



AKADÉMIAI KIADÓ




International Review of
Applied Sciences and
Engineering

DOI:
10.1556/1848.2024.00833
© 2024 The Author(s)

ORIGINAL RESEARCH
PAPER



BIM based collision tracking at the intersections of different building engineering systems at the design stage

Sergey Pogorelskiy¹ and Imre Kocsis^{2*} 

¹ Doctoral School of Informatics, University of Debrecen, Hungary

² Department of Basic Technical Studies, Faculty of Engineering, University of Debrecen, Hungary

Received: March 3, 2024 • Accepted: July 1, 2024

ABSTRACT

Reduction of design errors, minimisation of rework and the improvement of the design productivity are key factors in building engineering systems (including structural and architectural solutions, ventilation systems, sewerage systems, water supply and heating systems, power supply systems, and communication networks). These goals can be achieved with a complex approach that prioritises the design of different building engineering systems in the model during the design phase, in order to provide a consistent design for different building engineering systems. The paper presents a novel approach for the application of plugins in building service systems with the elimination of collision in the focus. Collision reduction actions in this methodology are categorised into three levels: the code level, which pertains to plugin developers; the algorithm level, which relates to BIM coordinators; and the user level, which concerns engineers performing the check. This new systematic approach to collision resolution prioritises maintaining the consistency of collision detection across different systems and storing all information about each collision. Collision checking is based on several key factors, such as complying with the sequence of checking systems, excluding irrelevant collisions, and setting tolerances when joining system elements. The aim of our approach is to automate and expedite not only the identification of the intersections but also the subsequent work with it throughout the entire project life cycle. The results are demonstrated by a case study conducted in the frame of a real project.

KEYWORDS

building information modelling, building engineering system, data center, clashes management

1. INTRODUCTION

In this paper a new methodology for supporting the building system engineering design is presented. An algorithm is proposed for planning specialists' actions using information modelling technologies and includes the tools of engineering networks modelling. The main desired benefits of the algorithm is the consistency of design of different engineering systems and the improved efficiency by the systematic elimination of redundancies in the system development. As a result, a significant reduction in the number of collisions is expected.

The method of plug-in development is in the focus of our investigations aiming at the optimisation of the complex engineering subsystems at the design stage. To show the key ideas in our approach, the collision checking procedure is discussed at three levels: code level (plug-in developer), algorithm level (BIM coordinator) and user level (engineers performing the check).

Applications and the practical results of our approach, namely the reduced time needed for the identification of intersections and their correction and excluding duplicated intersections and efficient report generation on the intersections of the different engineering systems, are demonstrated through a case study discussed in Section 4.

*Corresponding author.
E-mail: kocsisi@eng.unideb.hu



When designing engineering systems, BIM utilises a shared digital information repository in the form of a three-dimensional computer model of the construction object. This model contains all the necessary information about the object and is created by specialists of different profiles, designing different systems simultaneously in one information environment. Therefore, with an appropriate management of the communication established between the process participants (real-time information exchange about actions using a unified structure for data storage and transfer) the rework (corrections) can be reduced to the minimum and the potential system overlaps at the start of data center design can be avoided.

Some potentials of BIM modelling used in our approach (to simplify the work of specialists, reduce the likelihood of design errors, and improve the quality control of documentation for designed objects):

- enables collaboration among specialists of different profiles in a single information environment, regardless of their location;
- allows specialists from different areas to monitor changes in the information model and receive notifications about them;
- includes tools for automated marking of modeled objects;
- generates specifications for modeled equipment and elements automatically; and in general allows for the minimisation of errors during the engineering systems design stage, preventing collisions between designed objects.

Additionally, a three-dimensional digital information model allows for clear tracing of the location of system elements in space and prevents their intersection.

The nature of collision formation has different background. Chahrour et al. [1] state that collisions can occur due to design rule violations, errors, uncertainty, model inaccuracy, and tolerance. Collision detection using BIM tools distinguishes between two types of collisions: relevant and irrelevant. If collisions result in a loss of productivity, interruptions, and rework, they are considered relevant to the project [2]. Relevant collisions are those that require resolution. In contrast, irrelevant collisions do not require resolution as they may be a single error repeated multiple times throughout the project or collisions that were deliberately created.

Collisions occur when elements in different ranges collide with each other. The research findings in [3] indicate that collisions are most likely to occur within the range of 30–299 mm, with the discrete category of 100–199 mm being the most common. It is important to note that these findings are based on predictive modeling. Therefore, implementing specific tolerances, tailored to the type of system, can prevent a significant number of unnecessary collisions.

Proper identification of relevant and irrelevant collisions and appropriate distribution of collision resolution among BIM project teams play an important role in the design process, where thousands of collisions can be captured using specialised software [2]. Estimating the time required for an

individual specialist to resolve a conflict is challenging due to the varying nature of intersecting systems and their crowding at intersections. According to the experience of the authors, it can be assumed that eliminating an average collision takes several minutes, therefore, automated collisions elimination results in significant savings in engineers' work. In the case study, the new methodology can save up to 35–40 h in one project (see Section 4).

According to Adamu et al. [4] causes of collisions can be classified as follows: use of incorrect or low level of detail [5]; uncertainty in design/use of “fillers” [6]; failure to follow design rules [6]; accuracy vs. timing [6]; 3D model objects that exceed acceptable clearances [6]; designers working in isolation from each other [7–9]; complexity of the design [6, 10, 11]; lack of time [11, 12]; use of 2D rather than 3D models [5, 13, 14]; design errors [6, 15, 16]; use of different file formats [17]; lack of experts [5, 11, 17–20].

Our goal is to detect and prevent collisions at early design stages, and eventually to minimise the resources require by the revision of the executive model, which forms the basis of the design documentation. The executive information model is a consolidated model created by engineers from different disciplines who have made changes to their respective sections of the information model. A result of our research is the development of a plug-in that serves as a basis for the optimisation of the design process in all modelled engineering subsystems. The new approach allows to exclude the duplicated intersections and track the history of each of them, in addition, it enables an optimal intersection reports generation. According to the case study, this method can result ten times faster of intersections identification compared to the generally used ones.

