TECHNOLOGICAL CHALLENGES OF BIM-BASED AUTOMATIC RULE CHECKING: A REVIEW

Cauê C. Santiago, Msc.[1] and Ana Regina Mizrahy Cuperschmid, PhD[2]
[1] School of Civil Engineering, Architecture and Urban Design, Department of Architecture and Construction, State University of Campinas (UNICAMP)
c226650@dac.unicamp.br
[2] School of Civil Engineering, Architecture and Urban Design, Department of Architecture and Construction, State University of Campinas (UNICAMP)
cuper@unicamp.br

ABSTRACT

BIM Systematics for Process Optimization have been widely developed in recent years, considering the benefits they provide to the Architecture, Engineering, and Construction (AEC) sector. One of the systematics that deserves to be highlighted is automatic rule checking, which allows for the automation of specific compliance relevant to several design modalities. With this in mind, it is imperative to search for ways to automate processes of design compliance. This study, by a systematic literature review (SR), seeks to understand the current state of the art regarding the automation of BIM-based design compliance. With it, it was possible to identify the main points inherent to the process, such as (i) the main tools used, (ii) how the Industry Foundation Class scheme was used, (iii) the procedures and challenges found for the development of design. The evaluation indicates that it is possible to highlight the complexity of the process, especially for its implementation. Each study approaches the topic in a different way, using specific strategies. The use of hard-coded mechanisms proved to be the most suitable for developing specific systems. The findings suggest that the greatest difficulties and limitations of the studies analyzed are directly related to the technology axis, more specifically regarding the treatment and manipulation of geometry in general.

KEYWORDS: *Rule Checking, Model Checking, BIM, Limitations.*

I. INTRODUCTION

The adoption of Building Information Modeling (BIM) is constantly growing worldwide, given its potential in the construction sector [1]. In the past five years, it has undergone major innovations and has a constant flow of growth from innovative companies, which have as their specialty the application of some automation and information technologies [2].

It is clear that the Brazilian government understands the importance of BIM, designing strategies for its dissemination nationwide. Presidential Decree No. 9,983 of 2019 establishes the Strategy Management Committee of Building Information Modelling, and it also discusses the BIM-BR Strategy. Corroborating this, Decree No. 10,306 of 2020 establishes that BIM should be used in works considered of great relevance, when in phases 1 and 2, and of medium or great relevance, when in phase 3, in the direct or indirect execution of engineering works and services carried out from 2021 by the agencies and entities of the Federal Public Administration.

In addition, the Law on Bids and Administrative Contracts, Law No. 14,133, of April 1, 2021, also indicates in its Chapter II, Of the Preparatory Phase, Section I, Of the Instruction of the Bidding Process, Art. 19, item V, paragraph 3, that BIM should preferably be adopted for the bidding of engineering and architectural works and services [3].

To enable the process of digital approval of design, several tools have been developed and used [4]. The uses of BIM include code checking, which can be applied in the most diverse disciplines and for various purposes, such as for checking fire and panic safety standards [5]. Code checking aims to verify the conformities of the model regarding the imposed rules. Model checking, using systematic automation, considerably increase the effectiveness of the design processes, as well as reduce the time with manual checks [6].

Code checking can be applied in a much broader aspect, from regulations of public agencies to specific instructions from clients and private organizations. In general, the scope of rule automation falls into the following categories [7]:

- Compliance of the good formation of a model;
- Checking construction standards;
- Feasibility and other requirements of the contractor;
- Security and other rules that enable corrective actions;
- Approval of guarantees (insurance);
- Completeness of BIM data for delivery to facility management.

However, automatic rule checking precedes BIM itself [8]. According to the authors, this topic has been a subject of research since the 1960s. Currently, several studies seek to understand the mechanisms and processes for the development of systems that automatically perform code checking in BIM. Several of them deal with the semantics involved in the processes, making use of Systematics such as RASE, PAS, NLP, among others. Other studies deal with the creation of specific tools, presenting a use case, also using several approaches. Several approaches on the integration of BIM and GIS have also been studied, and it is noticeable that an agreement between the semantic use, interface, model information, must be made in order to make the integration possible [9]. It is noticed that even with several studies carried out in recent years and with the various methodologies addressed, there is still no ideal solution that is comprehensive enough for the various specificities of the AEC sector.

This article presents a systematic literature review (SR), in search of understanding the technological challenges inherent to automatic rule checking processes in BIM. For this purpose, the following guiding questions were raised:

- Which systems can be identified in the process of automatic rule checking in BIM?
- Which approaches to automatic rule checking are most used?
- What are the main difficulties found in this process?
- For designers to be able to perform automatic rule checking in BIM, would using the IFC format be a prerequisite?
- It is possible to develop a customizable system for digital approval of BIM-based design without extensive knowledge in software programming languages?
- How could the system be incorporated into the digital approval of BIM-based design?

This paper initially provides a review of automatic rule checking in BIM, with a focus on delineating the steps and approaches involved in this process. Subsequently, we elucidate the SR method, grounded in the Integrated Method [43]. Following this, we present the results aimed at addressing the research inquiries. Lastly, in the conclusion, we engage in a discussion of the results, highlighting the technological challenges of BIM-based automatic rule checking.

II. AUTOMATIC RULE CHECKING IN BIM

We will address below the steps of code checking, the possible approaches, and levels.

2.1. Steps for automatic rule checking

In the context of automatic rule checking, it is possible to identify four steps for its implementation: (i) rule interpretation and logical structuring; (ii) preparation of the construction model; (iii) rule execution; and (iv) export of checking results [10] (Figure 1).

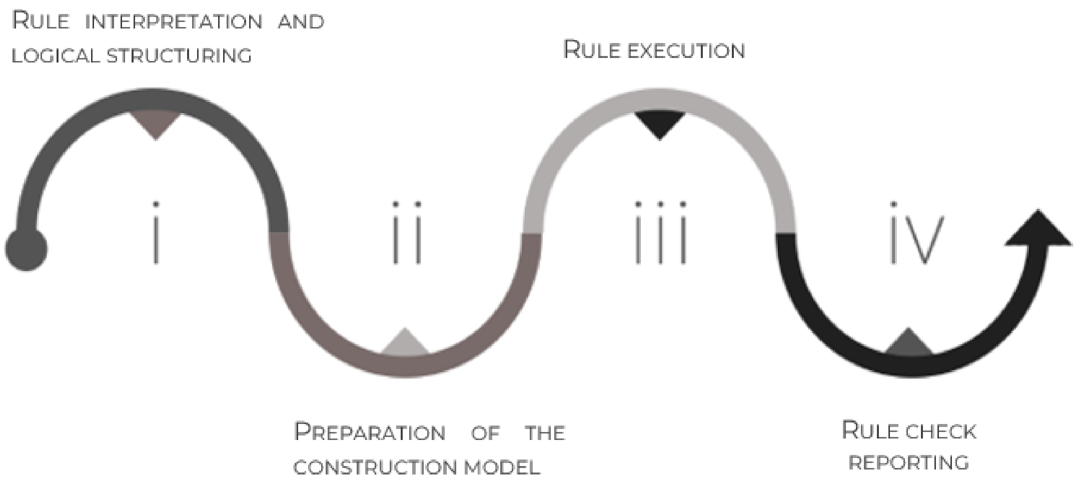

Figure 1. Steps for automatic rule checking

a) Rule interpretation and logical structuring

The rules, laws, decrees, and other legal documents that bring guidelines and restrictions regarding the approval of designs, whether in any scope, are generally texts created and represented in natural language, that is, they do not necessarily have a lexicon and logical structuring that can be interpreted by machines without modifications being made [11].

One of the great challenges found in this stage lies precisely in the transcription of rules written in natural language to a language that can be recognized and interpreted by the machine, also called translation from natural language (NL) to machine language (ML) [12]. The rules written and developed naturally often bring subjective information. Thus, they need to be translated into a language that can be interpreted by computer systems [13]–[16].

An intermediate language that maps natural rules and transposes them into machine language is the first-order-predicate logic [10]. The predicate is an expression that assigns a property to the object identified by the variable of the sentence, and the logic demonstrated deals with making substitutions due, in sentence excerpts, for the validation of the predicate, as true or false, also dealing with quantification sentences [17].

Sentences without affirmative values that lack proper attributions such as “it is an integer” do not have the proper attribution and, in turn, cannot be validated. It is only with the inclusion of an initial parameter that True-False statements can be made to the sentence. If the number “4” is included, then the sentence is true, since “4 is an integer” corresponds to reality. Similarly, if the number “3.14” is included, then the sentence is false, since “3.14 is an integer” does not correspond to reality.

Another way to create a logical structuring is by the ontology of names and properties [10]. The term ontology was borrowed from Philosophy, which deals with the nature of existence and, for automated or artificial intelligence systems, everything that "is" can be represented [18].

Systematics can be defined as a domain-specific conceptualization (e.g., civil engineering or software engineering) that provides explicit logical statements about classes, instances, and properties, and translates natural language rules into machine language [19]. In a simplified way, there is an interface between the ontology system, which has an initial information library, and the instances of the model [20]. From this, the union between the pairs of information is carried out, so that the sentence in natural language can be clearly transposed into machine language and be usable by checking systems. An example would be the ontology "OWL: cardinality (n)" that seeks information about the number of values that a given property has.

b) Preparation of the construction model

The current needs, regarding the requirements of a model, are much higher than the needs related to a two-dimensional design [10]. According to the authors, previously, the representation needed to have a visual similarity with what was intended to happen. However, within the BIM universe, each object has embedded specific information relevant to its type or class and not only to visual appearance.

The Industry Foundation Class (IFC) is the neutral scheme within the BIM universe [21]. Its purpose is the possibility of defining specifications of the information or exchange data between various sectors of knowledge [22]. The IFC file allows the exchange of data between the various sectors of the AEC, bringing out the possibility of interoperation [23]. To establish efficient communication between the various components of the sector, since each one has its own specificities, one must create a subset of information, which makes up the IFC file [24]. This subcomponent is called Model View Definition (MVD).

An MVD is a subset of the IFC scheme that is intended to facilitate the BIM workflow by selecting certain specifications that are more representative for each use (Figure 2) [25].

