



Knowledge graph exploitation to enhance the usability of risk assessment in construction safety planning

K.W. Johansen^a, C. Schultz^b, J. Teizer^{a,*}

^a Department of Civil and Mechanical Engineering, Technical University of Denmark, Denmark

^b Department of Electrical and Computer Engineering, Aarhus University, Denmark

ARTICLE INFO

Keywords:

Construction safety, hazards, and risk assessment
Construction safety design and planning
Cross-domain knowledge exploitation
Digital twin for construction safety
Knowledge graph representation
Ontology for construction safety

ABSTRACT

Construction projects and their dynamic and yet hazardous work environments face significant challenges. Despite advancements, many proposed solutions for information extraction and utilization remain impractical due to complexity and lack of interoperability. Information is often siloed in proprietary formats, making it difficult to integrate. This issue is evident in the construction safety domain, where advanced risk analysis tools provide detailed insights to hazards but can be overwhelming. Similar challenges exist in cost estimation, schedule evaluation, progress monitoring, and quality compliance checking. Decision-making in construction scheduling struggles to assess how changes impact site safety due to insufficient information and knowledge extraction capabilities, especially when it comes to cross-domain knowledge extraction. This study aims to make safety information accessible to safety and planning professionals. By leveraging Digital Twins, automated safety analysis, and knowledge representation, we enable decision-makers to gain deeper insights into their domain and understand the interplay between project planning and safety. We propose a framework for knowledge extraction, an ontology for capturing knowledge, and query building blocks to transform natural language questions into actionable queries. These methods are tested in a case study, revealing valuable insights into the cross-domain impact of decisions.

1. Introduction

The construction industry is experiencing thought-provoking hazard statistics compared to other industries [1]. The high number of hazards highlighted by these statistics is due to the production environment constantly changing, which makes safety planning labor-intensive. Automated safety analysis and prevention strategies such as [2–6] can identify and plan mitigation to prevent hazards, with greatly increased spatial and temporal resolution beyond what is achieved through the current manual practices of safety analysis. Despite promising results of automated safety analysis being demonstrated in research settings, construction practitioners have not yet been convinced to adopt the automated approaches in the industry. The reasons for this include the lack of integration of automated methods into current workflows, the complexity of correctness assessment, and the placement of responsibility [7,8]. Thus, the exploitation of automatically created safety plans must be investigated further to convince the practitioners about their relevance, applicability, and additional insight.

4D CAD was introduced more than two decades ago [9] and has been

advancing towards 4D BIM (3D BIM and schedules) and later into Digital Twins, which are foreseen to capture planned and real information to improve construction planning, execution, and maintenance [10]. Automated safety planning, inspection, tracking, and training are also part of the Digital Twins for Construction Safety (DTCS) [11]. However, the exploitation of the information and knowledge captured in the comprehensive analysis results is not yet clear. When safety analysis is performed in these automated approaches, it considers every planned change (i.e., part of the schedule) and analyses the impact of these to extract potential hazards and their prevention. The increased temporal and spatial resolution is necessary for identifying complex hazards but, at the same time, results in a large amount of complex information that is difficult to digest and assess for humans.

Over the last decade there has been a major and intensive effort in the research domain on developing smart and AI-enabled construction safety-enhancing solutions [5,12–14]. The application of these AI-based tools is based on the extraordinary amounts of real-time information that are now being extracted from the construction site. However, current workflows are not geared towards handling and disseminating

* Corresponding author.

E-mail address: teizerj@dtu.dk (J. Teizer).

<https://doi.org/10.1016/j.aei.2025.103305>

Received 4 July 2024; Received in revised form 25 February 2025; Accepted 4 March 2025

Available online 23 April 2025

1474-0346/© 2025 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

construction site information in a way where the full potential of this information is exploited. On the contrary, practitioners are overwhelmed with the vast amount of data and information that is now available to them [12,15]. Examples of the information that is becoming available are temporary mitigation equipment [16], personal protection equipment recommendations [17], safety incidents from video and location processing [18], noise level analysis [19], etc. When all this information is shared with the users, it is done through individual applications and interfaces, which is not viable when additional pieces of information start to add up. Therefore, the research presented in this paper aims to capture safety insights that are generated from AI solutions and then present this information to stakeholders in a less comprehensive way, thereby reducing the cognitive load experienced by stakeholders in processing complex information [20] and elevating the captured data and information into *knowledge* [21].

Besides the complexity of safety-related information and its integration with the safety of responsible persons on the construction site, this work is motivated by considering safety from a planning point of view. Current practices of construction planning do not allow overall planning and safety planning to be performed simultaneously [22], but that will have to change in the future DT-enabled construction industry [10,23]. Therefore, design, scheduling, and safety information need to be compiled into a single, unified knowledge representation to extract the combined knowledge from different points of view, and to achieve next-level cost, duration, and safety-optimized construction.

In recent years, Large Language Models (LLMs) have been employed for effective BIM model querying and analysis [24–26] and construction site analysis including safety engineering [27–29]. A major advantage of LLMs is that they provide a natural query interface for users (i.e. natural language) and can exploit data in the form of BIM models, natural language documentation (e.g. safety regulations [30]), and with appropriate prompt engineering and user training offer a distinct approach to automated safety engineering compared with existing methods. In the context of safety analysis, LLMs can be used to identify hazards and propose mitigation measures in the form of “answers” to a query. A primary concern, especially in the context of safety, is establishing the veracity of LLM query results: are they sound (i.e. all identified hazards and mitigation measures correct) and complete (i.e. no important hazards and mitigation measures are omitted).

The research presented in this paper aims to complement existing efforts in the application of LLM by providing a data model (in the form of a *knowledge graph*) for which mathematical properties can be formally proven and precise queries can be specified (Fig. 1). LLM-based interfaces can subsequently be used to rapidly translate from natural language into the target formal model in which properties can be rigorously proven. This approach is adapted from the decades of cyber-physical systems research on model-based safety engineering in other sectors such as automotive, avionics, manufacturing and others [31–33]. As such, an investigation into the efficacy of LLMs in safety engineering is not within the scope of the present paper, and rather this paper focused on the development of the knowledge graph data representation and corresponding query features.

The work investigates how to request such knowledge through

queries that are sent to a specialized safety analysis reasoning engine (through the proposed knowledge base) to improve the exploitation and utilization of safety analysis results in the planning efforts and vice versa. This research relies on previously studied automated safety analysis tools [34,35] and captures their output in the proposed knowledge graph that stakeholders can query to extract knowledge within and outside the safety domain. The study examines the following research questions:

RQ 1: How regular BIM and safety analysis results be captured to facilitate cross-domain/cross-responsibility collaboration and considerations in decision-making?

RQ 2: How can such knowledge representation give safety engineers increased insight into site safety level, crew exposures, and required personal protective equipment?

RQ 3: How can such knowledge representation be used to reveal the impact of delivery delays and crew reassignment on safety resource allocation and future tasks?

The investigation of RQ1-RQ3 leads to the following contributions C1-C3 in construction research and the current construction industry practices.

C 1: A series of query “building blocks” are identified and categorized to facilitate generalized natural language question-to-query transformation, utilized in our example (Q1-Q7).

C 2: A set of nodes and relations are identified and described to capture safety analysis components’ and building components’ connections to schedule information such as tasks and crews.

C 3: Showcasing the knowledge representation capabilities to obtain novel cross-domain knowledge that facilitates improved insight while decisions are made during the planning and execution of a construction project.

Many critical factors and aspects of construction sites are required to be addressed to develop a complete end-to-end digital pipeline for safety analysis. This research focuses on developing and investigating a formal data model of construction safety (i.e. ontology, knowledge graphs and structured query services). Other fundamental construction site factors that are beyond the scope of the present study include: how construction site sensor data is collected, transmitted, processed, stored and managed (IoT and related aspects); optimizing user interfaces (UI), and refining the UI to be fit for on-site application (i.e. as investigated through usability studies); handling uncertain and dynamic aspects of real construction sites, including misapplication of technology on-site; and incorporating image and other on-site sensor data e.g. via drones equipped with sensors, scanners, cameras. Computer vision-based methods and data-driven AI-based safety analysis [18,36–39] is complementary to our approach where the output of such AI-based methods can be used to populate a safety knowledge graph, and as such are not within the scope of the present study.

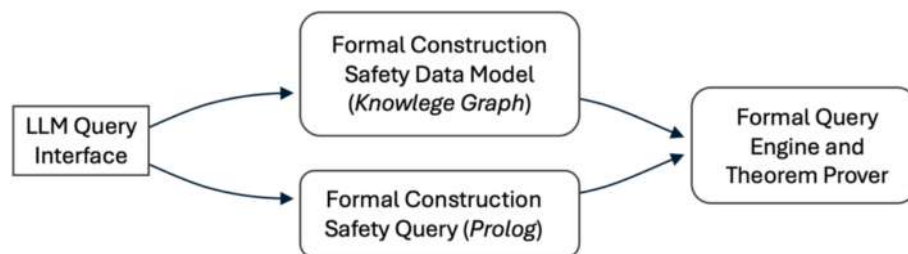


Fig. 1. Model-Based approach to automated construction safety engineering that leverages both the convenience and power of LLMs, and the mathematical rigor of formal analysis and theorem proving.

2. Related work

2.1. Knowledge representation in construction

Different knowledge representations have been proposed with the increasing interest and research of Digital Twins for construction. Several large research projects in Europe are ongoing (e.g., COGITO [40] and BIM2TWIN [41]), each developing an ontology to represent their captured knowledge and needs. While the different representations share the same objective (i.e., to capture the historical, planned, and future state of the construction situation to facilitate an improved construction industry), the approaches still vary slightly. Some strategies invite unprocessed data, and some prefer information and knowledge (i.e., processing data elevates it to information and combinations if information creates knowledge [23]). Ontologies capture the relevant classes and relations that are specific enough to represent the data, information, and knowledge but general enough for similar use cases. Many ontologies exist for different purposes, and reuse of both ontologies and schemas should be considered when creating new ones, as is the case with the Digital Twin Construction Ontology (DTCO) [42], which relies on, e.g., Building Topology Ontology [43] and Building Element Ontology [44]. However, DTCO does not represent safety-related classes and relationships.

Within the domain of safety in construction, previous work has been investigating ontologies to capture and represent safety-related classes and their relationships to other classes within and outside of the safety domain. Among others, [4] presented an ontology that connected the potential hazards (i.e., the new class of the safety ontology) to building elements through job steps, activities, and tasks (i.e., classes of existing ontologies). The safety ontology also captured the potential hazards and their relation to mitigation and safety regulation. In [45], the safety ontology presented in [4] was extended with the capabilities to represent potential hazards to define hazard spaces, represented as spatial artifacts. [34] extended and detailed the spatial artifacts to cover falls-from-heights and struck-by-falling objects hazards in 4D.

While ontologies are available for capturing construction project information and others for safety information, they must be merged to exploit interdisciplinary knowledge. This also means that additional classes and relations are needed to connect the domains. Thus, this study will reuse and extend the existing ontologies to incorporate these connections and additional safety-related perspectives.

There is a growing body of research that investigates the integration of knowledge graphs, ontologies and LLMs [46,47]. Key areas of integration include using knowledge graphs as a form of structure training data [48], and using LLMs for constructing knowledge graphs [49]. Towards exploiting such an integration in the construction safety domain, the research presented in this paper focuses on establishing an conceptually-rich ontology and data model of construction safety that can complement the growing role of LLMs in construction site analysis.

2.2. Digital Twins and related technologies

Digital twins represent a significant advancement in the construction industry, offering real-time monitoring and integration capabilities that surpass traditional methods such as 4D BIM and SPARQL-based query engines. While 4D BIM integrates time-related information with 3D models to enhance project scheduling and visualization, digital twins extend this by creating dynamic, real-time representations of physical assets. This allows for continuous monitoring and optimization throughout the project lifecycle [10].

4D BIM, which combines 3D models with the time dimension, has been instrumental in improving project planning and visualization. However, it lacks the real-time data integration and dynamic updating capabilities of digital twins [9]. SPARQL-based query engines are effective for querying structured data but do not support the dynamic, real-time updates required for construction safety management [50].

The proposed framework leverages the strengths of digital twins to enhance safety planning by integrating real-time data and enabling more effective cross-domain knowledge exploitation. This integration allows for the continuous updating of safety information, providing decision-makers with timely and actionable insights. Recent studies have highlighted the transformative potential of digital twins in construction management, emphasizing their ability to enhance real-time monitoring, improve stakeholder collaboration, and optimise project planning and execution [51,52].