2. LITERATURE REVIEW

The majority of the sources in the literature pertaining to the subject of identifying and working with collisions are more concerned with collision detection algorithms. Also, as noted by Pärn et al. [3], despite the benefits that machine learning has brought to the AECO sector, such as automating rule checking within BIM and controlling 4D BIM, clash detection still requires significant time and labor intervention. The current process is still mechanistic, requiring a BIM manager to manually scan and analyse each clash. According to interviews conducted by Lin and Huang [21] with senior project managers, clash reports were often selectively reviewed or completely neglected by BIM managers in Taiwan due to the significant time required to evaluate them. Additionally, Pärn et al. [3] emphasised the pressing need for automated methods that eliminate manual intervention and enable the creation of clash detection profiles.

Various tools are now available to detect collisions between objects in BIM, with most utilising the built-in functionality of Revit and NavisWorks. The literature was analysed to identify the main shortcomings of existing methods for identifying collisions in various ways. For instance, Nursultan B. et al. [22] utilised NavisWorks software



to examine potential losses resulting from merging current project documentation into BIM models, using one of the Nur-Sultan projects as an example. The article presents a comparative analysis of identifying intersections using the combinatorial principle. The section on constructive and spatial planning solutions is checked for any intersections with the sections on heating and ventilation, water supply, and sewerage. This process is then repeated. Exceptional combinations are selected to check for conflicts. The permissibility of deviations for various combinations of the main sections of the design documentation was considered during the conflict analysis. Some of these deviations were reviewed and it was noted that the conflict identification process is still not completely independent and needs to be improved. To prevent errors, it is recommended to develop the virtual model one section at a time rather than in parallel. This will allow the designer to identify any inconsistencies before moving on to the next section of the design documentation. By following this methodology, disputes can be avoided, rework costs can be reduced, and labour costs can be saved.

Kermanshahi et al. [23] also examine a physical object that includes BIM plans for architecture, structure, mechanics, electrical, and plumbing, as well as BIM models for plumbing (MEP) and their collision detection. The study employed Autodesk Revit and Autodesk Navisworks Manage as BIM tools to develop features for simplified and automatic collision detection. The collision detection system identifies three main types of collisions: hard collisions, which occur when a building component physically penetrates another; soft/clean collisions, which occur when components are located too close to each other; and time/4D conflicts, which may involve contractor schedules, equipment and materials delivery, and general time conflicts. The aim of this study is to assess the influence of BIM on the design phase of the construction process and its ability to identify all clashes in construction plans during this phase.

A similar approach is also taken by the authors of another paper [2]. The paper suggests utilising network analysis to enhance collision detection from a comprehensive viewpoint. As a building is a continuous entity, the effects of collisions are influenced by the dependency relationships between building components. To represent component dependencies, a building component network is constructed, which is centered on collision objects. This work defines three types of spatial dependence for building a network: collision, impact, and connection. The collision relationships are classified into four types: intersection, penetration, through, and containment, based on the intersection curves of collision objects. To improve generality, the Industry Foundation Classes (IFC) and Boundary Volume Hierarchy (BVH) models are used to query geometric information. Hierarchical structures, such as BVH, are utilised to expedite the query process by eliminating irrelevant comparisons. The paper investigates the effectiveness of the network method in improving collision detection by means of a real-life project. The results confirm that the method is successful in identifying irrelevant collisions in four

scenarios and reduces the number of such collisions by 17%. Additionally, the network is used to automatically group relevant collisions, leading to a reduction of over 50% of the originally reported collisions.

In addition to the standard Autodesk Navisworks tools, the article [24] suggests using a plug-in to automatically improve the collision detection process. Collisions detected by BIM tools are automatically grouped to prioritise resolution and identify irrelevant collisions. The study showed that relevant and irrelevant collisions can be identified using the proposed plugin.

According to the authors of a study [25], the BIM Vision IFC viewer, an application developed in Poland, is recommended. The purpose of this manuscript is to discuss the collision detection process and present three case studies based on different BIM models, both monodisciplinary and multidisciplinary. Different methodologies, which depend on the assumed process accuracy parameters, produced varying results in terms of the number of collisions reported. The study reports a range of collision numbers, from 0 to several thousand, depending on model complexity and expected accuracy. The study also measured the duration of individual collision detection, with results ranging from less than 1 s for the residential building model to 1 min and 20 s for the industrial structure model.

The authors of the study 'Clash Detection and Code Checking BIM Platform for the Italian Market' [26], conducted research in Italy and suggest using a different software product for checking interference, namely us BIM.clash, instead of ACCA's software system. The plugin can identify hard collisions, hard tolerance collisions, and clearance collisions in an IFC model or between models. The program uses computational geometry algorithms to check each element selected one by one. The system processes a list for each element considered using special data structures. This list contains the remaining model elements that could cause a collision. This brief analysis takes into account the selected test parameters. For instance, to determine if two objects collide, the corresponding bounding boxes are first calculated and checked for disjointedness before a more precise but computationally expensive test is conducted. To enhance performance, the algorithms perform similar calculations in multiple stages for all three types. If the test fails, it indicates that the two elements are not interfering. Therefore, you can remove the element from the list and continue with the next one.

When detecting collisions, it is important to consider methods for weighing their significance and correctly classifying a particular intersection. The authors of the study [27] aim to prioritise collision resolution before the construction phase. In this study, results from Autodesk Navisworks were used to improve the collision detection process. Additionally, this study employs a fuzzy AHP algorithm to assign weights to criteria. The resulting relationships provide a basis for prioritising collisions for resolution and identifying irrelevant collisions. The proposed method was tested on a real project, and a comparison with expert opinions showed that it effectively identifies



important and unimportant collisions. This study presents a logical and practical relationship to improve the collision detection process by using the weight and penetration degree of collision elements.

A similar approach is taken by the authors of 'Clash Relevance Prediction Based on Machine Learning' [28]. The accuracy of BIM-enabled collision detection has been questioned because its output includes many minor collisions that do not have a significant impact on the project or that can be resolved in subsequent design or construction phases. To enhance collision detection quality, this paper utilises supervised machine learning algorithms to differentiate between relevant and irrelevant collisions. Six types of classifiers were in comparison in a test case: J48-based decision tree, random forest, Jrip-based rule methods, binary logistic regression, naïve Bayes, and Bayesian network. The results of cross-validation and the Kruskal-Wallis test showed that Jrip was the most effective method for predicting clash relevance. The prediction accuracy averaged 80% in both the training and test datasets, surpassing the initial percentage of relevant clashes (59%) in the clash report suggested by BIM coordinators for the test case. This indicates the success of the proposed method.