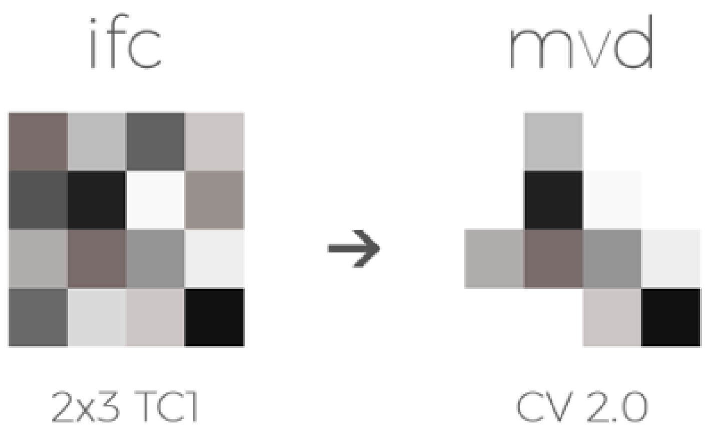

Figure 2. IFC and MVD

The creation of MVDs without proper standardization can lead to incompatibility between IFC models, thus generating visualization problems and lack of information [25]. To work around this problem, object-oriented mechanisms are developed by the creation of improved model or construction objects. This approach works for the vast majority of cases but may not be sufficient for more complex sets of spatial configurations [10].

Design creation, which derive from base models, or templates, is also an alternative to try to facilitate the preparation of their requirements concerning automatic rule checking [10]. A template can be developed and customized based on the specific criteria for each case, and it may contain ideal parameters for the checking procedures [26].

c) Rule execution

The rule execution phase plays the role of uniting the previous stages: the structured rules system and the duly consolidated model [10]. Also, according to the authors, having transcribed the rules from natural language to machine language and the connections between such information being consistent with the information contained in the model, a simple and objective process takes place, in which the appropriate checks are made, and the system can be tested.

The model can also go through a previous check, to check for errors, omissions, interferences, and contradictory definitions, to later go through the sieve of automatic rule checking [27]. This pre-checking can be done by authoring software or systems, or external commercial software created specifically for the situation.

Since 2009, it has been pointed out that, for many years, code reviews would be a mixture of automatic and manual validation, due to the implementation and incremental release of the base rules, arising from laws, decrees, etc. [10]. Currently, it is observed that the uses of BIM have undergone significant advances. Nevertheless, regarding the approval of BIM-based designs in an automated way, a certain delay is noticeable concerning the developments of platforms, systems, etc.

d) Rule check reporting

The last step in the process of creating an automatic rule checking system is the check report. The conditions imposed by the rules – in the first step –, if satisfied, should receive a positive return, otherwise they should receive a negative return [28]. The way this report will be shown to the user may change, and may appear in graphic or textual forms, showing which rules have not been met.

It is clear that the stage of rule interpretation and logical structuring, especially regarding the translation of rules from natural language to machine language, is a crucial stage for the development of a functional rule checking system, being the main obstacle to a more substantial implementation of this BIM use [29].

2.2. Approaches for automatic rule checking

It was possible to observe four types of approach for implementing a rule checking system, namely: (a) hard-coded; (b) object-oriented; (c) logical; and (d) semantic/ontological (Figure 3) [11].

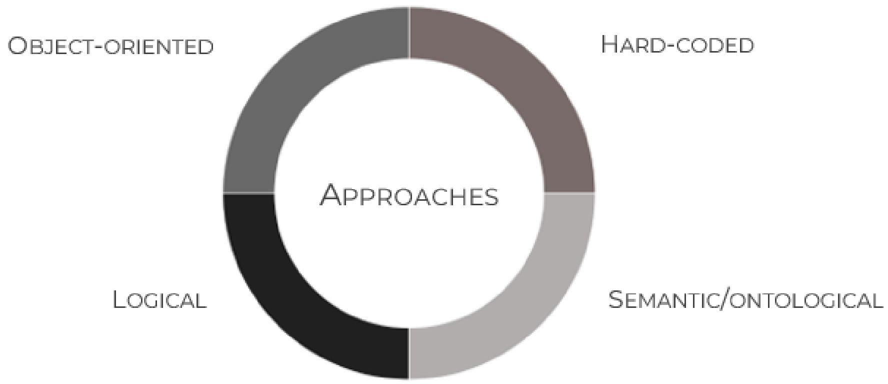

Figure 3. Different approaches

a) Hard-coded

The hard-coded approach uses commercial software or checking systems and has the convenience of being more practical, since several pre-programmed rules are already embedded within such software [30]. Some of the best-known commercial software for this approach: Solibri Model Checker (SMC), SMARTreview, FORNAX (CORENET System), Jotne Express Data Manager (EDM) Model Checker, among others.

However, due to the great manual effort in extracting pertinent information from legal texts, hard-coded systems are less scalable [31] and consequently do not achieve a fully automated process [32]. The fact that they are programs that: (i) use black-box approaches, that is, they do not make it clear to the user how the compliance process was carried out; (ii) have a high acquisition cost; (iii) are inflexible with respect to modifications and; (iv) have a lack of generalist structures, which can be applied in various situations and various modalities of the AEC sector, contribute to the use of other checking approaches [15].

For some authors, black-box approaches should not be carried out in a research context, since it is not possible to verify the analyses and the critical reflection of the processes becomes flawed [28].

b) Object-oriented

The object-oriented approach, also called object-based, addresses construction objects and rule objects [30]. An advantage of the object-oriented approach over the hard-coded is in the possibility of creating rules in such a way that the selection of elements can be alternated as needed by the user, thus generating a greater malleability of the system [16].

Because there is the possibility of creating rules and control over the checking processes, the advantages over black-box mechanisms are large, as it becomes more suitable for customization [28]. In general, the object-oriented approach, when combined with visual programming languages (e.g. Dynamo, Grasshopper, Marionette), can facilitate the understanding of the logic of the check programming [33], [34]. In addition to the above, the object-oriented approach has been gradually adopted in machine learning, which has been the subject of research in the area [30].

c) Logical

The logical approach, on the other hand, is basically based on two distinct parts, namely the conversion of NL to ML and obtaining a properly detailed BIM model [35]. Within the spectrum of rule checking, intelligent systems such as Artificial Intelligence (AI) can play an important role regarding the semantic enrichment of parts of the concept code, defined in the MVD, the enrichment of topological relations, and rule checking, using Deep Learning [36].

For the translation of natural order rules, some research points to the Requirement, Applicability, Select, and Exception method, or RASE. This methodology works with markers for specific words in a text, assigning one of four possible operators, then creating a structure that can be interpreted by machines [37].

Another possibility for the logical approach is using the Building Environment Rule and Analysis, or BERA, which is a domain-specific programming language (Figure 4) aimed at analyzing and accessing rules that concern the built environment [38].

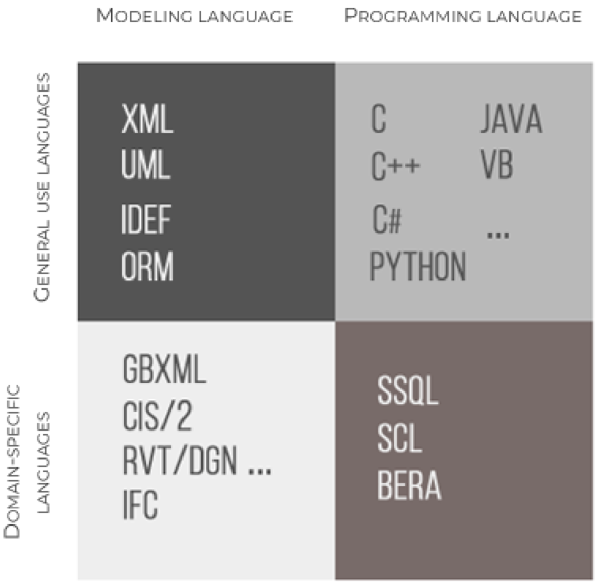

Figure 4. Differences between languages (Adapted from [38])

These and other ways of dealing with the translation of rules from NL to ML focus on the structure of the rules themselves and do not deal with the problems of semantic interpretation that natural rules bring with them [39].

d) Semantic/ontological approach

Basic concepts of construction models and their relationships can be described by ontological modeling. Web semantics and ontological technologies are already used in the AEC sector [30]. Since the IFC does not describe all the relationships and interdependencies between construction elements, the idea of their semantic enrichment becomes attractive [40].

Such an approach can facilitate information exchange, integrations, and optimize the ability to collaborate across distinct systems [41]. For the AEC sector to promote the use of web semantic approaches, BuildingSMART standardized an ifcOWL ontology. From there, protocols such as SPARQL and query languages, such as RDF or Semantic Web Rule Language (SWRL) for data extraction, begins to become more widespread [42].

III. METHOD

There are several methods for conducting a systematic literature review (SR) [43]. However, it is noticeable that there is a common core between the different ways of treating an SR. This research is carried out by the Integrated Method [43], who compile and expands the existing methods in literature. This method goes through the following steps: (i) definition of the central theme and conceptual framework; (ii) choice of the work team; (iii) search strategy; (iv) search, eligibility, and coding; (v) quality assessment; (vi) synthesis of results; (vii) presentation of the study.

This SR has as its central theme the analysis of research involving the automatic rule checking in BIM, focusing on the verification of procedures adopted in use cases. The main question that motivated the SR was: regarding the automatic rule checking in BIM, which are the main tools used, how was the use of the IFC scheme, what are the procedures and challenges found for the development of the design?

Concerning the search strategy, the following databases were used: ScienceDirect, Scopus, and Web of Science, and for each of these databases, searches were made in journal articles published in English between 2016 and 2021, excluding review articles. The search strings were the following: ((“CODE”

OR "RULE" OR "COMPLIANCE") AND ("CHECK" OR "CHECKING") AND ("BIM" OR "BUILDING INFORMATION MODEL")). These terms were searched in the abstract, title, or keyword of the articles (Figure 5).

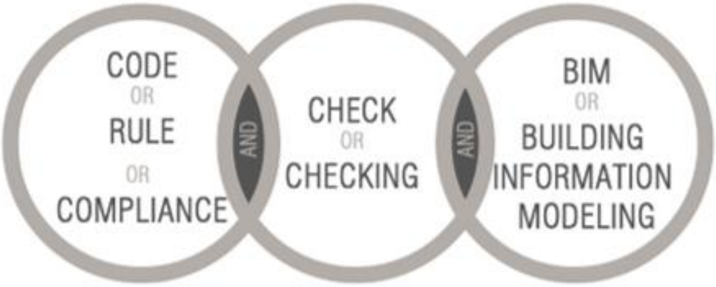

Figure 5. Search string

Regarding search, eligibility, and coding, the following exclusion criteria were defined for obtaining bibliometric data and consequent analysis of the research: (1) duplicate articles; and (2) articles in a language other than English. As inclusion criteria (3), only articles dealing with automatic rule checking were considered. For this purpose, the titles and abstracts were read.