By combining digital twins with knowledge graphs, our approach uniquely enables the seamless integration of safety and planning data, facilitating more informed decision-making and improving overall project safety and efficiency.

2.3. Technologies for improved safety in construction

As it is described in DTCS [53], safety should be considered concerning four main responsibilities: 1) planning (Prevention through Design and Planning), 2) inspection (conformance checking), 3) right-time identification and mitigation of arising hazards in the production (proactively), and 4) improving work crews' safety performance (training), which all require data and creates information that can improve the safety situation in construction.

Different strategies have been proposed in the planning domain to automate identifying and mitigating construction hazards in 3D BIM models, e.g., [2,5,54]. Some studies extend the functionality to consider the schedule in combination with the 3D BIM model (i.e., 4D BIM or Digital Twins, where geometry and schedule are also represented) [4,34,55] or semantic analysis, e.g., [7,56,57]. Recent research highlights the transformative potential of Digital Twins in construction management, emphasizing their ability to enhance real-time monitoring, improve stakeholder collaboration, and optimize project planning and execution [51,52]. For instance, Digital Twins have been shown to increase the transparency of information and provide real-time feedback, which is critical for effective safety management [51]. However, the connection between safety, construction elements, and schedules still must be captured.

Spreadsheet-based checklists (e.g. via Excel) are an effective tool in construction safety engineering owing to their simplicity and familiarity by many users [58–60]. The main advantage of knowledge graphs over only generating checklists is enhanced comprehensibility through visualizations, semantic richness, verifiability and automation: such checklists can be automatically generated if one has access to a knowledge graph; 3D visualizations and 2D diagrams can be automatically generated from knowledge graph linked to a BIM model; knowledge graphs can be formally verified (i.e. assessed in terms of geometry, soundness and completeness, and so on).

3D models can be complex and challenging to assess upfront. Therefore, manual procedures often only perform short-term look-aheads, which means information about the total amount of safety material and where to apply it is rarely available [22]. Furthermore, 4D models can be represented as a sequence of evolving 3D models, which increases the complexity and, therefore, the time consumption for manual approaches. The complexity of the problem is also rising for automated systems, but the impact of necessary manual labor input and costs related to safety planning is constant. However, the complexity of interacting with the results of automated safety planning is increasing when the planning is performed for longer durations and for additional hazard types..

Even though efforts in safety planning are becoming increasingly accurate, the reality might deviate from the plan. There are several reasons for temporarily removing the mitigation measures to access workspaces, and sometimes, these are forgotten to be replaced. Equipment can also be worn out due to the rough environment or even non-compliant installations from the beginning. The safety-responsible person(s) must traverse the job site frequently to ensure continued

compliance, which is labor-intensive, especially when large sites have high activity in several locations and issues gradually evolve. Researchers investigated different strategies to support compliance checking (a.k.a safety inspection), including computer vision from different perspectives [3] and point clouds in combination with deep learning [61]. However, these methods all produce information about compliance but cannot fix the issues themselves. Therefore, humans must be brought into the loop, and the identified compliance issues must be presented clearly and concisely to the practitioners to avoid overwhelming them.

While the details of BIM models and schedules are improving, it is still impossible to foresee all hazard incidents solely based on these. Therefore, identification and mitigation of safety incidents, besides compliance issues, during the execution of the construction project are necessary. Those cases are referred to as dynamic incidents, e.g., overhead crane swings or pedestrian workers not using the designated paths, machine-to-machine- and pedestrian-to-machine-incidents. Identifying these kinds of incidents requires additional sensor techniques to record where the entities are and how they interact. [62–67] are all examples where location tracking technologies are used to identify dynamic hazards, and some, in combination with the cloud, enable them to operate in Digital Twin settings. Similarly, these incidents have to be presented to the stakeholders to increase awareness and improve safety behavior. However, presenting long lists of dynamic safety incidents to the practitioners is overwhelming and does not allow them to act easily to improve the situation.

Besides acting directly on the construction site, the safety insights about hazards and increased model details facilitate the creation of realistic training scenarios for the workers in Virtual Reality (VR). The virtual training allows the workers to be trained in environments representing the concrete construction site at the concrete state where the current progress is considered (i.e., as built) [68–70]. The training impacts the workers' behavior and awareness, and the training performance can be used in the planning when the crew allocation is decided.

The many technology-driven approaches to improve the construction safety situation all produce information that needs to be assessed by human stakeholders. However, if this information is not provided in format that is easy to understand and exploitable then it ends up overwhelming practitioners, resulting in critical safety information being lost. Additionally, there are cases where information produced from individual tools can be even more useful if it is combined with other pieces of information and delivered to stakeholders as *knowledge*. As the safety planning domain is the most developed and streamlined compared to other domains, such as safety inspection and safety behavior monitoring, this work focuses on capturing and representing that information.

2.4. Construction scheduling

Construction scheduling plays a vital role in successfully executing construction projects. The main objective of scheduling is to estimate the cost and time of completion and organize the workflow to fulfill dependencies [71]. Planning a complete construction project is a highly time-consuming task, which motivates the efforts of automated construction scheduling and planning. These automated approaches include case-based reasoning [72], knowledge-based [73], model-based [74], constraint-based [75], genetic algorithms [76], expert systems [77], neural networks [78], and other methods [79]. The decision-makers must select the most desirable one from a selection of automatically generated schedules based on some Key Performance Indicators (KPIs), such as duration, cost, and planning risk assessment of critical path [80]. However, automated construction scheduling that considers Lean principles and incorporates automated safety planning is still unavailable for decision-makers when selecting the desired construction schedule. Therefore, it has yet to be possible to identify the impact that a changed construction task sequence will have on the safety situation on site.

While the construction planning domain has been investigated to assist practitioners in planning procedures, there is still not a lot of focus on integrating safety planning into the broader construction planning domain. One reason is the lack of maturity of automated safety analysis and the lack of support for stakeholders in their interaction with safety analysis results. The lack of maturity is evident as the tools are not yet widely implemented in current practices due to a lack of trust, lack of fit to current workflows [81], lack of support by regulation, and lack of organizational commitment [82]. Additionally, there is a lack of standardization in the automated approaches, which also covers the standardization of how to capture the safety analysis results, meaning that it is hard for the construction planning community to incorporate the safety information in their approaches [81]. This work also attempts to define the integration of safety analysis results in construction plans.

3. Methodology

While the domains of construction safety and construction planning are pushed forward individually in academia, they remain disconnected, as also pointed out in the previous section. This is also the case for the solutions that are already adopted and applied in the industry as pointed out in the introduction. Therefore, this work investigates how the knowledge within an automatically created safety plan can benefit other domains of construction management, such as construction planning. Fig. 2 illustrates the proposed framework, capturing the steps that will be performed and their outcome. The remainder of this document will follow the structure presented in this figure.

The first step is to identify domain-relevant questions to ensure that the solution is relevant to the industry and the practitioners. The questions have been identified through interactions with practitioners on construction sites in Germany, Spain, and Finland, which ensures a wide perspective within Europe. The practitioners are domain experts in construction management with years of experience working on different aspects of construction management. Subsequently, these questions are sorted based on query category and the current availability of information in the safety plan (i.e., the information that is extractable with the safety analysis tool used). The query categories (i.e., Entity, Existential, Relationship, Exploratory, Attribute, Comparative, List, Count search (further explained in Table 3)) are used to determine the nature of the query, which can identify the expected result type.

Two things must be in place before the query engine can be used to answer the questions. First, the natural-language-formulated questions need to be transformed into queries that a query engine can interpret using the query building blocks (Table 3). The query building blocks can ease the later transformation in this study, but also in the longer term, where others have to extend the solution. The other is to define an ontology to capture the BIM, schedule, and safety analysis result information in a knowledge graph that can be queried for extraction of domain knowledge. The two tasks are interdependent as the queries must be formed such that they are using the proposed ontology. It has been chosen to capture the knowledge graph in the ontology that was described [42], which captures the construction elements, processes, and tasks. However, the previous ontology does not describe the safety-related components which have been added on top using the safety ontology proposed previously by the authors [34]. This is further described in the following section 4.2.

After defining the ontology and queries, a knowledge graph needs to be populated for validation. For this study, it has been decided to capture the graph in a local database of Prolog facts, consisting of mainly two types of predicates, called *node* and *edge*. The primary reason for selecting Prolog [83,84] is that, owing to its foundations in mathematical logical, it is tailored for domain-specialised knowledge representation and reasoning, which aligns well with our aim to capture construction safety knowledge, and to make safety analysis more readily accessible to stakeholders, which we accomplish via Prolog's native querying engine. Moreover, Prolog is a suitably powerful programming

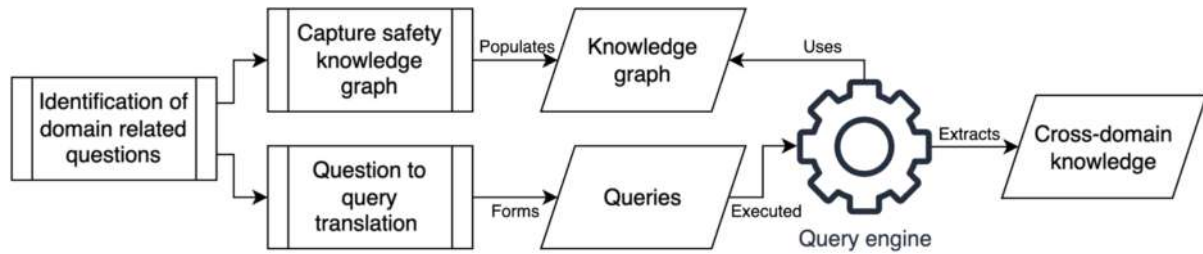


Fig. 2. Research framework capturing the steps required to extract cross-domain knowledge.

language (i.e. Turing complete) and has been proven to be stable and reliable in many application domains. This is also why it is among the most popular declarative languages in research and commercial products. Its query engine allows for fast interaction with a domain captured as facts, which allows for fast lookups and recursions that allow tracing through (logical) relations. Other options for representing knowledge graphs exist (e.g., Neo4J [85], GraphDB [86], or other declarative options [87–89]).

Neo4j offers robust performance for large-scale graph data but may require more complex query syntax for certain logical operations. SPARQL, while powerful for querying RDF data, lacks the native support for recursive queries and logical reasoning provided by Prolog. Our performance tests indicate that Prolog executes queries in less than one second for the case study graph, consisting of approximately 3,800 nodes and 20,000 edges. This efficiency makes Prolog a suitable choice for our framework, balancing performance and ease of use. Therefore, as this study does not aim to identify the better options to represent the graph, it has been decided to move forward with the Prolog approach, which has low overhead, a large and active development community, and a knowledge base that is easy to share.

To address concerns about query efficiency and scalability, we conducted additional tests to evaluate the framework's performance with larger datasets. Our findings indicate that while the current implementation handles the case study graph efficiently, performance bottlenecks may arise with significantly larger graphs. To mitigate this, several approaches can be employed:

- **Limiting the Query Space:** By filtering queries based on relevance, such as introducing spatial details like quadrant and level, the query space can be significantly reduced, improving efficiency.
- **Distributing Workload:** Implementing distributed computing solutions can help manage larger datasets by spreading the workload across multiple processors, thereby increasing computational power.
- **Optimizing the Graph:** Further optimization of the graph using various algorithms can enhance performance. This approach becomes particularly relevant as the graph size increases.

Translating questions into queries might still be too technical and challenging for the average construction manager. However, researchers have been investigating the topic to facilitate and ease translation (e.g., [90–92]). Moreover, the drastic recent improvement of large language model capabilities (e.g., [93]) makes it reasonable to exclude generic query translation, besides the examples that are brought forward, from the scope of this work. Additionally, some LLMs aim to transform natural language-formulated questions into SPARQL queries in a given context, i.e., the ontology [94].

4. Safety knowledge exploitation

This section will present a set of domain questions, and their domain relevance will be reasoned. Subsequently, the ontology will be formed and described, and a knowledge graph will be populated for validation. In the end, the section will briefly describe the translation of the natural-language question into formal queries that the query engine and the

query engine can answer using the knowledge graph.

4.1. Identification of domain questions

As mentioned in the research questions, this study investigates both questions that relate to the individual domains (i.e., RQ 2) and cross-domain (i.e., RQ 3). The first part ensures better exploitation within the safety domain, and the second part ensures that the solution can also move the boundaries in the more advanced cross-domain impact estimation and optimization situation. Even though the questions in the safety domain consider the effect of the schedule (safety analysis has been based on 4D BIM), the answers do not consider changes and, thus, become specifically safety-related. On the other hand, the cross-domain questions consider the impact of safety changes on the construction plan and, vice versa, the impact of the construction plan changes on the safety situation.