3. MATERIAL AND METHODS

In contrast with most research available in the literature focusing on collision detection algorithms, our aim is to develop a systematic approach for prioritising sequences of modelled systems to minimise early collisions during the design checking stage. For this purpose, using the way of prioritising the design stages of various engineering systems published in [29], an algorithm for the planning of specialist actions when working on a project using information modelling technologies is proposed by the authors, which eliminates collisions in the BIM model.

3.1. Collision checking methodology

One issue with collision checking is that not all intersections in the model necessarily indicate collisions. Quite often pipe connections with fittings or equipment may be displaced, also when working together elements may be duplicated in case the engineer has set to track the change of elements of the adjacent section, or the accounting of equipment service areas may be made in the form of three-dimensional objects, however, hitting, model elements in the peripheral part of such an area, in fact not carrying global problems, will also be defined as a collision. In cases such as the crossing of flexible ducts, it is more efficient to separate them during installation rather than correcting them during the design stage. A significant number of collisions fall within the specified tolerances. These situations, among others, lead to an increase in the number of collisions.

To solve this problem, it is recommended to filter objects before generating a report. Additionally, collisions in the already formed report can be addressed by assigning tolerances

to account for them. Autodesk Navisworks tools may not allow for these tolerances. It is also important to check for intersections by dividing the object into zones or sections provided in the project. The decision on which elements to filter should be made on a project-by-project basis.

Clash detection in BIM tools differentiates between two types of clashes: relevant and irrelevant.

Relevant clashes are those that require resolution because they result in lost productivity, disruption and reworking. Irrelevant clashes, on the other hand, do not require resolution, as they may be a single error repeated many times throughout the project, or clashes that have been deliberately created.

Note that when generating a report on intersections in Navisworks, the table should list the results of the x-th object with the y-th object, and vice versa. As a result, the same collision will appear twice in the report, but in different parts of it. Simultaneously, when one object intersects with multiple objects, such as a duct intersecting with several pipes, the number of collisions naturally increases significantly.

Another issue is the tolerances required to fit different elements together. For instance, a flexible pipe with a cable connected to a power distribution box is a technically correct solution. However, in the model, it may cause a conflict. Therefore, it is crucial to set the correct tolerances when checking the joining of system elements.

Therefore, it is advisable to group the intersections by combining lines with identical elements before sending the report to the engineers for collision correction. In this case, the report automatically reduces the number of collisions.

Another significant drawback of working on collisions in Navisworks is the correct display of the intersection area of objects. The automatically generated view often lacks the correct trimming area, which creates inconvenience for further consideration and work on the collision.

In order to eliminate the identified shortcomings, a plugin was developed (in C#) and tested by BIM experts and engineers of various systems at the design stage.

The methodology proposed in our article for checking collisions is based on several key factors. These include compliance with the sequence of checking systems, exclusion of irrelevant collisions, and setting tolerances when joining system elements.

In our methodology for collision checking, we have introduced three levels: code level (plugin developer), algorithm level (BIM coordinator) and user level (engineers performing the check). At the developer level, forms are developed for data visualisation and code for the logic of user interaction with the database. The work of the principal blocks of the plugin at the code level can be visualised by means of the diagram shown in Fig. 1.

The principal data blocks are the TBD_Clash table in the database, which stores information about all the unique collisions that have ever occurred in the project. A unique collision is defined as the combination of e11_id, e11_file, e12_id, e12_file that is unique.

The ClashResult form displays current collisions related to the current open model at the current time. Collisions