Initially, a bibliometric analysis was performed with all articles that passed the exclusion and inclusion criteria. The articles were classified according to the year of publication, journal, co-occurrence of keywords, and nomenclature.

Concerning quality assessment, relevant studies were selected for the review. We verified their adequacy to the focus of the review, that is, cases in which automatic rule checking has been described in the article. For this purpose, a segmentation was carried out according to the pertinence of the study, discriminating the other research focuses addressed in the articles.

Subsequently, the synthesis of the results was carried out with the articles that presented a use case. Such articles were classified according to the purpose of the checking, the approach used, the modeling and checking software used, whether they use the IFC scheme, which programming language was used by the authors (in the case of development of specific tools), and difficulties found.

IV. RESULTS

4.1 Search, eligibility, and coding

After searching the databases, 52 articles were found in ScienceDirect, 82 articles in Scopus, and 86 articles in Web of Science, totaling 220 articles. When applying the exclusion criterion of duplicity of articles, 117 studies were obtained. The language exclusion criterion resulted in the removal of one article.

Regarding the inclusion criterion, after reading the titles and abstract, we found that 57 articles did not deal with automatic rule checking and 59 addressed the theme. Thus, the approved articles corresponded to 50.42% of the total articles (Figure 6).

Figure 6: Approved articles

From the 59 resulting articles, a bibliometric analysis was performed, aiming to obtain an overview on the production of the articles. The results show 13 articles published in 2016, seven published in 2017, five published in 2018, 13 published in 2019, 14 published in 2020, and seven published in 2021 (Figure 7). An average of 9.83 articles were published per year in the indicated period, without, however, a continuous growth/decrease.

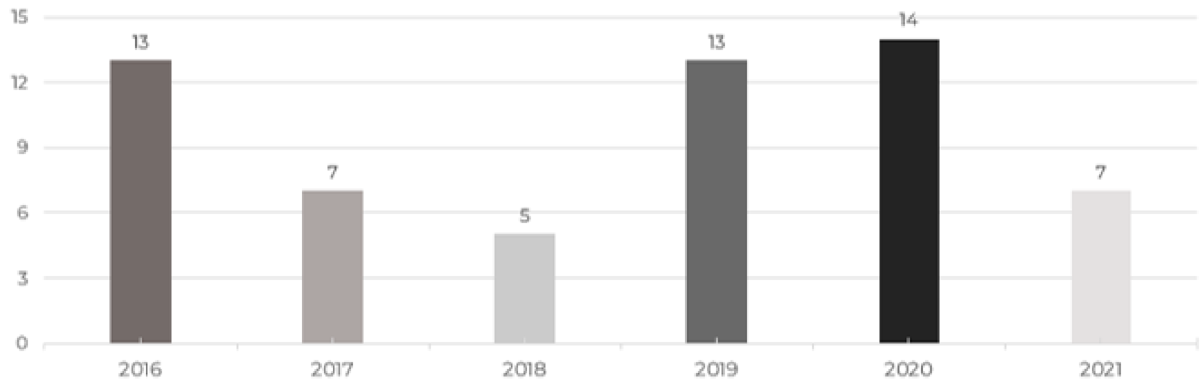

Figure 7: Year of publication

It was also possible to analyze which journals publish the most on the topic. The main one is the Journal Automation in Construction, since among the 59 approved articles, 19 were published in it; four were published in the Journal of Computational Design and Engineering; three in the Journal of Computing In Civil Engineering; and three in Advanced Engineering Informatics (Figure 8). The other journals had only two publications.

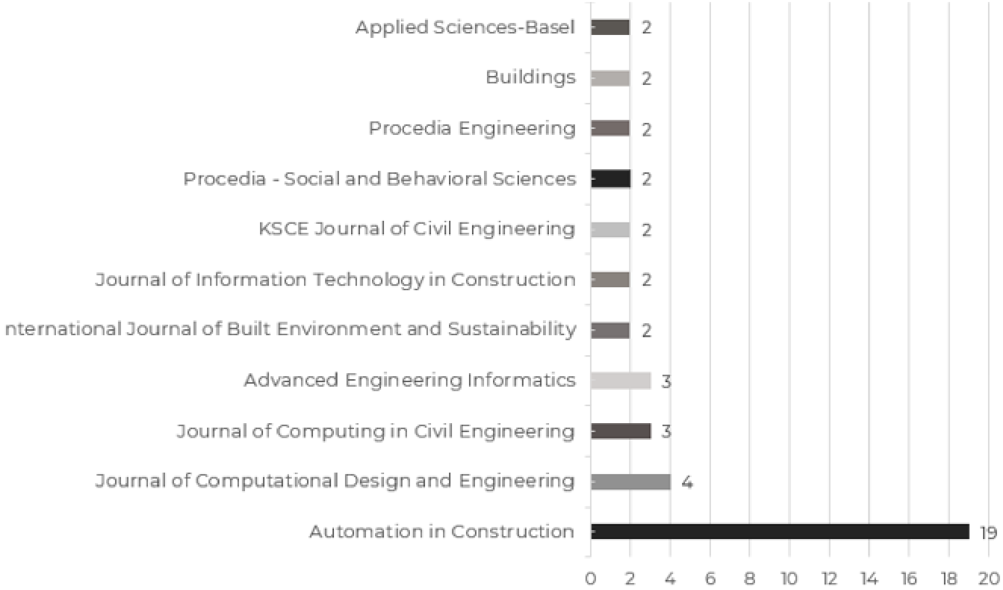

Figure 8: Publications per journal

It was possible to analyze the co-occurrence of keywords, based on the approved articles (Figure 9), and the term BIM – with several correlates, such as Building Information Modeling and Building Information Modelling grouped in the acronym – appears in 46 articles, followed by the term IFC, which appears in 14 articles. The words interoperability and rule checking both appear seven times; compliance checking, six times; automated compliance checking, five times; code checking, building code, and model view definition (MVD), four times. The remaining words appear between one and three times.

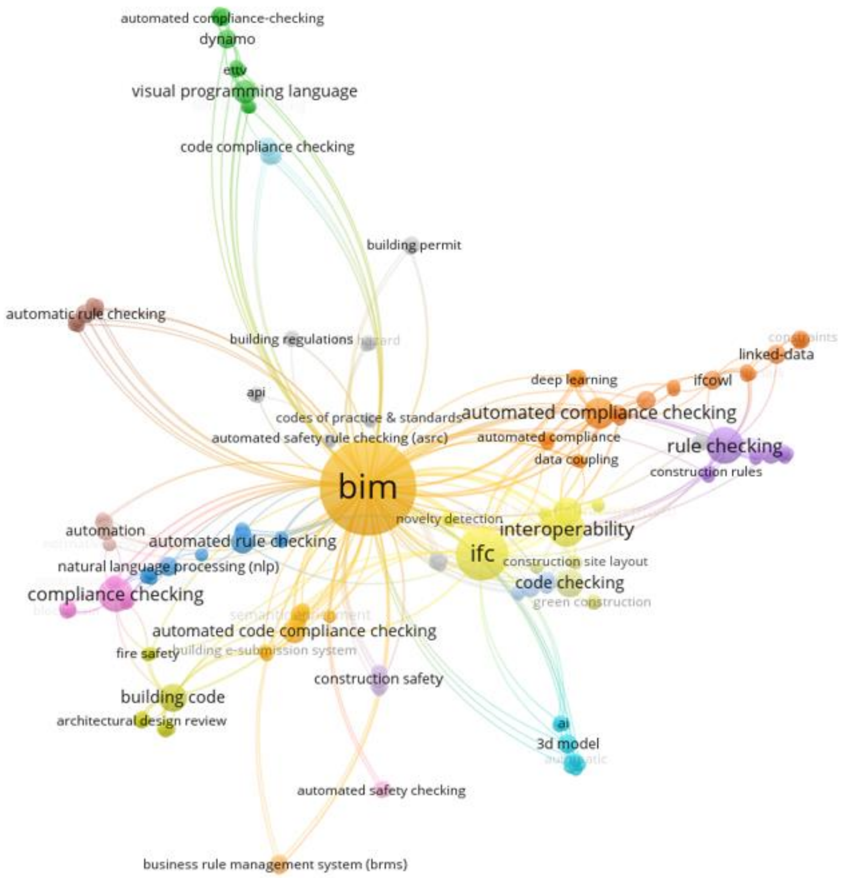

Figure 9: Co-occurrence of Keywords

Regarding the nomenclature, it was possible to verify that rule checking in BIM is sometimes called automated code checking, automated code compliance checking, automated compliance audit, automated compliance checking, but one can notice that, in most cases, generic terms are used, such as rule checking (RC), contextually associated with some specificity of the research being done at the time, for instance "automated safety rule checking."

It is clear that there is no consensus on the specific nomenclature. The most used terms were: (a) Rule Checking, seven times; (b) Automated Compliance Checking, six times; (c) Automated Code Checking, four times; (d) Automated Code Compliance Checking, four times; (e) Automated Rule Checking, four times; and (f) Code Checking, four times. The other names appeared between once and twice and are shown in Table 1.

Table 1: Nomenclature adopted in research

Nomenclature	Number of times it appears	Source
Rule Checking	7	[44]–[49]
Automated Compliance Checking	6	[50]–[54]
Automated Code Checking	4	[13], [14], [55], [56]
Automated Code Compliance Checking	4	[15], [57]–[59]
Automated Rule Checking	4	[16], [29], [34], [60]
Code Checking	4	[31], [61]–[63]
Automated Model Checking	2	[64], [65]
Automated Safety Rule Checking	2	[66], [67]
Automatic Rule Checking	2	[30], [68]
Compliance Checking	2	[69], [70]
Automated Compliance Audit	1	[71]
Automated Regulatory Compliance Checking	1	[72]
Automated Rule Base Inspection	1	[73]
Automated Rule Based	1	[24]
Automated Rule Compliance	1	[74]
Automated Rule Compliance Checking	1	[11]
Automated Rule-Based Checking System	1	[75]
Automatic Checking	1	[76]
Automatic Checking Of Semantic Information	1	[28]
Code Checking System	1	[12]
Code Compliance Checking	1	[77]
Rule Based	1	[78]
Rule Based Checking	1	[79]
Rule Based Code Checking	1	[80]
Rule-Based Checking	1	[81]
Utility Compliance Checking	1	[82]

4.2 Quality assessment

The quality assessment was performed with all articles approved in the search, eligibility, and coding process. To classify the relevance of the research to the study in question, the articles were segmented according to their main purpose, which are: (a) articles that address the implementation and evaluation of a code checking software or framework (b) articles that address the semantic issue of information from BIM models, objects, and processes; and (c) articles that use BIM for automatic rule checking.