The following questions (Q1 to Q7 in *italics*) have been identified in situations where the practitioners have been presented with safety planning of a scheduled construction situation, captured in 4D BIM. Thus, the safety analysis is performed automatically each time the construction site was planned to change (i.e., completion of construction tasks). Additionally, the questions have been inspired by the reasons that have been identified to be the source of real incidents, such as the accident in Ludwigshafen, Germany [95], and in Hamburg, Germany [96], which in both cases involved poor planning of hazard mitigation and synchronization between stakeholders. The questions will represent domain-specific variables with “#” followed by a letter (e.g., #X or #Y), and their rationale is presented in the following:

Q1: “Which hazards are work crew #X exposed to while they complete task #Y?”.

Knowing which kinds of hazards have been identified to expose a crew during the task makes them more aware of what to pay extra attention to, and in some cases, it is even worthwhile for them to learn or refresh their mind on how to act in these situations. Eventually, this learning can be performed in a VR environment as proposed in [68–70], where the selected training scenarios are based on the specific hazards exposing the crew during their task(s).

Q2: “Are hard hats required while performing task #X?”.

Similarly, the stakeholders are interested in particular hazard types, e.g., if the crew is exposed to struck-by falling object hazards and must wear hard hats while performing a certain task. Current practice on the construction site is to make a site-level requirement of wearing the hard hat, which in some cases is not reasonable. Still, the only option for safety planning with the manual approaches does not practically allow safety planning in high enough resolution to separate specific areas. This leads to safety fatigue, and the workers think there is no reason to comply with the requirements, even though the dangers are lurking around the site. This situation can be avoided if the workers were given a concrete reason to wear hard hats, and the requirement can be lifted when there was no reason to uphold it. However, it should be mentioned that automated safety tools at this stage still need further testing and improvements of scheduled activities (i.e., only planned tasks can be automatically safely assessed). For this reason, the tools should only be considered as assisting measures.

Q3: “Are any guard rails installed for shorter than #X number of days?”.

There are situations where the *falls-from-height* hazards appear for only a short duration of time, e.g., when a slab element has been placed and the wall elements are placed immediately after. For these cases, it does not make sense to install guardrails, especially if there are no other tasks planned in the area between the slab installation and wall installation, i.e., the only crew that would be presently exposed to the fall hazards would be the crew removing the guardrails to install the wall. In this case, it is beneficial to identify if the fall hazard remains longer than a practical duration and if any other crews are planned in that area in the meantime. If that is the case, it is more profitable and safe to fence off access to the slab and instruct the wall installation crew to use harnesses instead of installing and removing guardrails. This is also part of common practices nowadays, but this work enables the construction manager to list all the situations and retrieve formal arguments for the verdict. Additionally, it allows the construction manager to gain further insight into the impact of different durations.

Q4: “How is the overall safety situation be affected if crew #X is swapped for crew #Y?”.

If historical information about the construction crews’ safety performance is available, it can be used to improve the crew assignment and obtain improved safety. The information can be collected based on a combination of safety training performance and records of loss of control events [55]. With this information available, it is possible to identify cases where swapping two crews could lead to a lower overall hazard exposure for the construction workers. Additionally, it will have a more significant impact if the scheduled tasks can be swapped, but that would require capturing scheduling dependencies in the knowledge graph to know the available options and that the planning is performed iteratively between general planning and safety analysis. This way, scheduling constraints can remain fulfilled while the wiggle room is exploited to improve the safety of the construction site.

Q5: “How much guardrail must be installed and removed tomorrow?”.

Foreseeing how much hazard mitigation equipment to install and remove the following day, in three days, or during the next week enables the safety management to plan the workforce needed to install it. Additionally, this knowledge can be used by the planning team to consider the local transportation of the equipment. Moving the equipment closer to the application point must be planned in the crane

schedule to ensure optimal utilization and avoid disturbance of the other planned crane activities. Therefore, it is useful from a site logistic perspective and safety perspective to ensure that the required amount of guardrailing is available on the site and in the right area of the construction site or building story in the building when needed.

Q6: “How much guardrail is needed to keep the site safe?”.

In the initial planning stage, the safety management must ensure that the necessary mitigation equipment is available to keep a site compliant with the regulation. Extracting the amount directly from the safety plan eliminates the need to estimate and guess the amount, affecting the site’s storage and continued availability. Capturing the number is, however, more complex as there might be varying reuse policies based on the equipment location and temporal overlap (i.e., different combinations of the installed and removed amount).

Q7: “How is the project be affected if delivery truck #X is postponed by one day?”.

In a situation where the construction management is informed about a delivery truck being delayed for an unforeseen reason and needs to quickly know how this affects the construction project regarding planned safety. It depends on the truckload, but based on a list of delayed building elements, it can be identified if and which hazard spaces and mitigation equipment the elements would replace and the crews to inform about the delay. The crews must be informed that those pieces of safety mitigation equipment are to be kept even though they may appear in their plan to be removed. Consequently, it improves the ability to plan day-to-day activities more agile and adapt to unforeseeable situations that are part of the nature of the construction industry.

4.2. Capturing the knowledge graph

This section describes the ontology proposed to answer the questions identified in the previous section 3 (the ontology is shared at [97]). It has been decided to split the visual representation of the ontology into two figures based on the hazard type, i.e., Fig. 3 and Fig. 4. However, the two hazard types are captured in the same graph, referred to as the knowledge graph (KG) in the following, and consist of approximately 3.8 k nodes and 20 k edges. In addition to the graphical presentation of the ontology, each type of node is detailed in Table 1, and all the edges are described in Table 2.

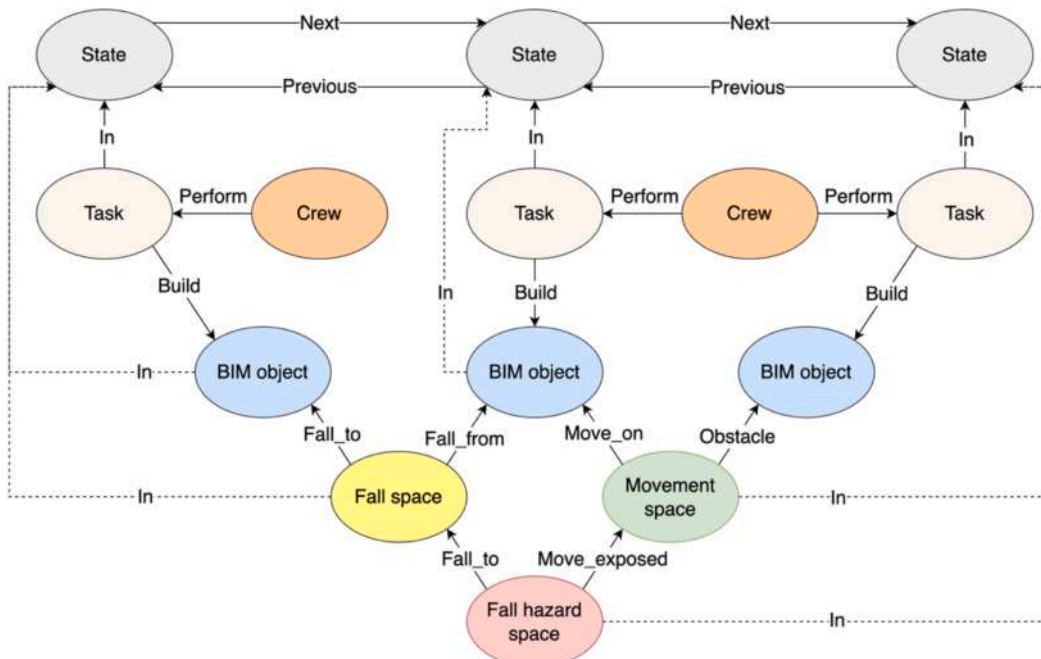


Fig. 3. Illustration of ontology focusing on falls-from-height hazards. The complete ontology is shared on GitHub [97].

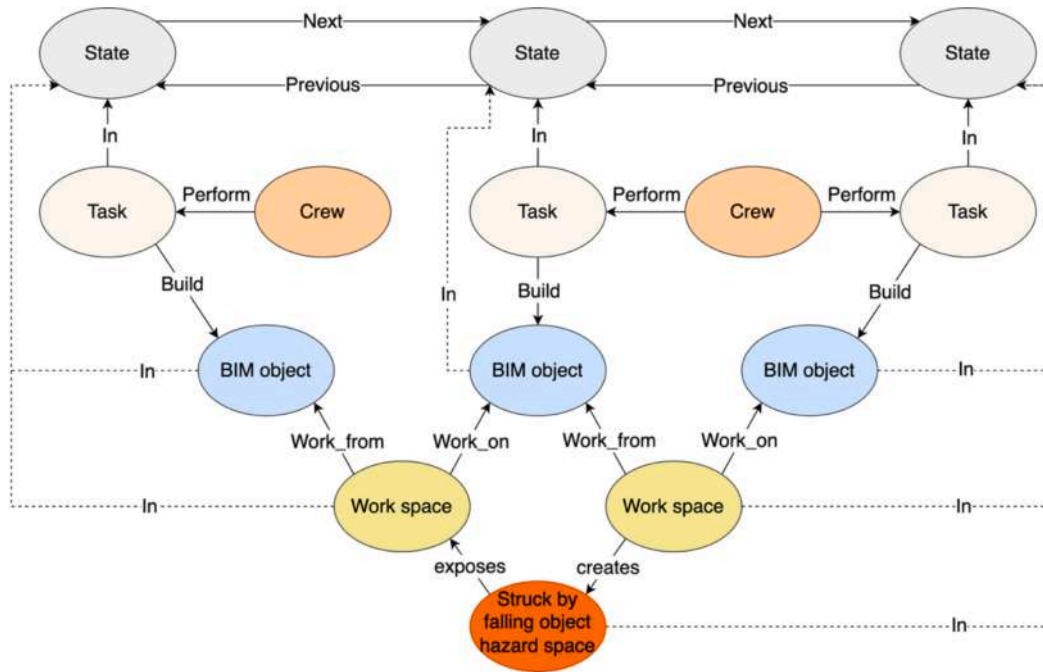


Fig. 4. Illustration of the ontology focusing on struck-by falling object hazards. The complete ontology is shared on GitHub [97].

Table 1

an overview of objects (nodes) in the ontology [97] with descriptions and references to the origin and additional information on the node object types.

Node type	Description	Shown in	Origin
State	A State node represents a planned state of the construction site. A state node is added to the graph for each time step in the set of unique task's planned end times.	Fig. 3 & Fig. 4	[42]
Task	A Task node represents a planned construction task. A task node is created for each individual task of the construction schedule.	Fig. 3 & Fig. 4	[42]
Crew	A Crew node represents a crew that is planned to perform work on the construction site. The crews are captured from the construction schedule.	Fig. 3 & Fig. 4	[42]
BIM object	A BIM object node represents a planned BIM element of the BIM model. A BIM object node is added for each BIM element of the BIM model.	Fig. 3 & Fig. 4	[42]
Fall space	A Fall space node represents the identified fall spaces that are identified by the automated safety planning algorithm. The fall spaces, and the following spatial artefact, are identified in each state, which means there might be replicas or changed versions of the same space (these connections are inferred later). It models where workers or objects can drop more than the regulation allows.	Fig. 3	[54]
Movement space	A Movement space node represents the spatial artifact of type movement space that was identified by the automated safety algorithm. It captures where workers can move around.	Fig. 4	[54]
Fall hazard space	A Fall hazard space node represents a spatial artifact, that is identified by the automated safety planning algorithm. It captures where workers can access fall spaces from movement spaces.	Fig. 3	[54]
Work space	A Work space node represents the spatial artifact that capture, where workers are expected to be present, when they perform a task. These spatial artifacts are also identified by the automated safety planning algorithm.	Fig. 4	
Struck-by falling object hazard space	A Struck-by falling object hazard space node represents the spatial artifact that captures a cone-formed exposure of getting struck by an object dropped from a higher workspace. The automated safety algorithm identifies these spatial artifacts.	Fig. 4	

To capture the knowledge graph in Prolog, we use predicates where a node is represented as a *node(node_type, node_id)* and edges are represented as an *edge(edge_type, from_node, to_node)*. Additionally, we capture a node's attributes with a predicate in the form of *attr(attribute_type, node_id, attribute)* and mapping from local to global variable (e.g., local to global identifier) with *mapping(mapping_type, local, global)*.