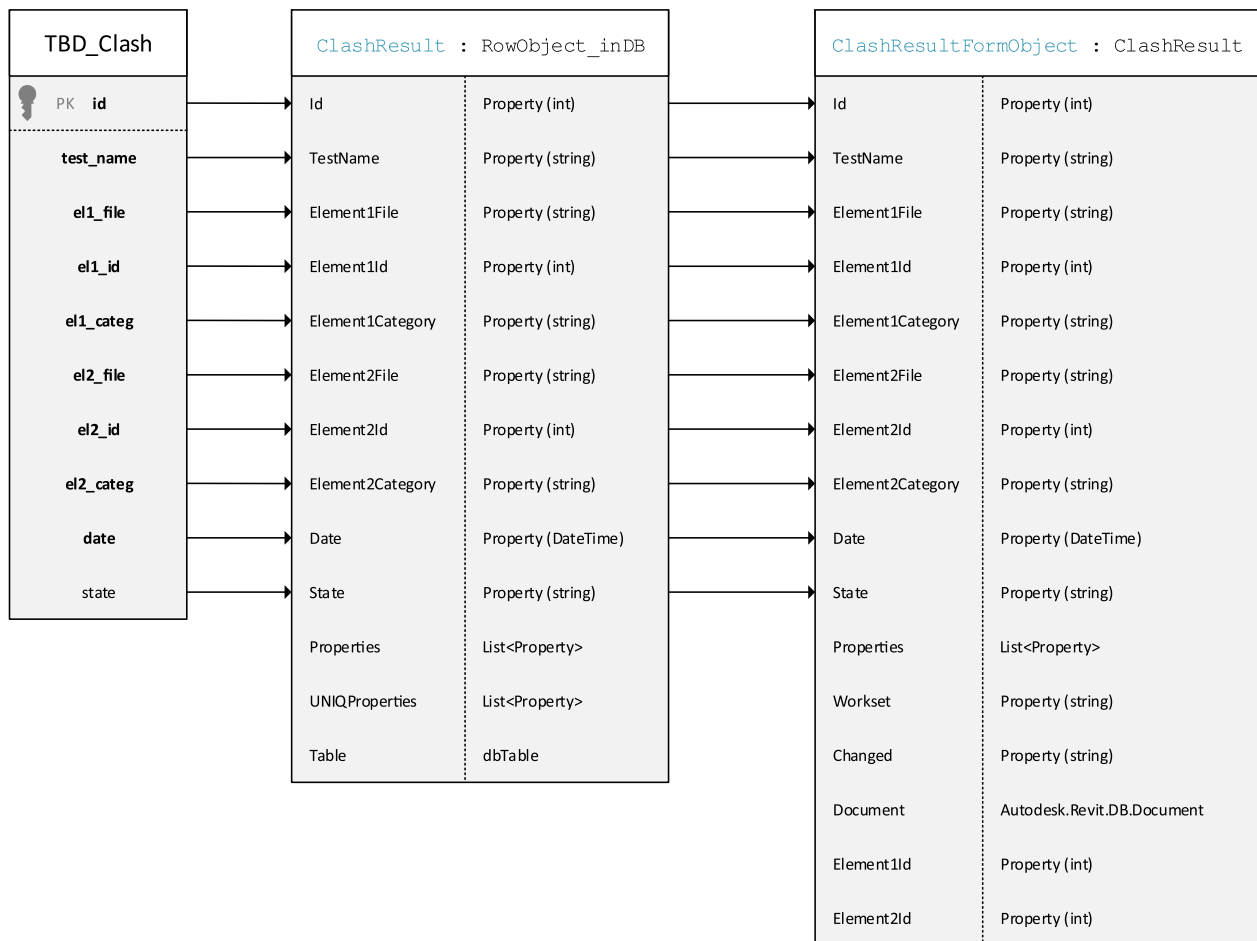


Fig. 1. Code level, diagram of the principal plug-in blocks

with the most recent date are considered current. Upon loading the subsequent collision report, the date at which the file was saved is employed as the date. In the event that a collision already exists within the table, the date of the report being read is substituted for that of the collision.

The collision class contains two constructors. The first constructor is used when reading an XML collision report for subsequent writing to the database. It accepts XAttribute, XElement and date (as a string) as input.

The DataRow object represents the outcome of a database query, and its constructor is employed to generate a set of collision objects for display in a given form.

At the algorithmic level, the function of the plugin can be represented by the following sequence of actions:

- Assignment by the BIM coordinator of the sequence of checking various systems and tolerances for intersection, docking;
- Generation of a collision report in XML format;
- Loading the report into the plugin.

At the user level, the user directly works with the form for searching and visualising collisions and eliminating single collisions. The user's task is to check collisions in a certain sequence and eliminate them step by step. After

elimination, an identifier is placed in the form of a checkmark to enter information into the database that this or that unique collision has been eliminated, information about the date of elimination and the name of the user who last performed the action.

The aforementioned levels (code, algorithm, and user) are inextricably linked and collectively serve as the foundation for the plugin's functionality.

3.2. Plug-in information model

The plug-in menu is a panel embedded in the Revit window, consisting of buttons available to users of different levels. Figure 2 shows the block diagram of the plug-in.

The sequence of its plug-in can be broken down into the following basic steps:

- The engineer selects the model/system files required for comparison, e.g. HVAC and walls;
- The plug-in detects all intersections of the previously selected systems within a given accuracy tolerance;
- The engineer sequentially goes through the list of collisions, noting the collision status (checked or not). The plug-in also provides the possibility to display a 3D view of the required collision.



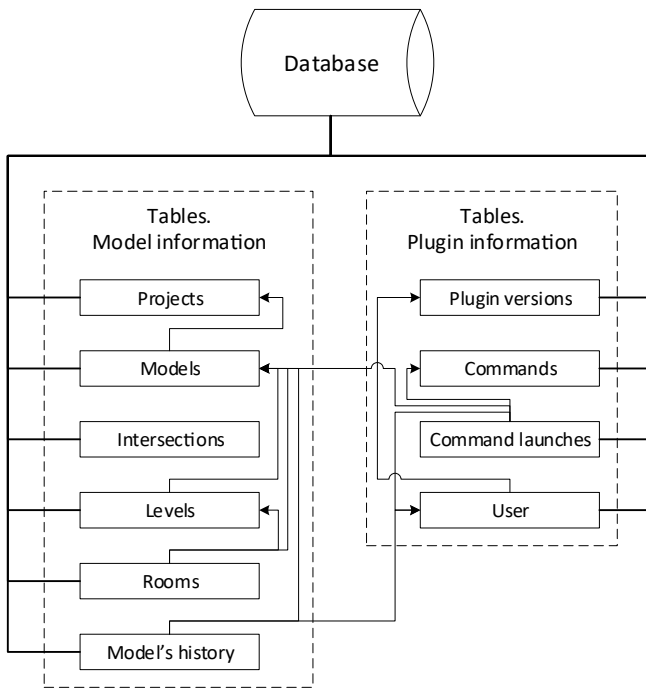


Fig. 2. Block diagram of the plug-in

An important point is that each collision has its own unique ID, which allows to accurately identify it in the database. Also, the speed of displaying the necessary collision on the screen, relative to the standard capabilities of Revit is increased by 20 times (from 20 s to 1 s per operation), which is a significant plus in speeding up the work of engineers.

3.3. Plug-in databases

The information model of the plug-in can be represented as a diagram containing databases and shown in Fig. 3.

The description of each database is given in Table 1.

4. CASE STUDY

The system in the case study included several floors with telecommunication rooms, with more than 1,000 racks. In addition to the architectural and structural part, the project included such systems as HVAC, sewerage pipelines, power supply, security and fire-fighting systems, information systems.

In the project, despite discussing complex components and designing engineering systems using the BIM platform, thousands of collisions were found between systems. Collision check reports are presented in an exportable HTML table. Working with such a massive amount of data with the specified number of intersections is almost impossible. Most of BIM platforms only allows for viewing the summary model and does not provide the ability to edit a specific intersection.

The modeling elements were prioritised in the following order: structural and architectural solutions, ventilation

systems, sewerage systems, water supply and heating systems, power supply systems, and communication networks.

The engineers work sequentially in predetermined areas of the project, modelling the main volumes of the model within the structural and architectural sections. The ventilation system designers then begin designing in this area, due to the significant volume occupied by the ducts, connectors and equipment in the space.

