushwereb, H. Liu, M. Al-Hussein, A Knowledge-Based Approach Towards Automated Manufacturing-Centric BIM: Wood Frame Design and Modelling for Light-Frame Buildings, in: Modul. Offsite Constr. Summit Proc., Banff, AB, 2019: pp. 100–107. https://doi.org/https://doi.org/10.29173/mocs82.

[46] S. Isaac, T. Bock, Y. Stoliar, A methodology for the optimal modularization of building

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

design, Automation in Construction. 65 (2016) pp. 116–124. https://doi.org/10.1016/j.autcon.2015.12.017.

[47] P. Jensen, T. Olofsson, H. Johnsson, Configuration through the parameterization of building components, Automation in Construction. 23 (2012) pp. 1–8. https://doi.org/10.1016/j.autcon.2011.11.016.

[48] M. Mekawy, F. Petzold, Exhaustive Exploration of Modular Design Options to Inform Decision Making, in: A. Fioravanti, S. Cursi, S. Elahmar, S. Gargaro, G. Loffreda, G. Novembri, A. Trento (Eds.), Proc. 35th Int. Confer- Ence Educ. Res. Comput. Aided Archit. Des. Eur., Rome, Italy, 2017: pp. 107–114.

[49] A.Q. Gbadamosi, A.M. Mahamadu, L.O. Oyedele, O.O. Akinade, P. Manu, L. Mahdjoubi, C. Aigbavboa, Offsite construction: Developing a BIM-Based optimizer for assembly, Journal of Cleaner Production. 215 (2019) pp. 1180–1190. https://doi.org/10.1016/j.jclepro.2019.01.113.

[50] M. Wang, S. Ahn, Y. Zhang, M. Sadiq Altaf, M. Al-Hussein, Y. Ma, Automatic Material Estimation by Translating BIM Data into ERP Readable Data for Panelized Residential Construction, in: Modul. Offsite Constr. Summit Proc., 2019: pp. 9–16. https://doi.org/10.29173/mocs71.

[51] S.P. Root, J. Cribbs, A.D. Chasey, Case Study: Off-site manufacturing of EIFS Panelized Wall Assemblies to Gain Efficiency in Construction Sequencing, Modular and Offsite Construction (MOC) Summit Proceedings. (2019) pp. 357–364. https://doi.org/10.29173/mocs114.

[52] L. Sterling, E. Shapiro, The Art of Prolog: Advanced Programming Techniques, Second Edi, The MIT Press, Cambridge, Massachusetts, 1994 ISBN: 9780262193382.

[53] A.-L. Johansson, A. Eriksson-Granskog, A. Edman, Prolog Versus You An Introduction to Logic Programming, Uppsala, Sweden, 1989. https://doi.org/10.1017/CBO9781107415324.004.

[54] M.P.H. Uszkoreit, M.V.W. Wahlster, M.J. Wooldridge, B.G. Buchanan, P.J. Hayes, J.A. Hendler, N. Jennings, H. Kamp, R. Kowalski, H. Levesque, S. Oviatt, An Introduction to Language Processing with Perl and Prolog, 2006. https://doi.org/10.1007/3-540-34336-9.

[55] J. Zhang, N.M. El-Gohary, Semantic-based logic representation and reasoning for automated regulatory compliance checking, Journal of Computing in Civil Engineering. 31 (2016) pp. 1–12. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000583.

[56] J.A. Makowsky, Why Horn Formulas Matter in Computer Science: Initial Structures and Generic Examples, Journal of Computer and System Sciences. 34 (1987) pp. 266–292. https://doi.org/10.1007/3-540-15198-2_24.

[57] M. Triska, The Power of Prolog, Self-published, 2005.

[58] N.-F. Zhou, B-Prolog User's Manual (Version 8.1): Prolog, Agent, and Constraint Programming, 2014.