When working with a graph, finding related nodes through one or more hops (i.e., edges) is often helpful. To simplify the following code listings, we create a predicate that allows us to formulate a recipe of edges and directions to follow. Lines 1 to 4 in the code snippet below denote a base case that is used when there is only one element in the list (also the case when the end of the list is reached). Also, there is the

recursive step (lines 6–10), which takes the head (i.e., the first element of the list), follows the edge in the direction (i.e., the tuple *(Edge,Dir)*), and calls the predicate again using the tail (i.e., the remaining part of the list) and the node identified in the previous call. There are two versions of this predicate, the one shown and a simplified one that assumes the direction to be *to*. (Listing 1).

We use two kinds of relations: (1) extracted during the analysis procedure and (2) inferred relations identified and added through Prolog queries. From Fig. 3 and Fig. 4, one may notice that some of the *In*-relations are visualized with a stamped line. This is an example of one of the inferred relations identified run time. However, storing these in inferred relationships in memory using the table functionality of Prolog is possible. The additional inferred relations are introduced in this study

Table 2

An overview of the relations (edges) in the ontology [97] with descriptions and usage exemplifying the nodes that are connected by the relation.

Edge type	Description	Usage
Next	The Next edge is used to refer to the next state in the sequence of planned construction states.	(State) → (State)
Previous	The Previous edge is used to refer to the previous state in the sequence of planned states.	(State) → (State)
In	The In edge is used to connect tasks to the state that they are ending in. The connection captures which tasks that have been completed in the specific state. To capture all the completed tasks at a state, one must accumulate the tasks of all previous states following the previous edges.	(Task) → (State)
Perform	The Performs edge connects the crew that is performing the task to the task. This connection captures the assignment of the crews to the tasks in the schedule.	(Crew) → (Task)
Build	The Build edge connect the task to the BIM object(s) that it is the result of performing the task. The connection is captured from the schedule.	(Task) → (BIM object)
Fall onto	The Fall onto edge is used to connect the fall space to the BIM object that a worker or object is falling onto if they are present in the space i.e., the BIM object catching the dropping worker.	(Fall space) → (BIM object)
Fall from	The Fall from edge is used to connect the fall space to the BIM object where the fall space is a worker or object is falling onto if they are present in the space.	(Fall space) → (BIM object)
Move on	The Move on edge connect the movement space to the BIM object from which the movement space emerges, i.e., the surface which the worker will be moving on e.g., a slab	(Movement space) → (BIM object)
Obstacle	The Obstacle edge connect the movement space to BIM objects that interrupts the movement space, e.g., a wall standing on a slab	(Movement space) → (BIM object)
Fall to	The Fall to edge connects the fall hazard space to the fall space, which is contributing to the creation of the fall hazard	(Fall hazard space) → (Fall space)
Move exposed	The Move exposed edge connects the fall hazard space to the movement space that is exposed to the fall hazard	(Fall hazard space) → (Movement space)
Work on	The Work on edge connects the work space to the BIM object that is worked on when the work space is occupied	(Work space) → (BIM object)
Work from	The Work on edge connects the work space to the BIM object that is worked from when the work space is occupied	(Work space) → (BIM object)
Creates	The Creates edge, connects the work space that is creating the struck-by hazard to the hazard node	(Work space) → (Struck-by space)
Exposes	The Exposes edge connects the works space that is exposed by the existence of a struck-by hazard	(Struck-by space) → (Work space)

```

1: follow_edges(Node, Edges, Destination):-
2:   [(Edge, Dir)|[]]=Edges,
3:   ((from=Dir, edge(Edge,Node, Destination)); % <- relation search
4:    (to=Dir, edge(Edge, Destination, Node))). % -> relation search
5:
6: follow_edges(Node, Edges, Destination):-
7:   [(Edge, Dir)|Tail]=Edges,
8:   ((from=Dir, edge(Edge,Node, Intermediate_node)); % relation search
9:    (to = Dir, edge(Edge, Intermediate_node, Node))), % relation search
10:  follow_edges(Intermediate_node, Tail, Destination). % recursion

```

Listing 1. Recursive follow-edges predicate.

to simplify the subsequent queries. A concrete example of the inferred *In*-relations is the one from Fall Hazard space nodes to the state that they exist in. The relation is inferred through the existing relation and uses a combination of relation searches (see Table 3). The example is shown below, which follows the *move_exposed* → *move_on* → *build* → *in* relations to infer the new edge explicitly. (Listing 2).

The same edges can be identified using the previously defined *follow_edges*-predicate, where the route is defined in the list (lines 3–5). (Listing 3).

Additionally, we infer edges that capture the continuation of the spatial artifacts. In lines 2 and 3 of the below, two movement spaces (i.e., MS1 and MS2) are identified, then line 4 identifies a connection between

Table 3

Query categories with examples, where (**bold**) refers to a particular node, (*italic*) refers to a variable, and underscore refers to an anonymous variable.

	Query category	Example	Prolog query	Returns
Relation	Entity search	Find a node (Crew) of type crew .	?-node(crew , Crew).	Node
	Existence search	Are there any states (S) after the current state (s)	?-edge(next, s, _).	Bool
	Relation search	What kind of relationship (RS) do Work space (ws) and BIM object (bo) have?	?-edge(RS, ws , bo).	Type
	Explorative search	Find a node (N) in the neighbourhood of BIM object (bo)?	?-edge(_, N, bo); edge(_, bo , N).	Node
Attribute	Attribute search	What is the length (L) of the railing needed to prevent a particular fall hazard space (fhs)?	?-attr(railing_length, fhs , L).	Value
	Comparative Search	Does one fall hazard space (fhs1) require more railing than another (fhs2)?	?-attr(railing_length, fhs1 , L1), attr(railing_length, fhs2 , L2), L1 > L2.	Bool
List	List search	Which tasks (T) are in a state(s)? Return in list (L)	?-findall(T, edge(in, T, s), L).	List
	Count search	What is the count (C) of objects in a list (L)?	?-length(L, C).	Value

```

1: edge(in, FHS, State) :-
2:   edge(move_exposed, FHS, MS),
3:   edge(move_on, MS, BIM_obj),
4:   edge(build, Task, BIM_obj),
5:   edge(in, Task, State).

```

Listing 2. In-edge predicate.

```

1: edge(in, FHS, State) :-
2:   follow_edges(FHS, [
3:     (move_exposed, to), % gives MS
4:     (move_on, to), % gives BIM object
5:     (build, from), % gives Task
6:     (in, to)], % gives state
7:   State).

```

Listing 3. In-edge predicate using follow-edges predicate.

subsequent states. Lines 5 and 6 make sure that MS1 and MS2 belong to the subsequent states, and lines 7 and 8 make sure that the same BIM object creates the movement spaces. If all these in conjunction (i.e., Boolean and captured with “,”) are satisfiable, we identify the connection. (Listing 4).

In a similar approach, the same is done for the fall spaces. The difference is that we identify fall spaces in lines 2 and 3, and the edge from the spaces is *fall_to* instead of *move_on*.

The previously identified *changed_to* edges identified for movement spaces and fall spaces are used when the *changed_to* edge is captured for fall hazard spaces. In a similar approach, shown in the following listing, two fall hazard spaces in subsequent states are identified in lines 2 to 6. Also, it is checked if the fall hazard space exposes a movement space that has the *changed_to* connection and that the fall hazard space would *fall_to* has the same *changed_to* connection. Consequently, we add the *changed_to* connection between these fall hazard spaces. (Listing 5).

4.3. Translation of questions

To initiate the translation efforts, a set of query “building blocks” has been identified and described to exemplify how different objectives can be achieved and their results. These can also be combined to construct more sophisticated queries, as shown in the following examples. Table 3 presents each query block with an explanation and example of its use in the KG of this work. The following queries are commented with the query category (e.g., “entity search” and “relation search”) to show the usage in query construction.

Q1: “What hazards are work crew #X exposed to while they complete task #Y?”.

Fig. 5 illustrates how to retrieve the exposure of a crew while they perform a task. Firstly, we must decide if we want to identify a crew automatically, i.e., allowing the engine to replace the crew variable. Alternatively, we can bind the crew variable, forcing the query engine to search for solutions involving that crew. This is general for the remaining nodes in this and other queries, but this example is generally about Crews, Tasks, BIM objects, etc Fig. 5.

The crew is identified with an entity search, which is only limited to

```

1: edge(changed_to, MS1, MS2) :-
2:   node(movement_space, MS1),
3:   node(movement_space, MS2),
4:   edge(next, State_prev, State_next),
5:   edge(in, MS1, State_prev),
6:   edge(in, MS2, State_next),
7:   edge(move_on, MS1, BIM_node),
8:   edge(move_on, MS2, BIM_node).

```

Listing 4. Changed-to-edge predicate for movement spaces.

```

1: edge(changed_to, FHS1, FHS2) :-
2:   node(fall_hazard, FHS1),
3:   node(fall_hazard, FHS2),
4:   edge(next, State_prev, State_next),
5:   edge(in, FHS1, State_prev),
6:   edge(in, FHS2, State_next),
7:   edge(move_exposed, FHS1, MS1),
8:   edge(move_exposed, FHS2, MS2),
9:   edge(changed_to, MS1, MS2),
10:  edge(fall_to, FHS1, FS1),
11:  edge(fall_to, FHS2, FS2),
12:  edge(changed_to, FS1, FS2).

```

Listing 5. In-edge predicate for fall hazard spaces.

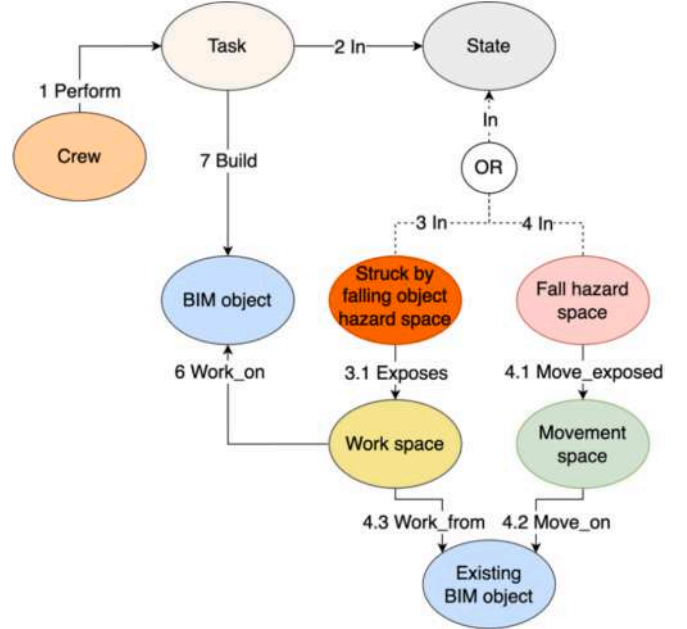


Fig. 5. Graphical representation of query Q1, extracting crew exposures. The numbers prepended to the relations are for references in the text and capture the sequence.

the type of crew. Having a crew assignment allows the engine to identify one of the tasks they are performing (#1), and having the task allows the engine, through a relation search, to find the state where the task is performed (#2). As Q1 is generally talking about hazards, we are interested in struck-by hazards OR fall hazards, and from here, we follow two different routes to identify if the hazards are exposing the crew. The two routes arise from the different nature of the hazards and their relations to the remaining nodes. If the hazard node is a struck-by hazard (i.e., left path #3), then we follow the exposed relation (#3.1) and find the BIM object that is worked on (#6). If the BIM object worked on is also built by the task (#7) identified in the first step (#1), then the query has identified an exposure of a struck-by hazard to the crew.

Similarly, if the entity search identifies a fall hazard, we perform a relationship search to find the exposed movement space (#4.1). Based on the movement space’s relations, we need to find the existing BIM object (#4.2) from which the work is performed (#4.3). As the routes merge from step #4.3 and on the queries, they use the same technique to identify the exposure of a crew. Having the exposures of a crew enumerated allows one to do an entity search to find out whether hard hats are required (i.e., Q2) or to find the hazard source (task and crew), allowing the management to do comparative searches to facilitate decision making that can potentially improve the safety on the site (i.e., Q4). The reader is referred to Appendix A for comparison and additional technical details.

5. Interpretation of query results

In this section, we perform the queries on a high-rise construction project (model and schedules have been shared in [34]), which has been safety assessed. While the case study presented focuses on a high-rise construction project, the proposed framework is adaptable to various construction scenarios, including bridge construction and industrial facilities. The underlying principles of knowledge graph construction and query formulation remain consistent across different project types. For instance, in bridge construction, the framework can integrate safety analysis with structural health monitoring data to provide real-time insights into potential hazards. Similarly, in industrial facilities, the framework can be used to monitor safety compliance and optimise maintenance schedules.