The order of distribution priority then passes to the engineers of sewerage systems, as the pipelines in this section are designed at a significant slope, and changing their routing is a labour-intensive process. Following these sections, the water supply and heating networks are designed, with power supply and communication networks being designed last. The engineers of each section must consider the work of their colleagues and make room for the systems of other sections by prior agreement when modelling their part and also at the collision checking stage.

The project BIM coordinator performs collision checks in the consolidated model with all sections several times a week, and reports are uploaded to Power BI. Engineers were initially recommended to perform regular daily checks for intersections within their section using tools available in Revit.

To identify potential problems that commonly arise during intersection checking, collisions were studied in a special field, namely in the design and operation of data centers. Some elements of our methodology were applied in a real project [30] and some specific results are discussed in the current research.

As the data center is a complex facility and the coordination of engineering systems at the design stage is a key task, the use of the plug-in significantly helped to optimise the process of eliminating the intersection of engineering systems.

The plug-in enables the identification and elimination of collisions ten times faster than classic tools like Navisworks and Revit.

The changes in the number of collisions within the case study project demonstrate our results. The general dynamics of collision changes is presented in Fig. 4. The real-life project was used as a testing environment. The graph illustrates the frequency of system collisions over time throughout the project's duration.

The number of collisions increases during the engineering systems' work and reaches its peak when all systems are in the model. Subsequently, the number of intersections systematically decreases while using our developed plugin. It offers an effective way to monitor dynamics when eliminating overlaps between components in complex engineering systems.

5. RESULTS AND DISCUSSION

The methodology proposed in this article for collision checking is based on several key factors. These include complying with the sequence of checking systems, excluding

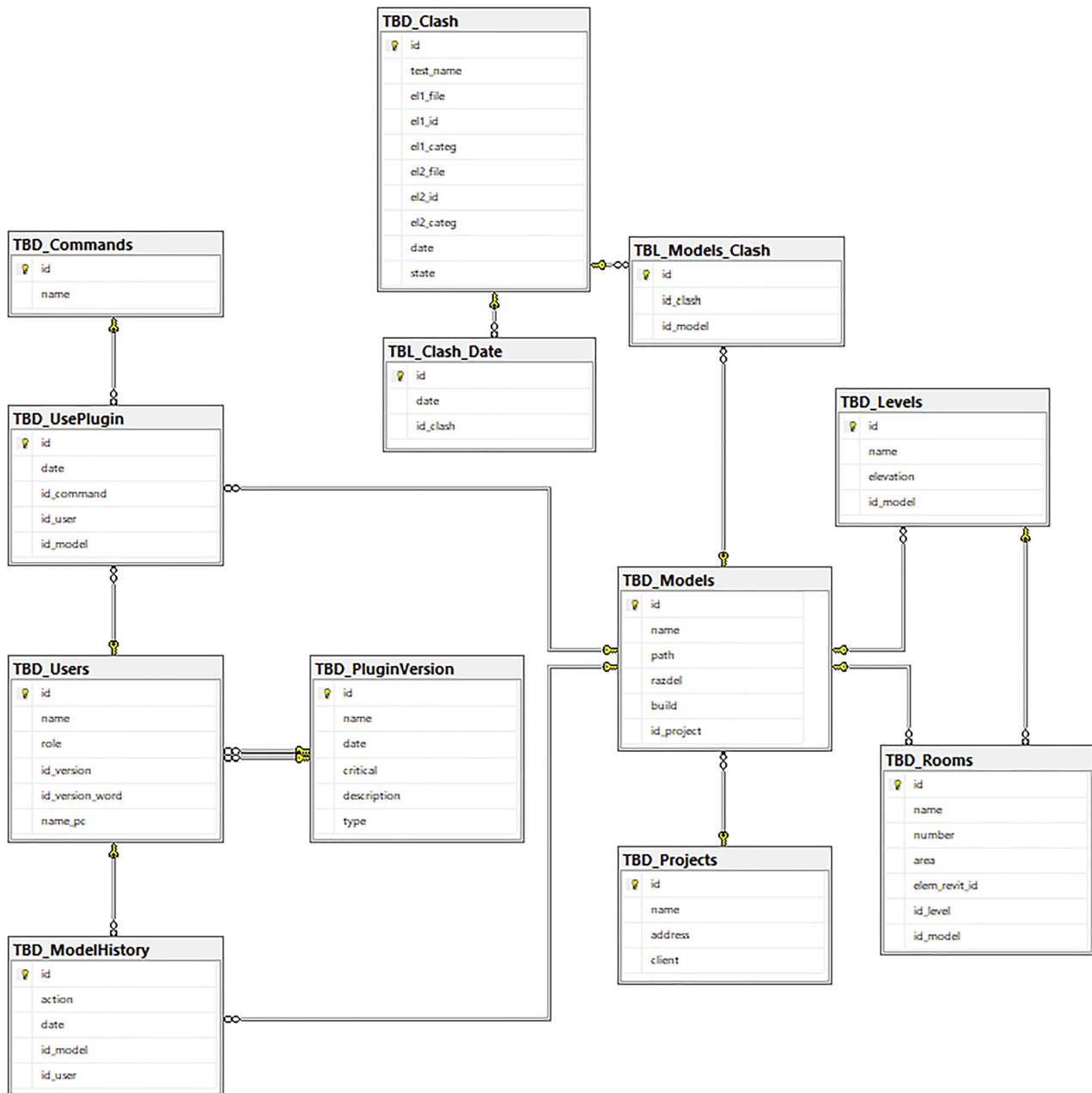


Fig. 3. Plug-in databases diagram

irrelevant collisions, and setting tolerances when joining system elements. For the implementation of the methodology a novel special purpose plug-in was created.

The three levels identified in the context of checking collisions are the following. At the developer level, forms are developed for data visualisation and computer code for the logic underpinning the user interaction with the database. In the context of user-level interaction, users directly engage with the form for search and visualisation of collisions and elimination of singular collisions. Their objective is to identify and eliminate collisions in a prescribed sequence. Following elimination, an identifier is incorporated into the form, in the form of a checkmark. This serves to indicate the successful elimination of the collision, the

date of the elimination, and the name of the user who performed the elimination. The three levels are intimately intertwined and collectively serve as the foundational elements of the plug-in's functionality.