It was possible to verify that 14 articles address semantic data, 14 articles provide information on implementation, and 31 articles present a case of applying some rule checking system (Figure 10). There is a balance between articles dealing with software implementation and validation and with discussion of semantic data, and a greater amount of research presents a use case for a specific AEC discipline.

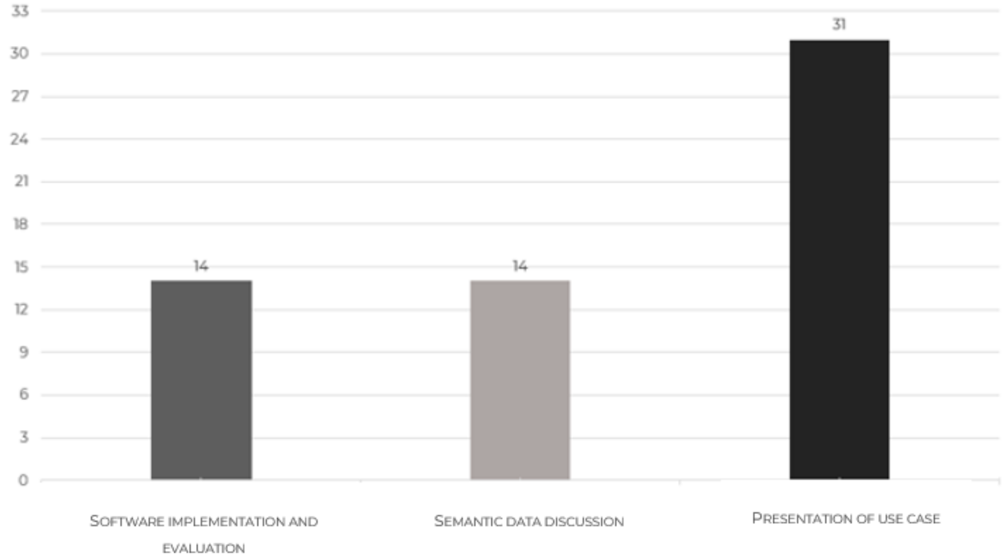

Figure 10: Purpose of the article

The articles on implementation and evaluation of software were categorized as follows: (a) addressed tests of benchmark – 1 article [45]; (b) described KBIM – 1 article [44]; (c) examined the application of code checking for productivity gain – 1 article [76](d) investigated the semantic differences and cognitive gap between the participants of the checking process – 1 article [12]; (e) reviewed the difficulties of applying the code checking system – 1 article [72]; (f) reviewed the approaches taken to date for the implementation of systematic automatic rule checking – 1 article [53]; (g) explored techniques, that is, possible application techniques for developing the use of automatic rule checking – 3 articles [64], [66], [83]; and (h) created a framework for using automatic rule checking – 5 articles [11], [30], [51], [67], [77] (Figure 11). Regarding the studies that dealt with software implementation and evaluation, most of them focused on creating a framework for automatic rule checking of some specific discipline of the AEC sector. Other studies made tests in different software to verify the processing time and examined the application for the productivity gain that the sector would have with the implementation of such system.

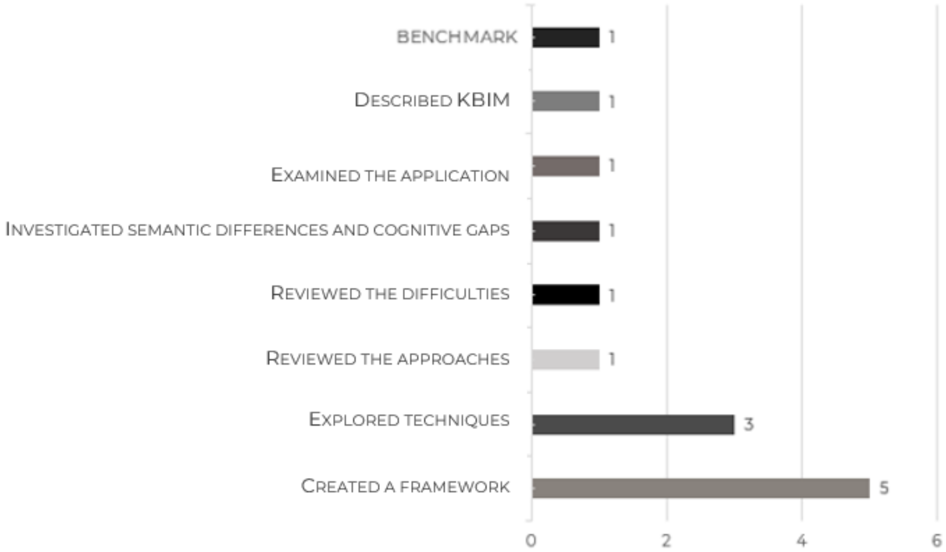

Figure 11: Software implementation and evaluation

The articles discussing semantic data were categorized as follows: (a) detection of misclassifications, in which a machine learning system is proposed to detect potential misclassifications that may occur in the mapping of a certain dataset – 1 article [84](b) data extraction, in which data extraction techniques are used to translate sentences, making use of deep learning – 1 article [29]; (c) integration of Natural Language Process (NLP) and Logic Reasoning, in which an integration of two techniques for dealing with semantic data is made – 1 article [32]; (d) translation from natural language (NL) to machine language (ML) – 1 article [85](e) data transformation, in which data alignment is performed with the use of deep learning and data are transformed into simplified schemes – 2 articles [54], [86]; (f) semantic enrichment, in which different methods of semantic enrichment are compared – 3 articles [50], [57], [59]; and (g) data validation – 5 articles [24], [47], [49], [81], [87] (Figure 12). It is noticeable that the processing of semantic data is indispensable, especially for the automation of rule and regulation translation, a process that is often entered manually. However, such processes are complex, involving machine learning, deep learning, natural language process, among others. The validation of the data that makes up automatic rule checking should be observed, since an erroneous attribution can generate false-positive results, decreasing the degree of effectiveness of the systems.

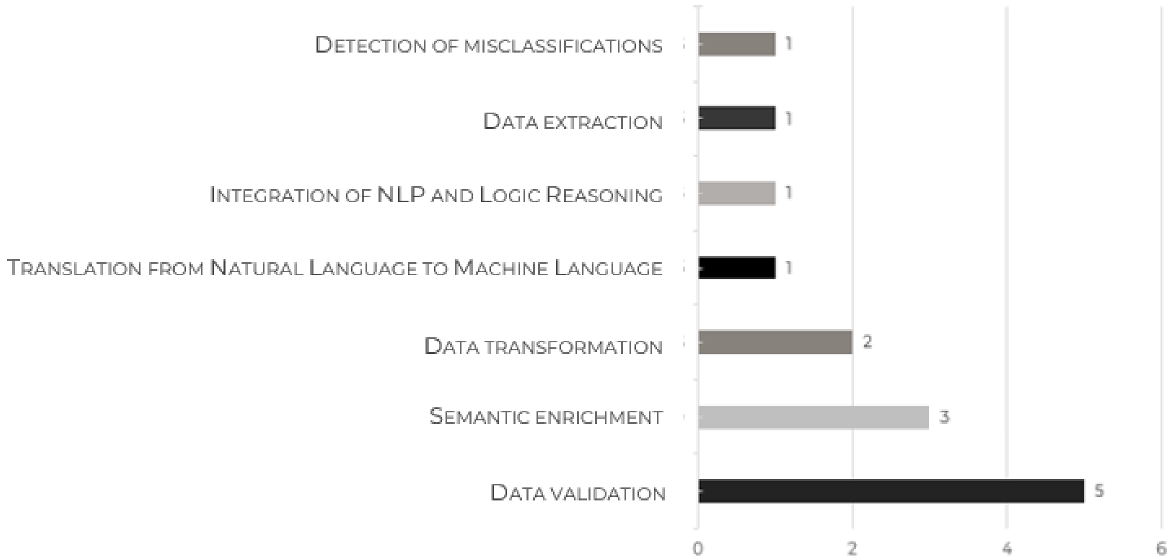

Figure 12: Discussion on semantic data

4.3 Summary of Results

The 31 articles that **presented a use case** were studied to answer the questions that motivated the SR. Thus, the articles were classified according to: (i) purpose of the checking; (ii) approach to rule checking; (iii) whether the studies used the IFC file; (iv) which main modeling software were used; (v) which specific software for rule checking were used; (vi) which programming languages were most used, for studies that developed their own systems; and (vii) what difficulties were found to perform automatic rule checking.

According to the purpose of the checking, there are: (a) maintenance and operation – 1 article; (b) sustainability, with proposals for automatic rule checking, aimed at checking and subsequent approval of models for obtaining environmental certificates and seals – 4 articles; (c) engineering/construction, addressing topics related to engineering or construction in general, such as construction site planning, cost estimation for design, among others – 6 articles; (d) safety, addressing checking aimed at construction site safety, accident prevention, security and fire and panic fighting, among others – 10 articles; and (e) architecture, addressing accessibility, design approval, following some specific system – 10 articles (Figure 13, Table 2).

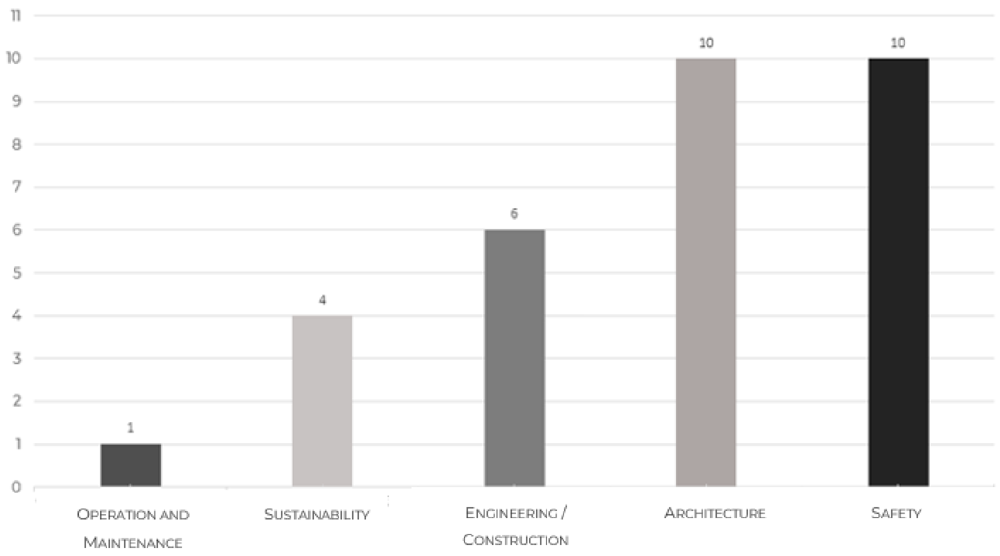

Figure 13: Purpose of the checking

Table 2: Purpose of the checking and its source

Purpose of the checking	**Source**
Maintenance and operation	[88]
Sustainability	[26], [31], [63], [78]
Engineering and construction	[13], [16], [28], [46], [69], [82]
Safety	[14], [34], [56], [61], [68], [71], [73], [74], [80], [89]
Architecture	[15], [48], [52], [55], [60], [62], [65], [70], [75], [79]

Concerning automatic rule checking, it is observed that the object-oriented approach was the most used, with 16 articles. Next, we have the logical approach, with seven articles; the hard-coded approach, with six; and the semantic/ontological approach and a combination between the logical and semantic approaches, with one article (Figure 14, Table 3). Few studies deal with the semantic/ontological approach, since it is a systematic approach that involves several specific processes and software.