The BIM model, schedule, and safety analysis information have been captured in the proposed knowledge graph, the example queries are showcased, and the query results are interpreted and explained.

Q1: “What hazards are work crew #X exposed to while they complete task #Y?”.

In the previous section, the question was turned into a query, which was then wrapped in the following predicate:

```
exposures_of_crew_doing_task(Crew_id,Task_id,List_of_exposures).
```

Based on the usage, the query engine returns different solutions. If the input is kept as variables (i.e., starting with an uppercase letter), the engine seeks and returns combinations of values that satisfy the body of the rule. If one or more variables are fixed (e.g., replacing Crew_id with the number 1), the engine only returns the combination where crew id is 1, which satisfies the body.

The following resembles the exact use case that is asked in Q1 where crew ID is captured in X and task in Y (only two first solutions separated with “;”): (Listing 6).

From this example, it can be determined that crew 1 is exposed to a struck-by hazard while they perform the task with ID ‘42c5bcd5-ed45-11ec-bd6c-f40228892bcc’. Additionally, the node ID of the hazard is returned along with the hazard type. Fixing the task can be used to identify which crew is exposed, doing that task, and fixing the crew can identify the tasks where the specific crew is exposed to hazards. The query returns *false* if the combination is not represented or there are no hazards exposing the combination. As the hazards are identified, it becomes possible to make dedicated training for the task, select the most hazardous tasks, and even check if the crews have performed the specific training and give them clearance similar to other industries, where safety and security have a high impact.

Q2: “Are hard hats required while performing task #X?”.

This more concrete question checks whether a task is exposed to struck-by hazards. In this example, we test the query with a task ID that was identified in the previous query, which shows that the struck-by exposure means that the workers should wear a hard hat while they are carrying out the task: (Listing 7): The predicate can also retrieve a list

```
Q: ?- guardrails_shorter_than(Hazard_node,2) .
A: Hazard_node = 489 ;
   Hazard_node = 491 ;
... % additional solutions
```

Listing 8. Query and answer for Q3.

of all the tasks that require hard hats. This can be achieved with the setoff functionality or asking for additional solutions with the task ID as a variable.

Q3: “Are any guard rails installed shorter than #X number of days?”.

With this query, it is possible to identify hazard nodes that exist shorter than a specified number of states. The query engine will identify hazard nodes that appear and disappear within the number of states. It will return the hazard nodes that satisfy the rule, which allows the safety manager to make decisions such as instructing the workers to use harnesses instead of installing the guardrail. However, this also means that the area should be fenced off, and signage should be applied. To identify the affected area, one could follow the *expose move* and *move on* relations to identify the affected BIM object. Additionally, it is possible to identify the task and crew from the BIM object through the *build* and *performs*-relations, respectively. (Listing 8).

Q4: “How is the overall safety situation be affected if crew #X is swapped for crew #Y?”.

In the following, we utilize the crew tasks-query in combination with a selection of the state and a mapping to the crew IDs reflected in the schedule. From the schedule, we know that both crews 3 and 4 are construction walls, so the two can be swapped. It can be seen from the query result that crew 3 has worked on tasks that have a ~ 10 % exposure to other work spaces, which for crew 4 is ~ 27 %. With a record of experience, training performance, and incidents, it could benefit the overall safety situation. Similar queries can be formed for critical path tasks, a crew’s experience, and records of delays to improve the chances to finish on time. (Listing 9).

Q5: “How much guardrail must be installed and removed tomorrow?”.

In the query below, the prolog is allowed to replace the state ID and enumerate the solutions, which are separated by “;”. There exist solutions before and after the show section, but states before do not require guardrailing as the building height has not reached the hazardous fall distances stated in the regulation. The following states continue like this, where it can be seen how much guard railing must be installed and removed in the individual states. (Listing 10).

Q6: “How much guardrail is needed to keep the site safe?”.

As explained in the previous section, the installed- and removed- at state predicates can be combined with a setoff function to identify the maximum entry of the set. The entries consist of the length and the state in a tuple formatted as (*Amount, State*), and Prolog allows us to extract each value directly with a tuple as the predicate parameter as follows. With this result, it is known that the maximum amount of guardrailing, considering both the removed and installed amount in each state,

```
Q: ?- exposures_of_crew_doing_task(X,Y,List_of_Exposures) .
A: X = 1,
   Y = '42c5bcd5-ed45-11ec-bd6c-f40228892bcc',
   List_of_Exposures = [(struck_by_hazard, 1466)] ;
   X = 1,
   Y = '42c5bcd5-ed45-11ec-bd6c-f40228892bce',
   List_of_Exposures = [(struck_by_hazard, 1464)] ;
... % additional solutions
```

Listing 6. Query and answer for Q1.

```
Q: ?- is_task_exposed_to_struck_by('42c5bcd5-ed45-11ec-bd6c-f40228892bcc') .
A: true.
```

Listing 7. Query and answer for Q2.

```

Q: ?- node(work_crew,Crew_node),State=177,
      node(state,State),mapping(work_crew,Crew_node,Crew_ID),
      crew_tasks(State,Crew_node,Sum_of_spatial_ol).
A: Crew_node = 22,
   State = 177,
   Crew_ID = 3,
   Sum_of_spatial_ol = 0.0977;
   Crew_node = 25,
   State = 177,
   Crew_ID = 4,
   Sum_of_spatial_ol = 0.2775
... % additional solutions

```

Listing 9. Query and answer for Q4.

```

Q: ?- mapping(state,State_node,State_id),
      installed_at_state(State_node, Installed_Length),
      removed_at_state(State_node, Removed_Length).
A: State_node = 41,
   State_id = 6,
   Installed_Length = 20.60,
   Removed_Length = 0.45 ;
   State_node = 52,
   State_id = 7,
   Installed_Length = 4.82,
   Removed_Length = 5.18 ;
... % additional solutions

```

Listing 10. Query and answer for Q5.

is ~ 61 m and that this will happen in state 37. (Listing 11).

Additionally, the remaining values are available (i.e., with minor changes to the query), such that it would be possible to identify how fast the amount increases, which can be relevant, e.g., in terms of delivery of the mitigation equipment when storage space or supply is limited.

Q7: “How is the project affected if delivery truck #X is postponed one day?”.

In the following, an IFC GUID (i.e., '2QUZHvxvTDdvGHPJrzsmO3') has been identified in the BIM model; the object is an *ifc slab*. With the previously made query, it is possible to identify how its absence impacts other objects. The first and last line of the query is solely there to map the node IDs to meaningful IDs that can be found in the BIM model and scheduled by the decision-makers. As mentioned, we rely on the *work from* relation, which captures that the crew would need to be present on the element to perform a task on the above element. The query does not

check for physical dependencies (e.g., a slab cannot be installed without the supporting columns). Physical dependency information has not yet been added to the example graph because it is unavailable from the schedule. (Listing 12).

6. Conclusions and future work

This study proposes a knowledge-driven approach based on a combination of 4D BIM and automated safety analysis results, enabling the decision-makers to consider impacts from a broader perspective through the interconnection of safety information and general construction information. The knowledge representation enables the extraction of further insight within a domain (e.g., listing hazards for a state or identifying hazards that would make better sense to mitigate with a harness) but also across domains (e.g., how time should be dedicated to

```

Q: ?- minimum_required_guardrailing((Amount,State)).
A: Amount = 60.90,
   State = 37.

```

Listing 11. Query and answer for Q6.

```

Q: ?- IFC_ID = '2QUZHvxvTDdvGHPJrzsm4D',
      mapping(bim_object, Node, IFC_ID),
      affected_by_delay(Node, Affected_bim_objects),
      mapping(bim_object, Affected_bim_objects, A_IFC_ID).
A: IFC_ID = '2QUZHvxvTDdvGHPJrzsm4D',
   Node = 259,
   Affected_bim_objects = 299,
   A_IFC_ID = '2QUZHvxvTDdvGHPJrzsm6j' ;
   IFC_ID = '2QUZHvxvTDdvGHPJrzsm4D',
   Node = 259,
   Affected_bim_objects = 323,
   A_IFC_ID = '2QUZHvxvTDdvGHPJrzsm6Y'
... % additional solutions

```

Listing 12. Query and answer for Q7.

hazard mitigation equipment installation and removal or how a crew assignment impacts safety). We propose a set of nodes and their relations in an ontology, a knowledge graph structure that supports querying. With a set of building blocks, we exemplify how individual blocks or combinations can extract knowledge, such as the above examples. This work identifies natural language questions from different domain and cross-domain perspectives and formalizes these into proper knowledge graph queries based on the proposed and shared ontology. The queries are executed on the query engine that uses the populated knowledge graph (i.e., nodes and their relations) to answer the questions. The efficiency of this system is demonstrated in a case study where all the queries can be performed in less than one second on a laptop for the case study and its graph, which consists of approximately 3.800 nodes and a total of 20.000 edges.

In a case study-inspired approach, we capture the BIM model, its schedule, and corresponding safety analysis results (from previously published work [34]) to populate the knowledge graph. The query execution on the knowledge graph showcases how knowledge beyond current standards can be achieved when the information is combined. Additionally, it is shown how a construction project's spatially and temporally complex nature can be greatly decomplexified and queried very quickly (all the queries presented return answers in less than one second).

The proposed knowledge graph and query system will be particularly valuable in a predefined query setup. In this setup, practitioners specify the desired functionality, and technical experts then build and expose the queries through user interfaces. This division of technical and domain knowledge responsibilities can enhance usability. Initially, this approach may be preferred, but exploring the text-to-query domain, where language models convert natural language into technical queries, is also promising. Both approaches will require a strict data capture schema, constructed from the knowledge of domain experts and information from both traditional and innovative data capture tools and methodologies.

To facilitate the practical implementation of the proposed framework, we recommend integrating it with existing construction management tools such as Autodesk BIM360. This integration would enable seamless data exchange and enhance the usability of the framework in real-world projects. By leveraging the capabilities of established platforms, our framework can provide real-time safety insights and improve decision-making processes.

While this work makes significant findings and contributions, combining safety and general construction information to elevate it towards knowledge, future research avenues remain to further enhance decision-making considering cross-domain impacts and dependencies. More intricate relations and information must be added to the knowledge graph to improve the exploration of cross-domain approaches. Enhancing it with additional schedule dependencies, safety analysis details such as compliance checking results, safety training performance, environmental conditions, and historical knowledge such as worker experience would facilitate even more sophisticated knowledge extraction. Additionally, integrating real-time technologies such as location tracking could identify highly valuable information, as many construction activities are not planned to fully sufficient detail to identify all that is worth knowing. Some activities do demand adaptation. While construction workers are experts at adapting to a situation, it is

impossible to consider all potential outcomes, hazards, or effects on the schedule and cost without capturing the interconnections in computer-assisting frameworks.

One key challenge is the scalability of the framework with larger datasets. As the size of the knowledge graph increases, query performance may be impacted. Future research should focus on optimizing the framework for scalability and exploring advanced indexing techniques to maintain query efficiency. Additionally, integrating real-time data from IoT devices and sensors can further enhance the framework's capabilities. This would enable more dynamic and responsive safety analysis, adapting to changing conditions on the construction site. Future work should also explore the use of language models (LLMs) for automating query generation, which could simplify the interaction with the knowledge graph and make the system more accessible to non-technical users.

In addition to enhancing the knowledge graph itself, the interaction with such solutions will also have to improve. The transformation from questions to queries has been described for the scenarios covered in this work but remains technical and needs simplification before a widespread application in the construction industry is realistic. However, the recent evolution of large language models has shown impressive results and will, in the near future, facilitate the adoption of technologies, such as the proposed, for many industries. Besides engaging the construction practitioners in query transformation, they are also indispensable in evolving knowledge-driven approaches for identifying the real needs of cross-domain knowledge.

Our analysis provides the captured safety and planning knowledge and makes it available to the stakeholders within and outside the safety domain on the construction site. An intriguing observation is that game engines can also address some aspects of properties disseminating captured knowledge among the stake holders and perhaps can be exploited for serious safety analysis [98–100]. This direction will be pursued in deep in future work.

CRedit authorship contribution statement

K.W. Johansen: Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **C. Schultz:** Writing – review & editing, Methodology, Conceptualization. **J. Teizer:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The research presented in this paper was funded by the European Union Horizon 2020 research and innovation program under grant agreements no. 958398 (BIM2TWIN) and BEEYONDERS (101058548).

Appendix A

Technical description of queries

Q1: “Which hazards are work crew #X exposed to while they complete task #Y?”