As collisions can lead to construction errors, it is crucial to remove any remaining collisions between the engineering systems after they have been correctly modelled. With the new approach for the plug-in development engineers' work in designing all the modelled systems can be optimised. It enables the identification and correction of intersections approximately ten times faster, while also excluding duplicated intersections and tracking the history of each one. In addition, the report generation process for the intersection of engineering systems has been optimised.

Table 1. Description of the databases

TBD_ModelHistory	History of the models, containing information about who and when opened this or that information model and made changes in it.
TBD_Users	It contains information about users and their roles in the project. Depending on the role of BIM manager or engineer - the menu presented of plug-in looks differently. The role of BIM manager has a great functionality and allows you to view statistics on the work of engineers with models.
TBD_UsePlugin	It contains statistics about how and when the plug-in was used and by which user actions were performed.
TBD_Commands	It contains information about all commands executed by the plug-in and their unique ID.
TBD_PluginVersion	It contains information about plug-in versioning and critical updates.
TBD_Clash_Date	It contains information about all collisions for the whole time of checks. This database is linked to the following database.
TBD_Clash	It contains unique id of collisions, and since one system can have overlap with many other systems, it prevents their duplication during repeated checks if the collision has been eliminated. In case the collision remains, the information about it remains in the database. Also contains information about the working set in which the intersecting systems are located.
TBD_Projects	Service information about projects in which this plug-in is used with their brief description.
TBD_Models	Information about models checked by the plug-in, their names and unique ID.
TBD_Models_Clash	Tables of links of collisions of some systems (models) with other systems (models).
TBD_Rooms	Service database of rooms, contains unique numbers of rooms and their id. If necessary, it allows you to track in which of the rooms there is this or that collision.
TBD_Levels	Service database of floor levels, has a direct link to the previous item and, if necessary, allows you to track the floor on which the collision occurred.

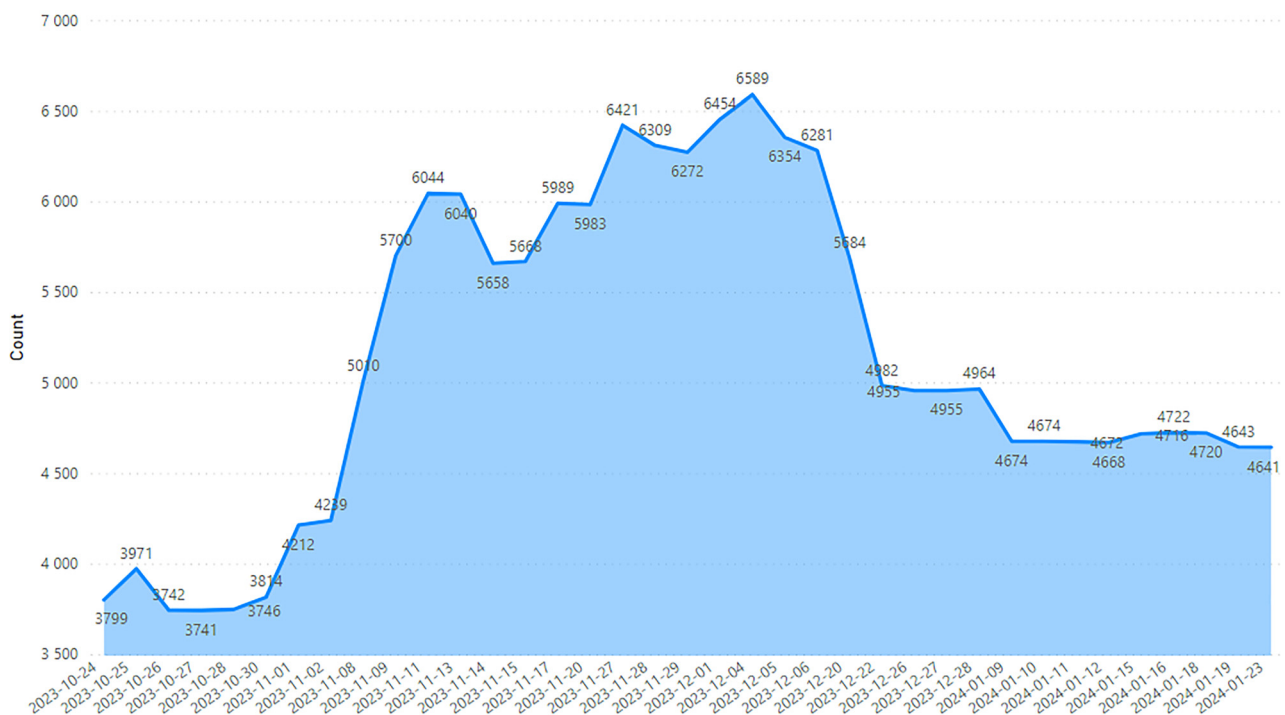


Fig. 4. Clashes for engineering systems (authorised for publication) [30]

The main results achieved with the proposed plug-in development approach backed up with case study data are as follows:

- The plugin's audience coverage refers to the number of engineers who have used it to fix collisions.
- Reducing the time of searching for a single collision. In terms of working hours it looks as follows: the conditional number of clicks - dozens, in the absence of the plug-in and in its presence 10,000 collisions are identical

to 500,000 clicks (without the plug-in) or 50,000 clicks (with the plug-in). If we take a click per second, the speed of collision identification increases tenfold.

- It becomes possible to collect reports in dynamics and statics. To collect such infographics manually, using only standard tools, one report would take several hours, with the plugin up to a few minutes. Data centers typical project lasts on average 1 year during this time approximately 150 reports are prepared, that's 450 h



without using the plugin or 38 h with the plug-in. The main databases from which reports are downloaded are TBD_Clash_Date and TBD_Clash. These databases contain information about all collisions for the entire duration of checks, including unique collision identifiers and information about the working sets in which the intersecting systems are located.

- D. It becomes possible to jointly fix collisions within the same model without additional coordination between employees. Which should save communication time within the project team.
- E. The change histories for each collision are collected in a single database, allowing each one to be tracked as needed.

This study proved that the proposed global solution significantly reduces the number of collisions by performing the systematic checking procedures from the beginning of the design process. The elimination of duplicate intersections has solved the problem of collisions appearing multiple times in the report.