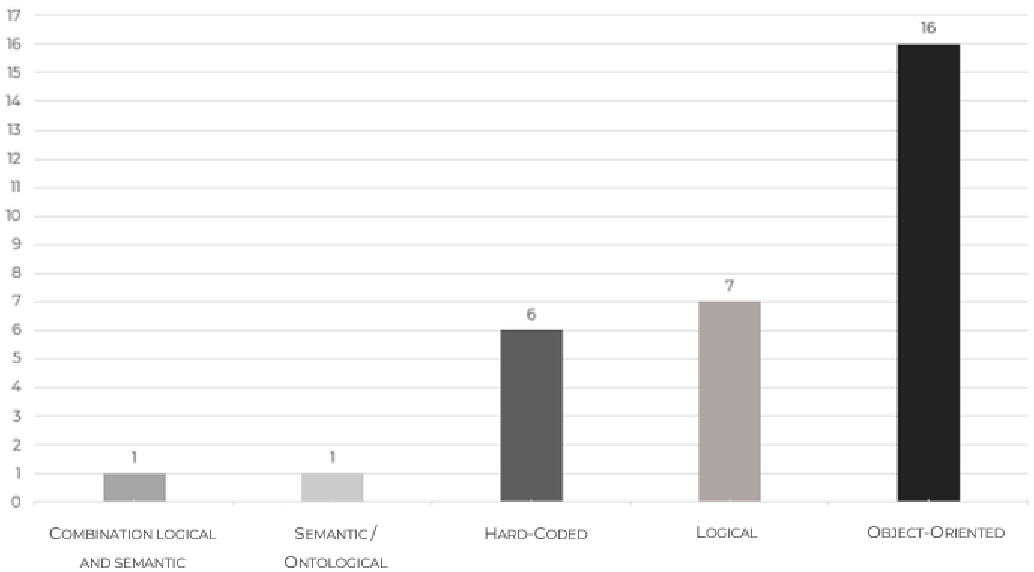

Figure 14: Approach

Table 3: Approach and its source

Approach	Source
Combination of logical and semantics	[31]
Semantic/ontological	[68]
Hard-coded	[14], [46], [55], [62], [63], [80]
Logical	[48], [61], [65], [70], [73], [79], [82]
Object-oriented	[13], [15], [16], [26], [28], [34], [52], [56], [60], [69], [71], [74], [75], [78], [88], [89]

The hard-coded approach ends up having as limitations the fact that, in general, software have a closed architecture (black-box), in which the user is not able to measure all the steps and processes of checking. Related to this is the high cost of software.

The logical approach, on the other hand, has perhaps the greatest ability to automate all stages of the rule checking process; this includes the transcription or translation of rules written in natural language into machine language. Nevertheless, it is a technique that requires great knowledge about data manipulation, machine learning, deep learning, natural language processing, among others, and this ends up limiting the advancement of this approach.

Possibly because it offers an ease in the elaboration of the processes, the object-oriented approach was the most used in the analyzed studies. Furthermore, in this approach, processes can be created directly in proprietary software, with the integration of visual language programs.

Regarding modeling programs, the vast majority – 17 articles – used Autodesk Revit for modeling; eight did not indicate which software they used; two used public models available on the internet; two used Archicad; one used Vectorworks; and one used AllPlan combined with DDS CAD (Figure 15). Most studies did not mention which version of the software was used.

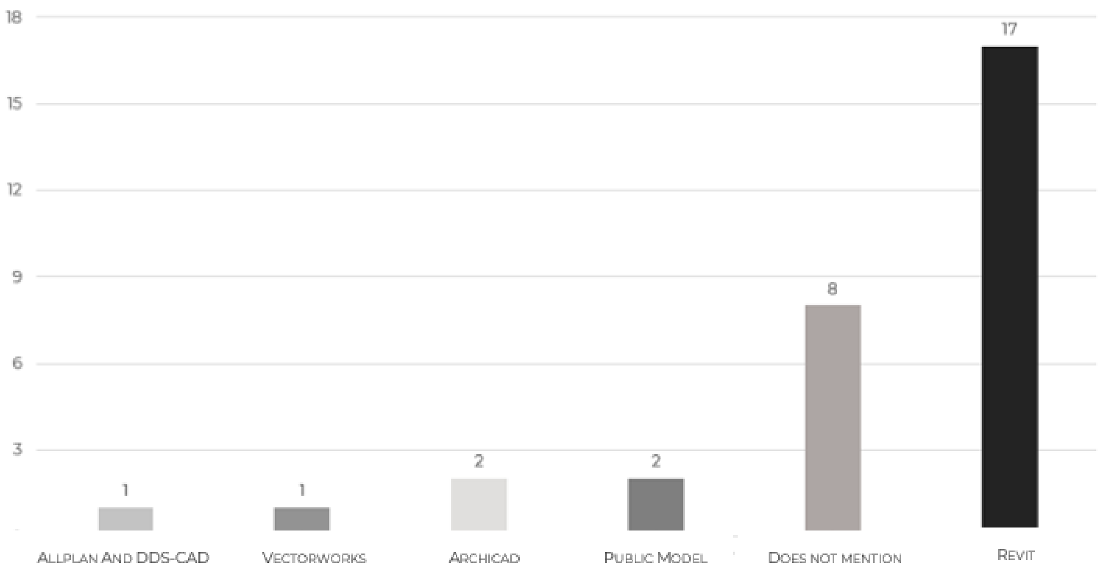

Figure 15: Modelling software

Commonly, interoperability in BIM is accomplished by the Industry Foundation Classes (IFC) scheme, and we therefore analyzed which research used it in their processes. For all articles that presented a concrete use case, 18 articles used IFC in their rule compliance processes, while 13 articles did not. Most studies did not bring a specific distinction between the IFC schemes, such as IFC 2x3 Coordination View 2.0 and IFC 4. Among the articles that present a use case and that make use of the IFC file, it is possible to perceive a preponderance in the use of software developed by the authors (Table 4). The use of commercial software is also made by the authors, as is the case of Solibri, Drools, and Open IFC Tools, as well as commercial software exclusive to a certain region, such as KBIM Assess and KBIM Assess-Lite.

Table 4: Software used and their sources

Software used	Source
Own development	[15], [31], [52], [65], [68], [70], [71], [75], [79], [89]
Solibri	[14], [46], [61]–[63], [80]
Drools and Open IFC Tools	[16]
KBIM Assess and KBIM Assess-Lite	[55]

Regarding the works that developed their own systems, three of them did it based on the Visual Basic C# Programming Language; three did not indicate which language was used; two used Java; one used mvdXML; and one used Visual C++ (Figure 16).

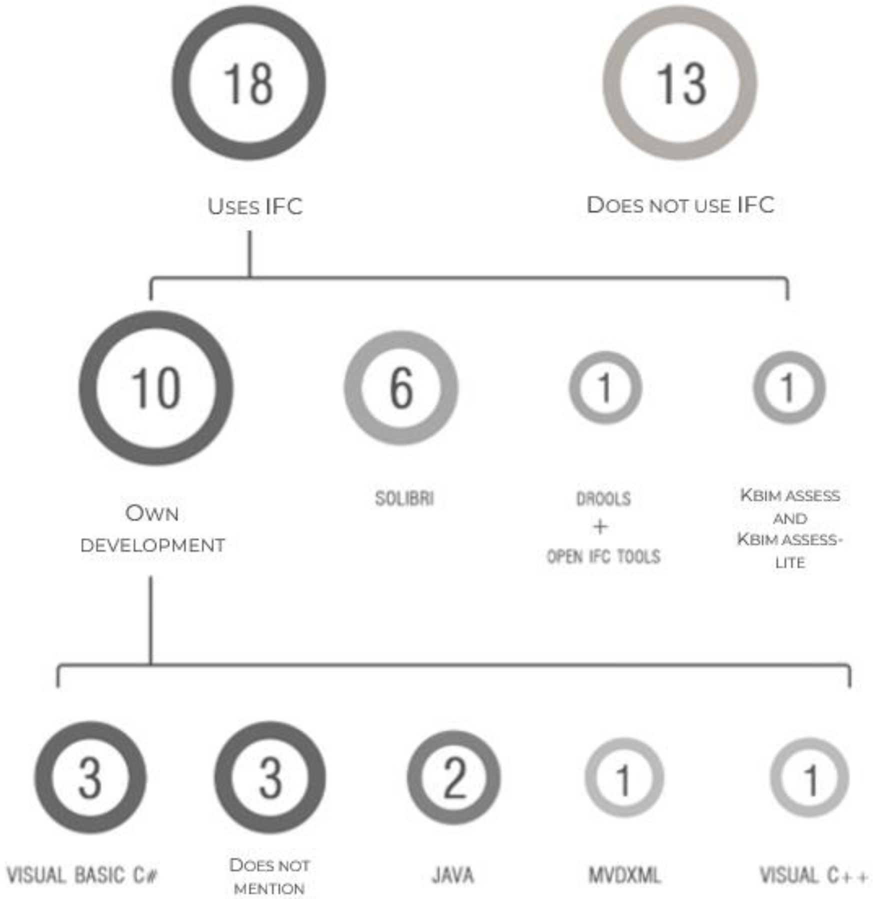

Figure 16: IFC use

Regarding the software used for the compliance procedures, regardless of the use of IFC files, four articles used the Dynamo visual language program; six used the Solibri Model Checker; and 15 articles used their own specific system (Figure 17). The other six articles used other software, indicated in Table 5.