The construction of the query is initiated by mapping the IDs from the schedule to the IDs of the graph nodes (lines 2 and 3). Having identified the crew, we follow the *performs*-connection (line 4) to identify which tasks they are involved in. The next step is to identify the state and used that state to identify which hazards are present there following the inferred *in*-relation (lines 5 and 6). Lines 7 and 8 filter the nodes as either fall or struck-by

hazards (Boolean disjunction captured with “;”). The *findall*-function builds a list of the tuple defined in line 10 consisting of the *hazard type* and the *hazard node* that fulfills the goal defined in lines 12–15, which captures the exposure of the task. The Boolean disjunction captures the two ways hazards are connected to the task through the edges (i.e., struck-by through work spaces and fall hazards through movement spaces). The tuples are captured in the *List_of_exposures* (line 17), which are exposed through the *exposures_of_crew_doing_task*-predicate (line 1) that captures the natural language formulation of the question.

```

1: exposures_of_crew_doing_task(Crew_id,Task_id,List_of_exposures):-
2:   mapping(work_crew,Crew_node,Crew_id), % node_id to schedule id
3:   mapping(work_task,Task_node, Task_id), % node_id to schedule id
4:   edge(performs,Crew_node,Task_node), % crew entity search
5:   edge(in,Task_node,State_node), % state entity search
6:   edge(in,Hazard_node,State_node), % hazard entity search 1/3
7:   (node(struck_by_hazard,Hazard_node); % hazard entity search 2/3
8:     node(fall_hazard,Hazard_node)), % hazard entity search 3/3
9:   findall(
10:     (HZ_type,Hazard_node), % tuple format
11:     (
12:       node(HZ_type,Hazard_node), % extract hazard type
13:       edge(build,Task_node,BIM_object),
14:       (follow_edges(Hazard,[exposes,work_on],BIM_object));
15:       (follow_edges(Hazard,[move_exposed,move_on,work_from,work_on],BIM_object)
16:     )
17:   ),List_of_exposures % resulting list
18: ).

```

The predicate can be used to receive a list of all the hazards that expose the crew doing a certain task if the predicate is used with fixed values for the *Crew_id* and *Task_id* variables. Additionally, the query can be utilized for exploration, keeping the two variables non-instantiated. Thus, the Prolog engine will find and return combinations that satisfy the body.

Q2: “Are hard hats required while performing task #X?”.

Like the previous query Q1, the first step is to map the schedule-based ID of a task to the graph nodes (line 2). Subsequently, all struck-by hazard entities are tested for satisfying the edges that would mean it is exposing the specific task. If the hazard node *exposes* a work space that is used, while crews are *work_on* on the BIM object that is *build* by the task it means that the task is exposed. If the query is satisfiable, it means that a task is exposed, if it is not satisfiable the query will return *false*, and the task is not exposed to struck-by hazards, and hard hats may not be strictly needed.

```

1: is_task_exposed_to_struck_by(Task_id):-
2:   mapping(work_task,Task,Task_id), % node_id to schedule id
3:   node(struck_by_hazard,Struck_by_hazard), % entity search
4:   follow_edges(Struck_by_hazard,[(work_on,to),(work_on,to),
5:     (build,from)],BIM_object).

```

Q3: “Are any guard rails installed shorter than #X number of days?”.

This query utilizes the inferred *changed_to*-relation to identify when spatial artifacts start and stop existing. Lines 2 to 6 captures that if a node exist, and it is mentioned in the *from_node* position or not in the *to_node* position, it must be the case that this is the first instance of it. Similarly, the ending existence of a node is captured finding all the nodes that are not mentioned in the from position of the *changed_to*-relation. Line 12 to 17 utilizes recursion to identify all the nodes that are successors. In line 12 and 13 the base case is identified, i.e., direct relation between successors and in lines 15 to 17 the recursive step captures the nodes that are related through a relation. Lines 19 to 25 captures a list of all the successors and add them to a list, which give a complete set of all the nodes that an initial node is *changed_to* from the start to the end of existence.

```

1: started_existing_at(Node, State):-
2:   edge(in,Node,State), % relation search
3:   (
4:     not(edge(changed_to,_, Node)); % relation search
5:     edge(changed_to, Node,_) % relation search
6:   ).
7:
8: stopped_existing_at(Node, State):-
9:   edge(in,Node,State), % relation search
10:  not(edge(changed_to, Node, _)). % relation search
11:
12: is_successor(S_A1, S_A2):-
13:   edge(changed_to, S_A1, S_A2). % relation search
14:
15: is_successor(S_A1, S_A2):-
16:   edge(changed_to, S_A1, Z), % relation search
17:   is_successor(Z,S_A2). % recursion
18:
19: trace_successors(Node,Set):-
20:   (started_existing_at(Node,_) ,
21:    setof(Next,is_successor(Node,Next),S) ,
22:    Set=[Node|S]);
23:   (started_existing_at(Node, S) ,
24:    stopped_existing_at(Node, S)
25:   ,Set=[Node]).

```

Based on the above predicates, it is possible to capture how long a specific artifact exists, which is utilized in the below query. After identifying a fall hazard entity in line 3, we use the trace_successors- predicate to retrieve a list of the continued existence, and based on the length of the set, it can be determined how long it exists.

```

1: guardrails_shorter_than(Hazard_node, Num_of_states):-
2:   node(fall_hazard, Hazard_node), % hazard entity search
3:   trace_successors(Hazard_node, Set), % successor relation search
4:   length(Set, Len), % length of successor set
5:   Len=<Num_of_states. % comparative search

```

Q4: “How would the overall safety situation be affected if crew #X is swapped for crew #Y?”.

The construction situation is impacted by the crews’ performance, the type of tasks, and experience, which could be identified by training performance or statistics. These crews impact not only safety but also the risk of delays and increases in cost. The following example intends to identify a crew’s summed spatial overlap of struck-by exposures from their work. The sum can be compared to other crews and might be used to decide that a crew should be swapped for another with more experience. To identify the summed spatial overlap per state, we perform a relation search of all the tasks in a state (line 5). The crew is identified through the *performs*-relation (line 6), and the potential struck-by hazard is identified in lines 7 to 9. Finally, the spatial overlap is captured from the attribute search in line 10 and added to the set, which is summed if any hazards are identified; otherwise, the spatial overlap is set to zero.

Q5: “How much guardrail must be installed and removed tomorrow?”.

To find out how much guard rail is planned to be installed in a state, it is necessary to identify the hazards that either start existing in the state (lines 5–8) or are extended, shown in lines 9 to 15. Line 13 performs a comparison, which essentially checks whether the guard railing length increases, and Line 14 assigns the increased amount to the *FHS_length* variable, which is added to the list. If railings are found in the set, the lengths are summed in

```

1: crew_tasks(State,Crew,Sum_of_spatial_ol):-
2:   setof(
3:     (Spatial_overlap),
4:     Spatial_overlap^(
5:       edge(in,Task_id,State), % relation search
6:       edge(performs,Crew,Task_id), % relation search
7:       follow_edge(Task_id,[(build,to),(work_on,from),(creates,to)],SFA_hazard).
8:       attr(spatial_overlap,SFA_hazard,Spatial_overlap) % attribute search
9:     )
10:   ,SetOfSQL)->
11:   sum_list(SetOfSQL,Sum_of_spatial_ol); % if there is a set then sum
12:   Sum_of_spatial_ol=0. % else return 0

```

line 18. Otherwise, the length is set to zero.

A similar predicate is created for the removed guard railings where the first scenario checks if fall hazards stop existing, and the second checks if any current installations are shortened.

```

1: installed_at_state(State_node,Length):-
2:   setof(FHS_length,
3:     FHS_node^(
4:       node(fall_hazard,FHS_node), % hazard entity search
5:       ( % complete new installation
6:         started_existing_at(FHS_node,State_node), % relation search
7:         attr(railing_length,FHS_node,FHS_length) % attribute search
8:       );
9:       ( % extension of existing
10:        edge(changed_to,FHS_node_old,FHS_node), % relation search
11:        attr(railing_length,FHS_node,FHS_length_new), % attribute search
12:        attr(railing_length,FHS_node_old,FHS_length_old), % attribute search
13:        FHS_length_new>FHS_length_old, % comparative search
14:        FHS_length=FHS_length_new-FHS_length_old % assignment
15:      )
16:     )
17:   ,Set)->
18:   sum_list(Set,Length); % if there is a set then sum
19:   Length=0. % else return 0

```

Q6: “How much guardrail is needed to keep the site safe?”.

The above installed at state and removed at state predicates can be combined to identify the minimum amount of guardrailing required to keep the site safe. Depending on the usage, they identify different cases. Depending on the schedule resolution, it may not be reasonable to assume that the railing is removed immediately in the morning (i.e., installing the building element that removes the hazard may not be planned for the morning). This means that when the minimum amount of the railing for a day is extracted, one must sum the installed and removed (line 8) amount of railing per state and find the maximum entry of the set. The maximum is found at the end of the set (i.e., last entry) as the set is sorted in ascending order.

(continued on next page)

(continued)

```

1: minimum_required_guardrailing(Minimum_amount):-
2:   setof(
3:     (Sum,S_ID),
4:     State^Sum^L_remove^L_install^
5:     (mapping(state,State,S_ID),
6:     installed_at_state(State,L_install),
7:     removed_at_state(State,L_remove),
8:     (Sum is L_install+L_remove)),
9:     Set),
10:    last(Set, Minimum_amount).

```

Q7: “How would the project be affected if delivery truck #X is postponed one day?”.

A delivery truck being delayed has an increasing impact, especially in precast construction, where storage space is limited and building elements are delivered shortly before they are needed. The postponed delivery impacts the crew planned to install it, but other tasks, crews, and building elements also likely depend on its delivery and installation. One direct effect could be that a future construction task will need that element installed to work on the next, and so forth. For example, the recursive functionality easily identifies how the impacts ripple through the schedule, considering the workspaces. Lines 1 to 3 cover the base case, i.e., the direct dependency, and the recursive step in lines 5 to 8 captures the dependency of dependencies. The query can also be based on other dependencies, such as topological requirements or crane scheduling.

```

1: affected_by_delay(Node, Other_bim_node):-
2:   (edge(work_from, Other_node,Node), % relation search
3:   edge(work_on,Other_node,Other_bim_node)). % relation search
4:
5: affected_by_delay(Node, Other_bim_node):-
6:   (edge(work_from, Other_node,Node), % relation search
7:   edge(work_on, Other_node, Z)), % relation search
8:   affected_by_delay(Z, Other_bim_node). % recursion

```

Retrieving a list of affected elements can now be used to find which hazards they are involved in, and it can be identified how these are affected based on their nature. In our schedule, there is no captured information about predecessor and task dependencies, but this information can enable the management to further identify dependencies and the effects of a delay.