However, it is important to note that not all intersections of two objects are collisions, and therefore, during the process, tolerances of intersections are set. Engineers should not be allowed to set their own tolerances when checking engineering systems. Instead, tolerances should be determined by established industry standards and best practices. This will ensure consistency and accuracy in subsequent works. Creating tolerances based on best practice standards is an adjunct to the further study.

Figure 4 presents project statistics that include irrelevant collisions. In the tested project, irrelevant collisions accounted for approximately 20% of duplicate collisions and 10% of systems where elements are joined.

The use of plugins can eliminate about 30% of these irrelevant collisions.

A comparison of a group of engineers who used a plug-in to eliminate collisions with a group of engineers who did not use a plug-in to eliminate collisions is as follows.

Specialists using the plug-in were able to reduce the time needed to eliminate 30% of the irrelevant collisions. Additionally, the process of identifying, visualising, and correcting 1,000 collisions required up to 40 h of specialist work, which is less than the time required without using the plugin.

The efficiency of the plugin increases significantly when scaling the object.

6. CONCLUSION

In the field of this study, most research focuses on collision detection algorithms and do not consider issues related to the prioritisation of checking various systems, tolerances for joining elements and visualisation of detected collisions. In contrast, our approach has the following benefits.

1. It is of paramount importance to prioritise the collision checking stages of different engineering systems. We presented a new systematic (algorithmic) approach to planning the order of working with engineering systems

in a modelling checking context; consistency of design of different engineering systems and avoidance of duplication are the main benefits of the method.

2. The establishment of tolerances for the various systems involved in the process of joining system elements is recommended to ensure the exclusion of irrelevant collisions. For instance, connecting one element to another is technically correct. However, incorrect tolerance settings when checking the joining of system elements may cause conflicts in the model.
3. A new plug-in has been developed with the intention of facilitating the identification of collisions. This allows the engineer to display a separate window with the selected collision for immediate elimination.

BIM tools utilise clash detection to distinguish between two types of clashes: relevant and irrelevant. Relevant clashes need to be resolved as they lead to lost productivity and rework. Irrelevant clashes, however, are not in need of resolution. They can be a single defect that is repeated many times throughout the project, or clashes that have been deliberately created.

The plugin developed in our article optimises the process of finding and eliminating collisions. Multiple specialists from different systems can work on it simultaneously.

A comparison was made between a group of engineers using our plug-in and those not using it. The specialists who used the plug-in were able to reduce the time needed to eliminate 30% of the irrelevant collisions. In addition, the identification, visualisation and correction of 1,000 collisions required up to week of specialist work, which is less than the time it would have taken without the plug-in.

As the object is scaled, the efficiency of the plug-in increases significantly.

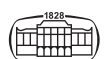
Testing of the plug-in revealed that violating the order of engineering system design significantly increases the number of collisions. Further research will test the plug-ins' operation in designing data centers of varying scales and compare the results. The developed methodology is assumed to be most effective in large data centers with over several hundred racks.

Conflict of interest: The 2nd author, Imre Kocsis is a member of the Editorial Board of the journal. Therefore, the submission was handled by a different member of the editorial team.

ACKNOWLEDGMENT

The authors would like to thank every technical expert of CROC Incorporated for their continuous support during the research activity.

The article credits Alexander Tyapkov and Irina Borodanova from the Department of Engineering and Multimedia Systems with the development of the plug-in. Additionally, engineers from various areas involved in designing data center systems participated in the testing process.