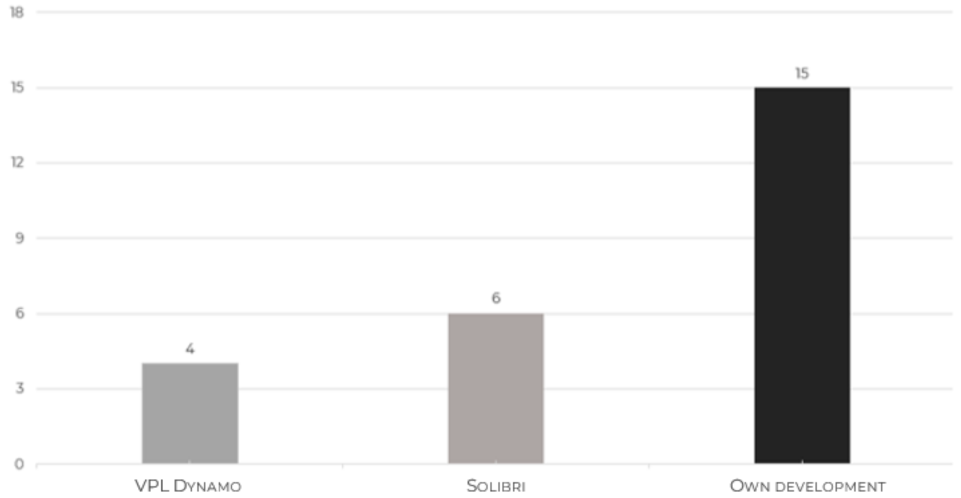

Figure 17: Checking software

Table 5: Rule checking software

Checking Software	Source
Own development	[13], [15], [31], [32], [48], [52], [65], [68], [70], [71], [73], [75], [78], [79], [82], [88]
Solibri	[14], [46], [61]–[63], [80]
VPL Dynamo	[26], [28], [56], [69]
VPL Marionette	[60]
VPL Grasshoper	[74]
Unity 3d	[89]
KBVL	[34]
KBIM Assess and KBIM Assess-Lite	[55]
Drools + Open IFC Tools	[16]

Concerning the difficulties and limitations found in the automatic rule checking processes, issues related only to technology were identified, such as: (i) data processing; (ii) geometry; (iii) scope of the investigation; and (iv) tests (Figure 18, Table 6).

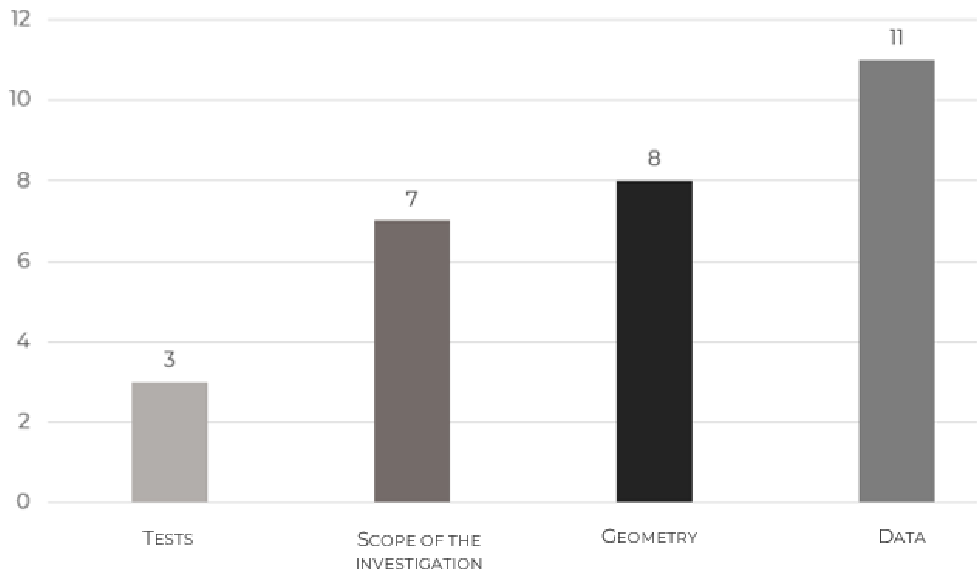

Figure 18: Difficulties found

Table 6: Difficulties found and its source

Difficulties	Source
Tests	[71], [78], [80]
Scope of research	[14], [34], [55], [56], [63], [70], [74]
Geometry	[15], [16], [28], [46], [48], [65], [69], [75]
Data processing	[13], [26], [31], [52], [60], [68], [73], [79], [82], [88], [89]

V. CONCLUSIONS

The presented SR focused on the analysis of research involving automatic rule checking in BIM. This is a process with a high degree of complexity, especially regarding certain stages of its implementation. We noticed that the possibilities of use extend to several sectors that have some type of manual process checking.

Furthermore, we observed that several studies present different strategies for automatic code compliance checking. In various situations, specific application developments for the desired uses were presented. This is due to the difficulty in some rule interpretation processes and logical structuring, especially regarding the transcription from natural language to machine language [13]–[16] and the creation of customized rules within commercial systems [11], [16], [28].

The use of hard-coded mechanisms facilitates the development of specific protocols when pre-programmed functions are used, generating significant time savings. Nevertheless, such mechanisms end up having limited versatility, since they are not editable in the vast majority of cases. Thus, in this situation, users enter the data, activate the checking mechanism, and receive the results; however, they are not able to understand or make modifications to the rules, only to the parameter values.

Some strategies were: implementation of visual programming language with the definition of pre-configured models; creation of assistant Application Program Interfaces (API) for proprietary programs and for commercial checking software; development of logical structuring, with the translation of rules or standards being done automatically, via RASE, PAS, and combinations between approaches; use of *machine learning* and/or deep learning systems for rule translation and for checking consistency of information. The visual programming language reduces the need for knowledge of more advanced programming, by its own intuitive conformation, thus making it more attractive to use in general.

Concerning the steps proposed to perform automatic rule checking, we observed, in some studies, that not all were automated [10]; for example, in the stage of rule interpretation and logical structuring, the transposition of rules from natural language to machine language was performed manually, and in the stage of export of the checking results, the report indicating the items that passed or did not pass the checks was not automatically generated.

The analyses carried out contribute to understand how the development of automatic rule checking systems is progressing. We observed that there is no unified or fully editable and interoperable approach, due to the complexities involved and specific demands. However, regarding the choice of the best approach to be followed, or even which processes to automate, the discussion presented helps in the choice of more efficient systems for future work, minimizing the time consumed in exploratory studies.

We noticed several challenges in the procedures for creating mechanisms that allow automated rule checking of BIM models. Issues related to data processing were the main obstacles found, including difficulties with data entry, inflexibility regarding the rules entered, identification of conflicting information, and non-automated information flow.

Several difficulties were linked to geometry, namely: problems in the treatment of complex geometries (such as curves and convex objects); lack of support for certain typologies of objects (such as ramps); need to simplify the modeling or specific configurations of the model following a certain pattern; and lack of information in the BIM components. Another significant limitation was related to the scope of the investigation, in cases where only specific points of the standard used as a basis were explored, by way of validation of the tool.

No articles were found addressing use cases focused on the specificities of the Brazilian market. This reflects the scarcity of research on the topic and shows that these systems, even if complex, should be studied further by the academia.

REFERENCES

[1] E. Hochscheid and G. Halin, "Micro BIM Adoption in Design Firms: Guidelines for Doing a BIM Implementation Plan," pp. 864–871, 2019, doi: 10.3311/ccc2019-119.
[2] R. Sacks, M. Girolami, and I. Brilakis, "Building Information Modelling, Artificial Intelligence and Construction Tech," Developments in the Built Environment, p. 100011, 2020, doi: https://doi.org/10.1016/j.dibe.2020.100011.
[3] Brasil, Lei de Licitações e Contratos Administrativos. Brasilia, 2021.
[4] DYNAMO BIM, "This is Dynamo," Dynamo Sandbox, 2021. http://dynamobim.org/download/ (accessed Jul. 16, 2022).
[5] R. C. Kater, Marcel; Ruschel, "Avaliando a aplicabilidade de BIM para verificação da norma de segurança contra incêndio em projeto de habitação multifamiliar," XV Encontro Nacional de Tecnologia do Ambiente Construído, no. i, pp. 2821–2831, 2014.
[6] C. Trebbi, M. Cianciulli, F. Matarazzo, C. Mirarchi, G. Cianciulli, and A. Pavan, "Clash Detection and Code Checking BIM Platform for the Italian Market," in Digital Transformation of the Design, Construction and Management Processes of the Built Environment, B. Daniotti, M. Gianinetto, and S. Della Torre, Eds., Cham: Springer International Publishing, 2020, pp. 115–125. doi: 10.1007/978-3-030-33570-0_11.
[7] W. Solihin and C. M. Eastman, "Classification of rules for automated BIM rule checking development," Autom Constr, vol. 53, pp. 69–82, 2015, doi: 10.1016/j.autcon.2015.03.003.
[8] W. Solihin and C. Eastman, "Classification of rules for automated BIM rule checking development," AUTOMATION IN CONSTRUCTION, vol. 53, pp. 69–82, 2015, doi: 10.1016/j.autcon.2015.03.003.
[9] Y. Demir Altintas and M. Emre Ilal, "Integrating building and context information for automated zoning code checking: A review," Journal of Information Technology in Construction, vol. 27, pp. 548–570, 2022.
[10] C. Eastman, J. J. kook J. min J. Lee, Y. suk Jeong, and J. J. kook J. min J. Lee, "Automatic rule-based checking of building designs," Autom Constr, vol. 18, no. 8, pp. 1011–1033, 2009, doi: https://doi.org/10.1016/j.autcon.2009.07.002.
[11] J. Soliman-Junior, C. T. Formoso, and P. Tzortzopoulos, "A semantic-based framework for automated rule checking in healthcare construction projects," CANADIAN JOURNAL OF CIVIL ENGINEERING, vol. 47, no. 2, SI, pp. 202–214, 2020, doi: 10.1139/cjce-2018-0460.
[12] W.-I. W.-I. Lee and N.-C. N.-C. Chiang, "An Investigation of the Methods of Logicalizing the Code-Checking System for Architectural Design Review in New Taipei City," Applied Sciences (Switzerland), vol. 6, no. 12, 2016, doi: 10.3390/app6120407.
[13] J. Farooq, P. Sharma, and R. S. Kumar, "A BIM-based detailed electrical load estimation, costing and code checking," International Journal of Electrical and Computer Engineering, vol. 8, no. 5, pp. 3484–3495, 2018, doi: 10.11591/ijece.v8i5.pp3484-3495.
[14] Md. M. Hossain and S. Ahmed, "Developing an automated safety checking system using BIM: a case study in the Bangladeshi construction industry," International Journal of Construction Management, 2019, doi: 10.1080/15623599.2019.1686833.
[15] N. O. Nawari, "A Generalized Adaptive Framework (GAF) for automating code compliance checking," Buildings, vol. 9, no. 4, 2019, doi: 10.3390/buildings9040086.
[16] K. Schwabe, J. Teizer, and M. König, "Applying rule-based model-checking to construction site layout planning tasks," Autom Constr, vol. 97, no. November 2018, pp. 205–219, 2019, doi: https://doi.org/10.1016/j.autcon.2018.10.012.
[17] M. L. O'Leary, A first course in mathematical logic and set theory. Hoboken, New Jersey : John Wiley & Sons, 2016.
[18] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?," Int J Hum Comput Stud, vol. 43, no. 5, pp. 907–928, 1995, doi: https://doi.org/10.1006/ijhc.1995.1081.
[19] B. Zhong, C. Gan, H. Luo, and X. Xing, "Ontology-based framework for building environmental monitoring and compliance checking under BIM environment," Build Environ, vol. 141, no. December 2017, pp. 127–142, 2018, doi: https://doi.org/10.1016/j.buildenv.2018.05.046.