References

- [1] B. of labour statistics BLS, Fatal occupational injuries by industry and event or exposure, all United States, 2020, (2020). https://www.bls.gov/iif/oshwc/foi/cftb0340.htm#foi_at_a1.f.2 (accessed February 16, 2022).
- [2] G. Hongling, Y. Yantao, Z. Weisheng, L. Yan, BIM and safety rules based automated identification of unsafe design factors in construction, *Procedia Eng.* 164 (2016) 467–472, <https://doi.org/10.1016/j.proeng.2016.11.646>.
- [3] K. Kim, Y. Cho, S. Zhang, Integrating work sequences and temporary structures into safety planning: automated scaffolding-related safety hazard identification and prevention in BIM, *Autom. Constr.* 70 (2016) 128–142, <https://doi.org/10.1016/j.autcon.2016.06.012>.
- [4] S. Zhang, F. Boukamp, J. Teizer, Ontology-based semantic modeling of construction safety knowledge: towards automated safety planning for job hazard analysis (JHA), *Autom. Constr.* 52 (2015) 29–41, <https://doi.org/10.1016/j.autcon.2015.02.005>.
- [5] G. Tekbas, G. Guven, BIM-based automated safety review for fall prevention, in: S. Ofluoglu, O.O. Ozener, U. Isikdag (Eds.), *Advances in Building Information Modeling*, Springer International Publishing, Cham, 2020, pp. 80–90, https://doi.org/10.1007/978-3-030-42852-5_7.
- [6] K. Kim, J. Teizer, Automatic design and planning of scaffolding systems using building information modeling, *Adv. Eng. Inf.* 28 (2014) 66–80, <https://doi.org/10.1016/j.aei.2013.12.002>.
- [7] A. Pinto, I.L. Nunes, R.A. Ribeiro, Occupational risk assessment in construction industry – Overview and reflection, *Saf. Sci.* 49 (2011) 616–624, <https://doi.org/10.1016/j.ssci.2011.01.003>.
- [8] K.W. Johansen, C. Schultz, J. Teizer, Automated performance assessment of prevention through design and planning (PtD/P) strategies in construction, *Autom. Constr.* 157 (2024) 105159, <https://doi.org/10.1016/j.autcon.2023.105159>.
- [9] B. Koo, M. Fischer, Feasibility study of 4D CAD in commercial construction, *J. Constr. Eng. Manag.* 126 (2000) 251–260, [https://doi.org/10.1061/\(ASCE\)0733-9364\(2000\)126:4\(251\)](https://doi.org/10.1061/(ASCE)0733-9364(2000)126:4(251)).
- [10] R. Sacks, I. Brilakis, E. Pikas, H.S. Xie, M. Girolami, Construction with digital twin information systems, *Data-Centric Eng.* 1 (2020), <https://doi.org/10.1017/dce.2020.16>.
- [11] J. Teizer, K.W. Johansen, C. Schultz, The Concept of Digital Twin for Construction Safety, (2022) 1156–1165. Doi: 10.1061/9780784483961.121.
- [12] T.D. Akinosho, L.O. Oyedele, M. Bilal, A.O. Ajayi, M.D. Delgado, O.O. Akinade, A. A. Ahmed, Deep learning in the construction industry: a review of present status and future innovations, *Journal of Building Engineering* 32 (2020) 101827, <https://doi.org/10.1016/j.jobe.2020.101827>.
- [13] Y. Pan, L. Zhang, Roles of artificial intelligence in construction engineering and management: a critical review and future trends, *Autom. Constr.* 122 (2021) 103517, <https://doi.org/10.1016/j.autcon.2020.103517>.
- [14] Z. Zhang, L. Zhu, A review on unmanned aerial vehicle remote sensing: platforms sensors, data processing methods, and applications, *Drones* 7 (2023) 398, <https://doi.org/10.3390/drones7060398>.
- [15] R. Nyqvist, A. Peltokorpi, O. Seppänen, Can ChatGPT exceed humans in construction project risk management? *Eng. Constr. Archit. Manag.* 31 (2024) 223–243, <https://doi.org/10.1108/ECAM-08-2023-0819>.
- [16] M. Tender, J.P. Couto, A. Gibb, P. Fuller, S. Yeomans, Emerging technologies for health, safety and well-being in construction industry, in: M. Bolpagni, R. Gavina, D. Ribeiro (Eds.), *Industry 4.0 for the Built Environment: Methodologies, Technologies and Skills*, Springer International Publishing, Cham, 2022, pp. 369–390, https://doi.org/10.1007/978-3-030-82430-3_16.
- [17] Z. Wang, Y. Wu, L. Yang, A. Thirunavukarasu, C. Evison, Y. Zhao, Fast Personal Protective Equipment Detection for Real Construction Sites Using Deep Learning Approaches, *Sensors* 21 (10) (2021). Doi: 10.3390/s21103478.
- [18] J. Lee, S. Lee, Construction site safety management: a computer vision and deep learning approach, *Sensors* 23 (2023) 944, <https://doi.org/10.3390/s23020944>.
- [19] N. Henrys, A. Parker, T. Evans, D. Jurevicius, Comparison of predicted and measured noise levels from major infrastructure construction projects, *J. Acoust. Soc. Am.* 154 (2023) A327, <https://doi.org/10.1121/10.0023691>.
- [20] D. Fensel, U. Şimşek, K. Angele, E. Huaman, E. Kärle, O. Panasiuk, I. Toma, J. Umbrich, A. Wahler, *Knowledge graphs: methodology, tools and selected use cases*, Springer International Publishing, Cham, 2020, <https://doi.org/10.1007/978-3-030-37439-6>.

- [21] J. Rowley, The wisdom hierarchy: representations of the DIKW hierarchy, *J. Inf. Sci.* 33 (2007) 163–180, <https://doi.org/10.1177/0165551506070706>.
- [22] B. Li, J. Teizer, C. Schultz, Non-monotonic Spatial Reasoning for Safety Analysis in Construction, in: *Proceedings of the 22nd International Symposium on Principles and Practice of Declarative Programming*, Association for Computing Machinery, New York, NY, USA, 2020: pp. 1–12. Doi: 10.1145/3414080.3414096.
- [23] A. Borrmann, J. Schlenger, N. Bus, R. Sacks, AEC digital twin data - why structure matters, in: S. Skatulla, H. Beushausen (Eds.), *Advances in Information Technology in Civil and Building Engineering*, Springer International Publishing, Cham, 2024, pp. 651–669, 10.1007/978-3-031-35399-4_46.
- [24] D. Liu, X. Zhou, Y. Li, Balancing performance and cost of LLMs in a multi-agent framework for BIM data retrieval, *Architectural Engineering and Design Management* 0 (n.d.) 1–18. Doi: 10.1080/17452007.2025.2456768.
- [25] K. Forth, A. Borrmann, Semantic enrichment for BIM-based building energy performance simulations using semantic textual similarity and fine-tuning multilingual LLM, *Journal of Building Engineering* 95 (2024) 110312, <https://doi.org/10.1016/j.jobe.2024.110312>.
- [26] T.-W. Kang, S.-H. Park, BIM knowledge expert agent research based on LLM and RAG, *KIBIM Magazine* 14 (2024) 22–30.
- [27] J. Lee, S. Ahn, D. Kim, D. Kim, Performance comparison of retrieval-augmented generation and fine-tuned large language models for construction safety management knowledge retrieval, *Autom. Constr.* 168 (2024) 105846, <https://doi.org/10.1016/j.autcon.2024.105846>.
- [28] R. Hussain, D. Lee, M.S. Abbas, A. Pedro, S.F.A. Zaidi, C. Park, Computer vision and large language model based intelligent assistant for onsite construction, *Saf. Manag.* (2025), <https://doi.org/10.2139/ssrn.5081881>.
- [29] S.F.A. Zaidi, M.S. Abbas, R. Hussain, A. Sabir, N. Khan, J. Yang, C. Park, iSafe chatbot: natural language processing and large language model driven construction safety learning through OSHA rules and video content delivery, in: *International Conference on Construction Engineering and Project Management*, 2024, pp. 1238–1245, 10.6106/JCCEPM.2024.1238.
- [30] S.-V.-T. Tran, J. Yang, R. Hussain, N. Khan, E.C. Kimito, A. Pedro, M. Sotani, U.-K. Lee, C. Park, Leveraging large language models for enhanced construction safety regulation extraction, *Itcon* 29 (2024) 1026–1038, <https://doi.org/10.36680/j.itcon.2024.045>.
- [31] P.G. Larsen, J. Fitzgerald, J. Woodcock, P. Fritzson, J. Brauer, C. Kleijn, T. Lecomte, M. Pfeil, O. Green, S. Basagiannis, A. Sadovykh, Integrated tool chain for model-based design of Cyber-Physical Systems: The INTO-CPS project, in: *2016 2nd International Workshop on Modelling, Analysis, and Control of Complex CPS (CPS Data)*, 2016: pp. 1–6. Doi: 10.1109/CPSData.2016.7496424.
- [32] J. Fitzgerald, P.G. Larsen, J. Woodcock, *Foundations for model-based engineering of systems of systems*, in: M. Aiguier, F. Boulanger, D. Krob, C. Marchal (Eds.), *Complex Systems Design & Management*, Springer International Publishing, Cham, 2014, pp. 1–19, 10.1007/978-3-319-02812-5_1.
- [33] P.G. Larsen, H.D. Macedo, J. Fitzgerald, H. Pfeifer, M. Benedikt, S. Tonetta, A. Margaglio, G. Veneziano, L. Sutton, S. Gusmeroli, G. Suciu, Hubcap, A novel collaborative approach to model-based design of cyber-physical systems, in: M. S. Obaidat, T. Oren, F.D. Rango (Eds.), *Simulation and Modeling Methodologies, Technologies and Applications*, Springer International Publishing, Cham, 2022, pp. 90–110, 10.1007/978-3-030-84811-8_5.
- [34] K.W. Johansen, C. Schultz, J. Teizer, Hazard ontology and 4D benchmark model for facilitation of automated construction safety requirement analysis, *Comput. Aided Civ. Inf. Eng.* (2023), <https://doi.org/10.1111/mice.12988>.
- [35] K. Johansen, K. Hong, C. Schultz, J. Teizer, Automated quantification of construction workers' exposure to falling object hazards, *Proceedings of the Institution of Civil Engineers - Management, Procurement and Law* (2024) 1–16. Doi: 10.1680/jmapl.23.00103.
- [36] S. Xu, J. Wang, W. Shou, T. Ngo, A.-M. Sadick, X. Wang, Computer vision techniques in construction: a critical review, *Arch Computat Methods Eng* 28 (2021) 3383–3397, <https://doi.org/10.1007/s11831-020-09504-3>.
- [37] S. Arshad, O. Akinade, S. Bello, M. Bilal, Computer vision and IoT research landscape for health and safety management on construction sites, *Journal of Building Engineering* 76 (2023) 107049, <https://doi.org/10.1016/j.jobe.2023.107049>.
- [38] V.S.K. Delhi, R. Sankaral, A. Thomas, Detection of Personal Protective Equipment (PPE) compliance on construction site using computer vision based deep learning techniques, *Front. Built Environ.* 6 (2020), <https://doi.org/10.3389/fbuil.2020.00136>.
- [39] A.S. Kulinar, M. Park, P.P.W. Aung, G. Cha, S. Park, Advancing construction site workforce safety monitoring through BIM and computer vision integration, *Autom. Constr.* 158 (2024) 105227, <https://doi.org/10.1016/j.autcon.2023.105227>.
- [40] COGITO Landing page, Cogito (2020). <https://cogito-project.eu/> (accessed November 30, 2023).
- [41] BIM2TWIN Landing Page, BIM2TWIN (2020). <https://bim2twin.eu/> (accessed November 30, 2023).
- [42] J. Schlenger, A. Borrmann, T. Yeung, M. Jhonathan, R. Sacks, N. Bus, Digital Twin Construction Ontology, (2023). <https://dtc-ontology.cms.ed.tum.de/ontology/index.html> (accessed November 30, 2023).
- [43] M.H. Rasmussen, P. Pauwels, M. Lefrançois, G.F. Schneider, C.A. Hviid, J. Karlshej, Building Topology Ontology (2017), <https://doi.org/10.13140/RG.2.2.32365.28647>.
- [44] P. Pauwels, Building Element Ontology, (2018). <https://pi.pauwel.be/voc/buildingelement/index-en.html#> (accessed November 30, 2023).
- [45] B. Li, C. Schultz, J. Teizer, O. Golovina, J. Melzner, Towards a unifying domain model of construction safety, health and well-being: SafeConDM, *Adv. Eng. Inf.* 51 (2022) 101487, <https://doi.org/10.1016/j.aei.2021.101487>.
- [46] L.-P. Meyer, C. Stadler, J. Frey, N. Radtke, K. Junghanns, R. Meissner, G. Dziwis, K. Bulert, M. Martin, LLM-assisted knowledge graph engineering: Experiments with ChatGPT, in: C. Zinke-Wehlmann, J. Friedrich (Eds.), *First Working Conference on Artificial Intelligence Development for a Resilient and Sustainable Tomorrow*, Springer Fachmedien, Wiesbaden, 2024, pp. 103–115, 10.1007/978-3-658-43705-3_8.
- [47] Q. Huang, Z. Wan, Z. Xing, C. Wang, J. Chen, X. Xu, Q. Lu, Let's chat to Find the APIs: connecting human, LLM and knowledge graph through AI chain, in: *Proceedings of the 38th IEEE/ACM International Conference on Automated Software Engineering*, IEEE Press, Echnernach, Luxembourg, 2024, pp. 471–483, 10.1109/ASE56229.2023.00075.
- [48] H. Abu-Rasheed, C. Weber, M. Fathi, Knowledge graphs as context sources for LLM-based explanations of learning recommendations, *Global Engineering Education Conference (EDUCON) 2024* (2024) 1–5, <https://doi.org/10.1109/EDUCON60312.2024.10578654>.
- [49] J. Vizcarra, S. Haruta, M. Kurokawa, Representing the interaction between users and products via LLM-assisted knowledge graph construction, in: *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, 2024, pp. 231–232, 10.1109/ICSC59802.2024.00043.
- [50] P. Pauwels, S. Zhang, Y.-C. Lee, Semantic web technologies in AEC industry: a literature overview, *Autom. Constr.* 73 (2017) 145–165, <https://doi.org/10.1016/j.autcon.2016.10.003>.
- [51] J. Park, J.-K. Lee, M.-J. Son, C. Yu, J. Lee, S. Kim, Unlocking the potential of digital twins in construction: a systematic and quantitative review using text mining, *Buildings* 14 (2024) 702, <https://doi.org/10.3390/buildings14030702>.
- [52] A. Ammar, H. Nassereddine, N. Abdulbaky, A. AbouKansour, J. Tannoury, H. Urban, C. Schranz, Digital twins in the construction industry: a perspective of practitioners and building authority, *Front. Built Environ.* 8 (2022), <https://doi.org/10.3389/fbuil.2022.834671>.
- [53] J. Teizer, K.W. Johansen, C.L. Schultz, K. Speiser, K. Hong, O. Golovina, A Digital twin model for advancing construction safety, in: J. Fottner, K. Nübel, D. Matt (Eds.), *Construction Logistics, Equipment, and Robotics*, Springer Nature Switzerland, Cham, 2024, pp. 201–212, 10.1007/978-3-031-44021-2_22.
- [54] K.W. Johansen, C. Schultz, J. Teizer, BIM-based fall hazard ontology and benchmark model for comparison of automated prevention through design approaches in construction safety, *29th International Workshop on Intelligent Computing in Engineering (EG-ICE)*, 2022, 10.7146/aul.455.c231.
- [55] O. Rozenfeld, R. Sacks, Y. Rosenfeld, "CHASTE": Construction hazard assessment with spatial and temporal exposure, *Constr. Manag. Econ.* 27 (2009) 625–638, <https://doi.org/10.1080/01446190903002771>.
- [56] M.R. Hollowell, J.A. Gambatese, Activity-based safety risk quantification for concrete formwork construction, *J. Constr. Eng. Manag.* 135 (2009) 990–998, [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0000071](https://doi.org/10.1061/(ASCE)CO.1943-7862.0000071).
- [57] S. Zhang, J. Teizer, F. Boukamp, Automated Ontology-based Job Hazard Analysis (JHA) in Building Information Modelling (BIM), (2012) 9.
- [58] F. Zaeri, J.O.B. Rotimi, M.R. Hosseini, J. Cox, Implementation of the LPS using an excel spreadsheet: a case study from the New Zealand construction industry, *Constr. Innov.* 17 (2017) 324–339, <https://doi.org/10.1108/CI-01-2016-0002>.
- [59] B. Dasović, U. Klanšek, Comparison of spreadsheet-based optimization tools applied to construction site layout problem. *IOP Conf. Ser.: Mater. Sci. Eng.*, 2021, 10.1088/1757-899X/1209/1/012010.
- [60] T.G.D. Amaral, P.B. Braga, J.P.B. Neto, Application of Dynamic Spreadsheets in the Analysis of Waste by Making-Do, in: *Berkeley, California, USA*, 2020: pp. 841–852. Doi: 10.24928/2020/0077.
- [61] Q. Wang, Automatic checks from 3D point cloud data for safety regulation compliance for scaffold work platforms, *Autom. Constr.* 104 (2019) 38–51, <https://doi.org/10.1016/j.autcon.2019.04.008>.
- [62] I. Awolusi, E. Marks, M. Hollowell, Wearable technology for personalized construction safety monitoring and trending: Review of applicable devices, *Autom. Constr.* 85 (2018) 96–106, <https://doi.org/10.1016/j.autcon.2017.10.010>.
- [63] J. Teizer, T. Cheng, Y. Fang, Location tracking and data visualization technology to advance construction ironworkers' education and training in safety and productivity, *Autom. Constr.* 35 (2013) 53–68, <https://doi.org/10.1016/j.autcon.2013.03.004>.
- [64] K. Hong, J. Teizer, A Data-driven Method for Hazard Zone Identification in Construction Sites with Wearable Sensors, in: *Digital Transformation of Health and Safety in Construction*, 2023: pp. 41–48. Doi: 10.24840/978-972-752-309-2.
- [65] H. Li, X. Yang, M. Skitmore, F. Wang, P. Forsythe, Automated classification of construction site hazard zones by crowd-sourced integrated density maps, *Autom. Constr.* 81 (2017) 328–339, <https://doi.org/10.1016/j.autcon.2017.04.007>.
- [66] O. Golovina, M. Perschewski, J. Teizer, M. König, Algorithm for quantitative analysis of close call events and personalized feedback in construction safety, *Autom. Constr.* 99 (2019) 206–222, <https://doi.org/10.1016/j.autcon.2018.11.014>.
- [67] O. Golovina, J. Teizer, K.W. Johansen, M. König, Towards autonomous cloud-based close call data management for construction equipment safety, *Autom. Constr.* 132 (2021) 103962, <https://doi.org/10.1016/j.autcon.2021.103962>.
- [68] S. Bürkrü, M. Wolf, B. Böhm, M. König, J. Teizer, Augmented virtuality in construction safety education and training, in: L.-C. Ungureanu, T. Hartmann (Eds.), *EG-ICE 2020 Workshop on Intelligent Computing in Engineering*, Proceedings, Universitätsverlag der TU Berlin, 2020, pp. 115–124.