REFERENCES

- [1] R. Chahrour, M. A. Hafeez, A. M. Ahmad, H. I. Sulieman, H. Dawood, S. Rodriguez-Trejo, and N. Dawood, "Cost-benefit analysis of BIM-enabled design clash detection and resolution," *Construct. Manage. Econ.*, pp. 1–18, 2020. <https://doi.org/10.1080/01446193.2020.18027>.
- [2] Y. Hu, D. Castro-Lacouture, and C. M. Eastman, "Holistic clash detection improvement using a component dependent network in BIM projects," *Automation Construct.*, vol. 105, 2019, Art no. 102832, ISSN 0926-5805. <https://doi.org/10.1016/j.autcon.2019.102832>.
- [3] E. A. Pärn, D. J. Edwards, and M. C. P. Sing, "Origins and probabilities of MEP and structural design clashes within a federated BIM model," *Automation Construct.*, vol. 85, pp. 209–19, 2018. <https://doi.org/10.1016/j.autcon.2017.09.010>.
- [4] A. O. Akponeware and Z. A. Adamu, "Clash detection or clash avoidance? An investigation into coordination problems in 3D BIM," *Buildings*, vol. 7, no. 3, pp. 1–28, 2017. <https://doi.org/10.3390/buildings7030075>.
- [5] F. Leite, A. Akcamete, B. Akinci, G. Atasoy, and S. Kiziltas, "Analysis of modeling effort and impact of different levels of detail in building information models," *Automation Construct.*, vol. 20, no. 5, pp. 601–9, 2011. <https://doi.org/10.1016/j.autcon.2010.11.02>.
- [6] I. D. Tommelein and S. Gholami, "Root causes of clashes in building information models," in *Proceedings of IGLC20: 20th Annual Conference of the International Group on Lean Construction*, San Diego, CA, USA, 18–20 July 2012.
- [7] D. L. Craig and C. Zimring, "Support for collaborative design reasoning in shared virtual spaces," *Automation Construct.*, vol. 11, no. 2, pp. 249–59, 2002. [https://doi.org/10.1016/s0926-5805\(00\)00102-3](https://doi.org/10.1016/s0926-5805(00)00102-3).
- [8] Y. E. Kalay, "P3: computational environment to support design collaboration," *Automation Construct.*, vol. 8, no. 1, pp. 37–48, 1998. [https://doi.org/10.1016/s0926-5805\(98\)00064-8](https://doi.org/10.1016/s0926-5805(98)00064-8).
- [9] T. M. Froese, "The impact of emerging information technology on project management for construction," *Automation Construct.*, vol. 19, no. 5, pp. 531–8, 2010. <https://doi.org/10.1016/j.autcon.2009.11.00>.
- [10] T. Korman and L. Simonian, "Using building information modeling to teach mechanical, electrical, and plumbing coordination," in *Proceedings of the 2010 Annual Conference of the ASEE*, Louisville, KY, USA, 20–23 June 2010. [https://doi.org/10.1061/41002\(328\)10](https://doi.org/10.1061/41002(328)10).
- [11] H. W. Ashcraft, "Implementing BIM: a report from the field on issues and strategies," in *Paper Presentation at the 47th Annual Meeting of Invited Attorneys*, 2008. Available online: URL <https://www.scribd.com/document/323993134/ImplementingBIMUpdate-Ashcraft>.
- [12] P. Benning, C. Dumoulin, S. Dehlin, J. Tulke, P. Åberg, S. Fristedt, and R. Holopainen, "Report-collaboration processes," in *Framework for Collaboration; InPro Consortium: Brussels, Belgium*, 2010, pp. 1–125.
- [13] T. Hartmann, "Detecting design conflicts using building information models: a comparative lab experiment," in *Proceedings of the CIB W78 2010: 27th International Conference*, Cairo, Egypt, 16–18 November 2010. URL <http://www.bc.vt.edu/cib2010/>.
- [14] M. T. Shafiq, J. Matthews, and S. R. Lockley, "Requirements for model server enabled collaborating on building information models," *Int. J. 3-D Inf. Model.*, vol. 1, no. 4, pp. 8–17, 2012. <https://doi.org/10.4018/ij3dim.2012100102>.
- [15] P. E. D. Love, D. J. Edwards, J. Smith, and D. H. T. Walker, "Divergence or congruence? A path model of rework for building and civil engineering projects," *J. Perform. Constructed Facil.*, vol. 23, no. 6, pp. 480–8, 2009. [https://doi.org/10.1061/\(asce\)cf.1943-5509.0000054](https://doi.org/10.1061/(asce)cf.1943-5509.0000054).
- [16] S. Han, P. Love, and F. Peña-Mora, "A system dynamics model for assessing the impacts of design errors in construction projects," *Math. Computer Model.*, vol. 57, no. 9–10, pp. 2044–53, 2013. <https://doi.org/10.1016/j.mcm.2011.06.039>.
- [17] K. Kensek and D. Noble, *Building Information Modeling*. New Jersey, NJ, USA: John Wiley & Sons Inc., 2014, pp. 179–80. ISBN 9781118766309.
- [18] M. Leon and R. Laing, "The review of BIM as an effective tool for collaborative design during the early design stages," in *Proceedings of the First UK Academic Conference on BIM*, Newcastle upon Tyne, UK, 5–7 September 2012, pp. 117–23.
- [19] F. Leite, "Capturing expert knowledge in design coordination to train novice designers," 2014. Available online: URL <http://ascelibrary.org/doi/pdf/10.1061/9780784413517.003>. accessed on April 14, 2015.
- [20] L. Wang and F. Leite, "Comparison of experienced and novice BIM coordinators in performing mechanical, electrical, and plumbing (MEP) coordination tasks," in *Proceedings of the Construction Research Congress*, Atlanta, GA, USA, 19–21 May 2014, pp. 21–30. <https://doi.org/10.1061/9780784413517.003>.
- [21] W. Y. Lin and Y. H. Huang, "Filtering of irrelevant clashes detected by BIM software using a hybrid method of rule-based reasoning and supervised machine learning," *Appl. Sci.*, vol. 9, no. 24, p. 5324, 2019. <https://doi.org/10.3390/app9245324>.
- [22] N. Baltabekov, S. Zharassov, T. Zhussupov, and Y. Utepov, "BIM for construction clash detection process after design stage: a case study of high residential complex in Nur-Sultan city," *Technobius*, vol. 1, no. 3, p. 0004, 2021. <https://doi.org/10.54355/tbus/1.3.2021.0004>.
- [23] E. K. Kermanshahi, M. B. M. Tahir, N. H. A. Shukor Lim, A. T. Balasbaneh, and S. Roshanghalb, "Implementation of building information modeling for construction clash detection process in the design stage: a case study of Malaysian police headquarter building," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 476, 2020, Art no. 012009. <https://doi.org/10.1088/1755-1315/476/1/012009>.
- [24] A. Hasannejad, J. M. Sardrud, and A. A. S. Javid, "BIM-based clash detection improvement automatically," *Int. J. Construct. Manage.*, vol. 23, no. 14, pp. 1–7, 2023. <https://doi.org/10.1080/15623599.2022.2063014>.
- [25] M. Juszczak, D. Wiczorek, H. Pacyno, and A. Dzienniak, "Clash detection based on IFC/BIM models – case studies and process parameterization," in *Proceedings of the 6th IPMA SENET Project Management Conference "Digital Transformation and Sustainable Development in Project Management"*, 21–24 September 2022, pp. 110–26. Published: July 6, 2023. <https://doi.org/10.5592/CO/SENET.2022.8>.



- [26] 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 book: digital transformation of the design," *Construct. Manage. Process. Built Environ.*, pp. 115–25, 2020. https://doi.org/10.1007/978-3-030-33570-0_11.
- [27] D. M. Savitri, J. Juliastuti, and A. A. Pramudya, "Clash detection analysis with BIM-based software on midrise building construction project," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 426, no. 1, 2020, Art no. 012002. <https://doi.org/10.1088/1755-1315/426/1/012002>.
- [28] Y. Hu and D. Castro-Lacouture, "Clash relevance prediction based on machine learning," *J. Comput. Civil Eng.*, vol. 33, no. 2, 2019, Art no. 04018060. [https://doi.org/10.1061/\(asce\)cp.1943-5487.0000810](https://doi.org/10.1061/(asce)cp.1943-5487.0000810).
- [29] N. A. Chervova and D. O. Lepeshkina, "Collisions of general utilities designing through the use of BIM-platforms," *Construct. Unique Build. Struct.*, vol. 66, no. 3, pp. 19–29, 2018. <https://doi.org/10.18720/CUBS.66.2>.
- [30] Private Project, JSC CROC, Data Center CROC.

Open Access statement. This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited, a link to the CC License is provided, and changes – if any – are indicated. (SID_1)