[59] J. Väänänen, Second-order and Higher-order Logic, The Stanford Encyclopedia of Philosophy. (2019). https://plato.stanford.edu/archives/fall2020/entries/logic-higher-order/.

[60] G. Luger, W. Stubblefield, AI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and Java, 2009. https://doi.org/10.1017/CBO9781107415324.004.

[61] G. Ritchie, Prolog Step-by-Step, (2002).

[62] M. a Covington, D. Nute, A. Vellino, Prolog programming in depth, Knowledge Creation Diffusion Utilization. (1995) pp. 515.

[63] F. Portaro, Automated Reasoning, The Stanford Encyclopedia of Philosophy. (2011). https://plato.stanford.edu/archives/sum2011/entries/reasoning-automated/.

[64] S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, Second, Pearson Education, Inc., Upper Saddle River, New Jersey, 2003 ISBN: 0-13-790395-2.

[65] A. Darko, A.P.C. Chan, M.A. Adabre, D.J. Edwards, M.R. Hosseini, E.E. Ameyaw, Artificial intelligence in the AEC industry: Scientometric analysis and visualization of research activities, Automation in Construction. 112 (2020) pp. 103081. https://doi.org/10.1016/j.autcon.2020.103081.

[66] K. Jain, D., K.H. Law, Knowledge representation with logic., in: Stanford Univ., Stanford, CA, 1989.

[67] OWL Working Group, Web Ontology Language (OWL), OWL. (2012). https://www.w3.org/OWL/ (accessed February 1, 2021).

[68] N.M. El-Gohary, T.E. El-Diraby, Domain Ontology for Processes in Infrastructure and Construction, Journal of Construction Engineering and Management. 136 (2010) pp. 730–744. https://doi.org/10.1061/(asce)co.1943-7862.0000178.

[69] J. Zhang, N.M. El-Gohary, Integrating semantic NLP and logic reasoning into a unified system for fully-automated code checking, Automation in Construction. 73 (2017) pp. 45–57. https://doi.org/10.1016/j.autcon.2016.08.027.

[70] B.I. Dahn, Robbins algebras are Boolean: A revision of McCune's computer-generated solution of Robbins problem, Journal of Algebra. 208 (1998) pp. 526–532. https://doi.org/10.1006/jabr.1998.7467.

[71] E. Kotelnikov, Automated Theorem Proving with Extensions of First-Order Logic, Chalmers University of Technology, 2018. https://research.chalmers.se/publication/504640/file/504640_Fulltext.pdf.

[72] P. Pauwels, W. Terkaj, EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology, Automation in Construction. 63 (2016) pp. 100–133. https://doi.org/10.1016/j.autcon.2015.12.003.

[73] E. González, J.D. Piñeiro, J. Toledo, R. Arnay, L. Acosta, An approach based on the ifcOWL ontology to support indoor navigation, Egyptian Informatics Journal. (2020). https://doi.org/10.1016/j.eij.2020.02.008.

This article was published in Automation in Construction, 129(September 2021), Wong Chong, O., and Zhang, J., "Logic Representation and Reasoning for Automated BIM Analysis to Support Automation in Offsite Construction," 103756, Copyright Elsevier (2021). https://www.sciencedirect.com/science/article/pii/S0926580521002077

[74] P. Pauwels, T. Krijnen, W. Terkaj, J. Beetz, Enhancing the ifcOWL ontology with an alternative representation for geometric data, Automation in Construction. 80 (2017) pp. 77–94. https://doi.org/10.1016/j.autcon.2017.03.001.

[75] J. Zhang, N.M. El-Gohary, Automated Extraction of Information from Building Information Models into a Semantic Logic-Based Representation, in: 2015 ASCE Intl. Work. Comput. Civ. Eng., ASCE, Reston, VA, 2015: pp. 173–180. https://doi.org/10.1061/9780784479247.083.

[76] P.H. Winston, Artificial intelligence, Wesley Publishing Company, United Kingdom, 1992 ISBN: 0201533774.