[20] B. T. Zhong, L. Y. Ding, H. B. Luo, Y. Zhou, Y. Z. Hu, and H. M. Hu, "Ontology-based semantic modeling of regulation constraint for automated construction quality compliance checking," Autom Constr, vol. 28, pp. 58–70, 2012, doi: https://doi.org/10.1016/j.autcon.2012.06.006.
[21] Building Smart International, "Industry Foundation Classes (IFC)," 2022. https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/ (accessed Jul. 09, 2022).
[22] Y.-C. Lee, "Rule Logic And Its Validation Framework Of Model View Definitions For Building Information Modeling," Georgia Institute of Technology, 2015.
[23] Building Smart International, "The importance of Industry Foundation Classes in Building Information Modelling," 2022. https://www.buildingsmart.org/the-importance-of-industry-foundation-classes-in-building-information-modelling/ (accessed Jul. 09, 2022).
[24] Y.-C. C. Lee, W. Solihin, and C. M. Eastman, "The Mechanism and Challenges of Validating a Building Information Model regarding data exchange standards," Autom Constr, vol. 100, no. January, pp. 118–128, 2019, doi: https://doi.org/10.1016/j.autcon.2018.12.025.
[25] Building Smart International, "Model View Definitions (MVD) - An Introduction," Technical Building Smart, 2022. https://technical.buildingsmart.org/standards/ifc/mvd (accessed Jul. 09, 2022).
[26] T. E. Seghier, L. Y. Wah, M. H. Ahmad, and W. O. Samuel, "Building Envelope Thermal Performance Assessment Using Visual Programming and BIM, based on ETTV requirement of Green Mark and GreenRE," INTERNATIONAL JOURNAL OF BUILT ENVIRONMENT AND SUSTAINABILITY, vol. 4, no. 3, pp. 227–235, 2017, doi: 10.11113/ijbes.v4.n3.216.
[27] V. Bazjanac, "Model based cost and energy performance estimation during schematic design," 22nd Conference on Information Technology in Construction, pp. 677–688, 2005.
[28] P. Burggräf, M. Dannapfel, M. Ebade-Esfahani, and F. Scheidler, "Creation of an expert system for design validation in BIM-based factory design through automatic checking of semantic information," Procedia CIRP, vol. 99, pp. 3–8, 2021, doi: https://doi.org/10.1016/j.procir.2021.03.012.
[29] J. Song, J.-K. J.-K. Lee, J. Choi, and I. Kim, "Deep learning-based extraction of predicate-argument structure (PAS) in building design rule sentences," J Comput Des Eng, vol. 7, no. 5, pp. 563–576, 2020, doi: 10.1093/jcde/qwaa046.
[30] D. Guo, E. Onstein, and A. D. La Rosa, "A Semantic Approach for Automated Rule Compliance Checking in Construction Industry," IEEE Access, 2021, doi: 10.1109/ACCESS.2021.3108226.
[31] S. Jiang, Z. Wu, B. Zhang, and H. S. Cha, "Combined MvdXML and Semantic Technologies for Green Construction Code Checking," APPLIED SCIENCES-BASEL, vol. 9, no. 7, 2019, doi: 10.3390/app9071463.
[32] J. Zhang and N. M. El-Gohary, "Integrating semantic NLP and logic reasoning into a unified system for fully-automated code checking," Autom Constr, vol. 73, pp. 45–57, Jan. 2017, doi: https://doi.org/10.1016/j.autcon.2016.08.027.
[33] C. Preidel, A. A. Borrmann, J. Dimyadi, W. Solihin, C. Preidel, and A. A. Borrmann, "Towards code compliance checking on the basis of a visual programming language," Journal of Information Technology in Construction, vol. 21, no. November, pp. 402–421, 2016.
[34] H. Kim, J.-K. Lee, J. Shin, and J. Choi, "Visual language approach to representing KBimCode-based Korea building code sentences for automated rule checking," J Comput Des Eng, vol. 6, no. 2, pp. 143–148, 2019, doi: 10.1016/j.jcde.2018.08.002.
[35] N. Nawari, "Smartcodes and BIM," in Structures Congress 2013: Bridging Your Passion with Your Profession - Proceedings of the 2013 Structures Congress, 2013, pp. 928–937. doi: 10.1061/9780784412848.082.
[36] R. Sacks, T. Bloch, M. Katz, and R. Yosef, "Automating Design Review with Artificial Intelligence and BIM: State of the Art and Research Framework," 2019, pp. 353–360. doi: 10.1061/9780784482421.045.
[37] E. Hjelseth and N. N. Nisbet, "Capturing normative constraints by use of the semantic mark-up RASE methodology," 2011.
[38] J. K. Lee, "Building Environment Rule and Analysis (BERA) Language," Georgia Institute of Technology, 2011.
[39] W. Solihin, "A simplified BIM data representation using a relational databse schema for an efficient rule checking system and its associated rule checking language," Georgia Institute of Technology, 2016.
[40] R. Sacks, L. Ma, R. Yosef, A. Borrmann, S. Daum, and U. Kattel, "Semantic Enrichment for Building Information Modeling: Procedure for Compiling Inference Rules and Operators for Complex Geometry," Journal of Computing in Civil Engineering, vol. 31, no. 6, p. 04017062, 2017, doi: doi:10.1061/(ASCE)CP.1943-5487.0000705.
[41] H. Nacer and D. Aissani, "Semantic web services: Standards, applications, challenges and solutions," Journal of Network and Computer Applications, vol. 44, pp. 134–151, 2014, doi: https://doi.org/10.1016/j.jnca.2014.04.015.

[42] D. Guo, E. Onstein, and A. D. La Rosa, "An Approach of Automatic SPARQL Generation for BIM Data Extraction," APPLIED SCIENCES-BASEL, vol. 10, no. 24, 2020, doi: 10.3390/app10248794.
[43] M. I. W. M. Morandi and L. F. R. Camargo, "Revisão sistemática da literatura," in Design Science Research: Método de pesquisa para avanço da ciência e tecnologia, São Paulo: Bookman, 2015, p. 204.
[44] S. Park, Y.-C. Lee, and J.-K. Lee, "DEFINITION OF A DOMAIN-SPECIFIC LANGUAGE FOR KOREAN BUILDING ACT SENTENCES AS AN EXPLICIT COMPUTABLE FORM," JOURNAL OF INFORMATION TECHNOLOGY IN CONSTRUCTION, vol. 21, pp. 424–435, 2016.
[45] P. Pauwels et al., "A performance benchmark over semantic rule checking approaches in construction industry," ADVANCED ENGINEERING INFORMATICS, vol. 33, pp. 68–88, 2017, doi: 10.1016/j.aei.2017.05.001.
[46] Y. Ji and F. Leite, "Automated tower crane planning: leveraging 4-dimensional BIM and rule-based checking," Autom Constr, vol. 93, pp. 78–90, 2018, doi: 10.1016/j.autcon.2018.05.003.
[47] Y.-C. Lee, C. M. Eastman, and W. Solihin, "Logic for ensuring the data exchange integrity of building information models," Autom Constr, vol. 85, pp. 249–262, 2018, doi: https://doi.org/10.1016/j.autcon.2017.08.010.
[48] S.-L. L. Fan, H.-L. L. Chi, and P.-Q. Q. Pan, "Rule checking Interface development between building information model and end user," Autom Constr, vol. 105, no. May, p. 102842, 2019, doi: 10.1016/j.autcon.2019.102842.
[49] R. K. Soman, M. Molina-Solana, and J. K. Whyte, "Linked-Data based Constraint-Checking (LDCC) to support look-ahead planning in construction," Autom Constr, vol. 120, p. 103369, 2020, doi: https://doi.org/10.1016/j.autcon.2020.103369.
[50] J. Zhang and N. M. El-Gohary, "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques," JOURNAL OF COMPUTING IN CIVIL ENGINEERING, vol. 30, no. 5, 2016, doi: 10.1061/(ASCE)CP.1943-5487.0000536.
[51] B. Zhong, H. Wu, L. Ding, H. Luo, Y. Luo, and X. Pan, "Hyperledger fabric-based consortium blockchain for construction quality information management," FRONTIERS OF ENGINEERING MANAGEMENT, vol. 7, no. 4, pp. 512–527, 2020, doi: 10.1007/s42524-020-0128-y.
[52] Y. D. Altintas and M. E. Ilal, "Loose coupling of GIS and BIM data models for automated compliance checking against zoning codes," Autom Constr, vol. 128, 2021, doi: 10.1016/j.autcon.2021.103743.
[53] R. Amor and J. Dimyadi, "The promise of automated compliance checking," Developments in the Built Environment, vol. 5, p. 100039, 2021, doi: https://doi.org/10.1016/j.dibe.2020.100039.
[54] P. Zhou and N. El-Gohary, "Semantic information alignment of BIMs to computer-interpretable regulations using ontologies and deep learning," Advanced Engineering Informatics, vol. 48, p. 101239, 2021, doi: https://doi.org/10.1016/j.aei.2020.101239.
[55] I. Kim, J. Choi, E. A. L. Teo, and H. Sun, "Development of kbim e-submission prototypical system for the openbim-based building permit framework," Journal of Civil Engineering and Management, vol. 26, no. 8, pp. 744–756, 2020, doi: 10.3846/jcem.2020.13756.
[56] K. Kincelova, C. Boton, P. Blanchet, and C. Dagenais, "Fire Safety in Tall Timber Building: A BIM-Based Automated Code-Checking Approach," BUILDINGS, vol. 10, no. 7, Jul. 2020, doi: 10.3390/buildings10070121.
[57] T. Bloch and R. Sacks, "Comparing machine learning and rule-based inferencing for semantic enrichment of BIM models," Autom Constr, vol. 91, no. July 2017, pp. 256–272, 2018, doi: 10.1016/j.autcon.2018.03.018.
[58] J. Dimyadi, W. Solihin, W. Solihin, and C. Eastman, "A knowledge representation approach in BIM rule requirement analysis using the conceptual graph," Journal of Information Technology in Construction, vol. 21, pp. 370–402, 2016.
[59] T. Bloch and R. Sacks, "Clustering Information Types for Semantic Enrichment of Building Information Models to Support Automated Code Compliance Checking," Journal of Computing in Civil Engineering, vol. 34, no. 6, p. 04020040, 2020, doi: 10.1061/(asce)cp.1943-5487.0000922.
[60] P. Ghannad, Y.-C. Lee, J. Dimyadi, and W. Solihin, "Automated BIM data validation integrating open-standard schema with visual programming language," Advanced Engineering Informatics, vol. 40, pp. 14–28, 2019, doi: https://doi.org/10.1016/j.aei.2019.01.006.
[61] A. L. C. Ciribini, S. Mastrolembo Ventura, and M. Paneroni, "Implementation of an interoperable process to optimise design and construction phases of a residential building: A BIM Pilot Project," Autom Constr, vol. 71, pp. 62–73, 2016, doi: 10.1016/j.autcon.2016.03.005.
[62] E. Türkyılmaz, "A Method to Analyze the Living Spaces of Wheelchair Users Using IFC," Procedia Soc Behav Sci, vol. 222, pp. 458–464, 2016, doi: https://doi.org/10.1016/j.sbspro.2016.05.136.
[63] J. Choi and I. Kim, "Development of rule-based building code compliance checking system for BIM-based quality improvement," Information (Japan), vol. 20, no. 4, pp. 2929–2936, 2017.