- [69] K. Speiser, J. Teizer, An ontology-based data model to create virtual training environments for construction safety using BIM and digital twins. *Proceedings of the 30th EG-ICE Conference on Intelligent Computing in Engineering*, 2023.
- [70] E.L. Jacobsen, A. Solberg, O. Golovina, J. Teizer, Active personalized construction safety training using run-time data collection in physical and virtual reality work environments, *Constr. Innov.* 22 (2022) 531–553, <https://doi.org/10.1108/CI-06-2021-0113>.
- [71] S. Rokooei, N. Pickavance, A. Shojaei, A. Ross, Construction Industry Perspective on Planning and Scheduling Content, 2020. Doi: 10.3311/CCC2020-043.
- [72] L. Xie, S. Wu, Y. Chen, R. Chang, X. Chen, A case-based reasoning approach for solving schedule delay problems in prefabricated construction projects, *Autom. Constr.* 154 (2023) 105028, <https://doi.org/10.1016/j.autcon.2023.105028>.
- [73] Z. Jiang, H. Zhou, N. Tao, B. Li, Knowledge-based curved block construction scheduling and application in shipbuilding, *J. Shanghai Jiaotong Univ. (sci.)* (2022), <https://doi.org/10.1007/s12204-022-2544-0>.
- [74] Y. Yue, J. Han, S. Wang, X. Liu, Integrated train timetabling and rolling stock scheduling model based on time-dependent demand for urban rail transit, *Comput. Aided Civ. Inf. Eng.* 32 (2017) 856–873, <https://doi.org/10.1111/mice.12300>.
- [75] I.-C. Wu, A. Borrmann, U. Beißert, M. König, E. Rank, Bridge construction schedule generation with pattern-based construction methods and constraint-based simulation, *Adv. Eng. Inf.* 24 (2010) 379–388, <https://doi.org/10.1016/j.aei.2010.07.002>.
- [76] V. Faghihi, K.F. Reinschmidt, J.H. Kang, Construction scheduling using Genetic Algorithm based on Building Information Model, *Expert Syst. Appl.* 41 (2014) 7565–7578, <https://doi.org/10.1016/j.eswa.2014.05.047>.
- [77] N. Markiz, A. Jade, Integrating an expert system with BrIMS, cost estimation, and linear scheduling at conceptual design stage of bridge projects, *Int. J. Constr. Manag.* 22 (2022) 913–928, <https://doi.org/10.1080/15623599.2019.1661572>.
- [78] J.I. Obianyo, R.C. Udeala, G.U. Alaneme, Application of neural networks and neuro-fuzzy models in construction scheduling, *Sci Rep* 13 (2023) 8199, <https://doi.org/10.1038/s41598-023-35445-5>.
- [79] V. Faghihi, A. Nejat, K.F. Reinschmidt, J.H. Kang, Automation in construction scheduling: a review of the literature, *Int J Adv Manuf Technol* 81 (2015) 1845–1856, <https://doi.org/10.1007/s00170-015-7339-0>.
- [80] Y. Hatipkarasulu, A conceptual approach to graphically compare construction schedules, *Constr. Innov.* 20 (2019) 43–60, <https://doi.org/10.1108/CI-01-2019-0001>.
- [81] M. Fargnoli, M. Lombardi, Building Information Modelling (BIM) to enhance occupational safety in construction activities: research trends emerging from one decade of studies, *Buildings* 10 (2020) 98, <https://doi.org/10.3390/buildings10060098>.
- [82] M. Shafiq, M. Afzal, Improving Construction Job Site Safety with Building Information Models: Opportunities and Barriers, in: 2021: pp. 1014–1036. Doi: 10.1007/978-3-030-51295-8_71.
- [83] A. Colmerauer, P. Roussel, The birth of Prolog, in: *History of Programming Languages—II*, Association for Computing Machinery, New York, NY, USA, 1996: pp. 331–367. Doi: 10.1145/234286.1057820 (accessed September 4, 2023).
- [84] W.F. Clocksin, C.S. Mellish, *Programming in prolog*, Springer, Berlin Heidelberg, 2003, 10.1007/978-3-642-55481-0.
- [85] Neo4J, Neo4j Graph Database & Analytics – The Leader in Graph Databases, Graph Database & Analytics (n.d.). <https://neo4j.com/> (accessed May 23, 2024).
- [86] GraphDB, GraphDB Downloads and Resources, (n.d.). <https://graphdb.ontotext.com/> (accessed May 23, 2024).
- [87] V. Lifschitz, *Answer set programming*, Springer International Publishing, Cham, 2019, 10.1007/978-3-030-24658-7.
- [88] L. Brownston, R. Farrell, E. Kant, N. Martin, *Programming expert systems in OPS5: an introduction to rule-based programming*, Addison-Wesley Longman Publishing Co., Inc, USA, 1985.
- [89] T. Fruhwirth, *Constraint handling rules*, Cambridge University Press, 2009.
- [90] G.J.D.R. Hains, Y. Khmelevsky, T. Tachon, From natural language to graph queries, in: *IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, 2019, pp. 1–4, 10.1109/CCECE.2019.8861892.
- [91] S. Hu, L. Zou, J.X. Yu, H. Wang, D. Zhao, Answering natural language questions by subgraph matching over knowledge graphs, *IEEE Trans. Knowl. Data Eng.* 30 (2018) 824–837, <https://doi.org/10.1109/TKDE.2017.2766634>.
- [92] S. Fuchs, J. Dimyadi, M. Witbrock, R. Amor, Improving the semantic parsing of building regulations through intermediate representations, *30th EG-ICE International Workshop*, 2023.
- [93] V. Câmara, R. Mendonca-Neto, A. Silva, L. Cordovil-Jr, DBVinci – towards the usage of GPT engine for processing SQL Queries, in: *Proceedings of the 29th Brazilian Symposium on Multimedia and the Web*, Association for Computing Machinery, New York, NY, USA, 2023: pp. 91–95. Doi: 10.1145/3617023.3617035.
- [94] Langchain, Ontotext GraphDB | LangChain, (n.d.). <https://python.langchain.com/v0.1/docs/integrations/graphs/ontotext/> (accessed May 23, 2024).
- [95] Unfall bei der BASF in Ludwigshafen 2016, Wikipedia (2024). https://de.wikipedia.org/w/index.php?title=Unfall_bei_der_BASF_in_Ludwigshafen_2016&oldid=243050265 (accessed May 30, 2024).
- [96] NDR, Unfall auf Baustelle in der Hafencity: Fünfter Bauarbeiter gestorben, (n.d.). <https://www.ndr.de/nachrichten/hamburg/Unfall-auf-Baustelle-in-der-Hafencity-Fuenfter-Bauarbeiter-gestorben,arbeitsunfall172.html> (accessed May 30, 2024).
- [97] K. Johansen, kakke14/SafeConOntology, (2024). <https://github.com/kakke14/SafeConOntology> (accessed May 30, 2024).
- [98] A. Ezzeddine, B. García de Soto, Connecting teams in modular construction projects using game engine technology, *Autom. Constr.* 132 (2021) 103887, <https://doi.org/10.1016/j.autcon.2021.103887>.
- [99] H. Voordijk, F. Vahdatikhaki, Virtual Reality learning environments and technological mediation in construction practice, *Eur. J. Eng. Educ.* 47 (2022) 259–273, <https://doi.org/10.1080/03043797.2020.1795085>.
- [100] F. Vahdatikhaki, A.K. Langroodi, D. Makarov, S. Miller, Context-realistic virtual reality-based training simulators for asphalt operations, *ISARC Proceedings* (2019) 218–225.