[64] S. Beazley, E. Heffernan, and T. J. McCarthy, “Enhancing energy efficiency in residential buildings through the use of BIM: The case for embedding parameters during design,” Energy Procedia, vol. 121, pp. 57–64, 2017, doi: https://doi.org/10.1016/j.egypro.2017.07.479.
[65] C. Sydora and E. Stroulia, “Rule-based compliance checking and generative design for building interiors using BIM,” Autom Constr, vol. 120, no. August, p. 103368, 2020, doi: 10.1016/j.autcon.2020.103368.
[66] R. Takim, M. H. Zulkifli, and A. H. Nawawi, “Integration of Automated Safety Rule Checking (ASRC) System for Safety Planning BIM-Based Projects in Malaysia,” Procedia Soc Behav Sci, vol. 222, pp. 103–110, 2016, doi: https://doi.org/10.1016/j.sbspro.2016.05.195.
[67] M. H. Zulkifli, R. Takim, and A. H. Nawawi, “A proposed initial framework of ASRC system for BIM-based projects in Malaysia,” J Teknol, vol. 78, no. 5–2, pp. 61–67, 2016, doi: 10.11113/jt.v78.8491.
[68] P. C. Lee, T.-P. Lo, M.-Y. Tian, and D. Long, “An Efficient Design Support System based on Automatic Rule Checking and Case-based Reasoning,” KSCE Journal of Civil Engineering, 2019, doi: 10.1007/s12205-019-1750-2.
[69] T. E. Seghier, M. H. Ahmad, and Y.-W. Lim, “Automation of Concrete Usage Index (CUI) assessment using computational BIM,” INTERNATIONAL JOURNAL OF BUILT ENVIRONMENT AND SUSTAINABILITY, vol. 6, no. 1, pp. 23–30, Jan. 2019, doi: 10.11113/ijbes.v6.n1.319.
[70] S. M. Ilal and H. M. Gunaydin, “Computer representation of building codes for automated compliance checking,” Autom Constr, vol. 82, pp. 43–58, 2017, doi: 10.1016/j.autcon.2017.06.018.
[71] J. Dimyadi, C. Clifton, M. Spearpoint, and R. Amor, “Computerizing Regulatory Knowledge for Building Engineering Design,” JOURNAL OF COMPUTING IN CIVIL ENGINEERING, vol. 30, no. 5, 2016, doi: 10.1061/(ASCE)CP.1943-5487.0000572.
[72] T. H. Beach, J.-L. Hippolyte, and Y. Rezgui, “Towards the adoption of automated regulatory compliance checking in the built environment,” Autom Constr, vol. 118, 2020, doi: 10.1016/j.autcon.2020.103285.
[73] J. Yuan, X. Li, X. Xiahou, N. Tymvios, Z. Zhou, and Q. Li, “Accident prevention through design (PtD): Integration of building information modeling and PtD knowledge base,” Autom Constr, vol. 102, pp. 86–104, Jun. 2019, doi: 10.1016/j.autcon.2019.02.015.
[74] N. Khan, A. K. Ali, M. J. Skibniewski, D. Y. Lee, and C. Park, “Excavation Safety Modeling Approach Using BIM and VPL,” Advances in Civil Engineering, vol. 2019, 2019, doi: 10.1155/2019/1515808.
[75] H. Ying and S. Lee, “A rule-based system to automatically validate IFC second-level space boundaries for building energy analysis,” Autom Constr, vol. 127, no. March, p. 103724, Jul. 2021, doi: https://doi.org/10.1016/j.autcon.2021.103724.
[76] D. R. Warren and N. Saleeb, “Improving Productivity by the Automation of Checking of 3D Parametric Modelling,” International Journal of Safety and Security Engineering, vol. 10, no. 4, pp. 441–450, 2020, doi: 10.18280/ijsse.100402.
[77] K. Shahi, B. Y. McCabe, and A. Shahi, “Framework for Automated Model-Based e-Permitting System for Municipal Jurisdictions,” Journal of Management in Engineering, vol. 35, no. 6, 2019, doi: 10.1061/(ASCE)ME.1943-5479.0000712.
[78] T. Kang, “Rule-Based LEED Evaluation Method considering BIM Linkage and Variability,” KSCE Journal of Civil Engineering, vol. 24, no. 1, pp. 110–121, 2020, doi: 10.1007/s12205-020-1026-x.
[79] H. Lai, X. Deng, H. Chen, and P. Shi, “Implementation of building information modeling standards for the Shanghai Metro, China,” PROCEEDINGS OF THE INSTITUTION OF CIVIL ENGINEERS-MUNICIPAL ENGINEER, vol. 173, no. 3, pp. 171–184, 2020, doi: 10.1680/jmuen.18.00040.
[80] V. Getuli, S. M. Ventura, P. Capone, and A. L. C. Ciribini, “BIM-based Code Checking for Construction Health and Safety,” Procedia Eng, vol. 196, no. June, pp. 454–461, 2017, doi: 10.1016/j.proeng.2017.07.224.
[81] Y. C. Lee, C. M. Eastman, W. Solihin, and R. See, “Modularized rule-based validation of a BIM model pertaining to model views,” Autom Constr, vol. 63, pp. 1–11, 2016, doi: 10.1016/j.autcon.2015.11.006.
[82] S. Li, H. Cai, and V. R. Kamat, “Integrating Natural Language Processing and Spatial Reasoning for Utility Compliance Checking,” J Constr Eng Manag, vol. 142, no. 12, 2016, doi: 10.1061/(ASCE)CO.1943-7862.0001199.
[83] C. Preidel, S. Daum, and A. Borrmann, “Data retrieval from building information models based on visual programming,” Visualization in Engineering, vol. 5, no. 1, 2017, doi: 10.1186/s40327-017-0055-0.
[84] B. Koo and B. Shin, “Applying novelty detection to identify model element to IFC class misclassifications on architectural and infrastructure Building Information Models,” J Comput Des Eng, vol. 5, no. 4, pp. 391–400, 2018, doi: https://doi.org/10.1016/j.jcde.2018.03.002.
[85] H. Lee, J. K. Lee, S. Park, and I. Kim, “Translating building legislation into a computer-executable format for evaluating building permit requirements,” Autom Constr, vol. 71, pp. 49–61, 2016, doi: 10.1016/j.autcon.2016.04.008.

[86] W. Solihin, J. Dimyadi, Y.-C. C. Lee, C. M. Eastman, and R. Amor, "Simplified schema queries for supporting BIM-based rule-checking applications," Autom Constr, vol. 117, no. May, p. 103248, 2020, doi: https://doi.org/10.1016/j.autcon.2020.103248.
[87] L. Young-Cheol, C. M. Eastman, and W. Solihin, "Rules and validation processes for interoperable BIM data exchange," J Comput Des Eng, vol. 8, no. 1, pp. 97–114, 2021, doi: 10.1093/jcde/qwaa064.
[88] A. A. Akanmu, J. Olayiwola, and O. Alfred Olatunji, "Automated checking of building component accessibility for maintenance," Autom Constr, vol. 114, 2020, doi: 10.1016/j.autcon.2020.103196.
[89] G. Hongling, Y. Yantao, Z. Weisheng, and L. Yan, "BIM and Safety Rules Based Automated Identification of Unsafe Design Factors in Construction," Procedia Eng, vol. 164, no. June, pp. 467–472, 2016, doi: https://doi.org/10.1016/j.proeng.2016.11.646.

AUTHORS

Cauê C. Santiago

Bachelor's degree in civil engineering from the Pontifical Catholic University of Minas Gerais (2016), possesses expertise in Civil Engineering, particularly in Reinforced Concrete and Metallic Structures, as well as BIM projects and processes. He holds a specialization in Master BIM from IPOG and a master's degree in architecture and technology from the State University of Campinas (UNICAMP), with a focus on Building Information Modeling (BIM) and an emphasis on automated rule checking.

Ana Regina Mizrahy Cuperschmid

Associate Professor at FECFAU-UNICAMP, engaged in both undergraduate and graduate education. Holds postdoctoral degrees from UNICAMP and USP, and earned a Doctorate and Master's degree from PPGATC-UNICAMP. Graduated in Architecture and Urbanism from UFV. Specializes in Project Management, with a focus on digital technologies in architecture. Leads research and publications in BIM, HBIM, Virtual and Augmented Reality. Currently serves as the Chair of the Department of Architecture and Construction..