[1]

Two Decades of Automated AI Planning Methods in Construction and Fabrication: A Systematic Review

SHERMIN SHERKAT, Institute for Computational Design and Construction (ICD). Department for Computing in Architecture (ICD/CA) Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart, Germany

THOMAS WORTMANN, Institute for Computational Design and Construction (ICD). Department for Computing in Architecture (ICD/CA), University of Stuttgart, Germany

ANDREAS WORTMANN, Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart, Germany

Task planning and scheduling are crucial for construction or fabrication (CF) processes. Automating them is necessary for more efficient plans in terms of time and resources. However, most construction planning processes are still performed manually despite the existence of various AI methods. Symbolic AI automated task planning (ATP) techniques offer a variety of features to tackle task planning problems, but their application to CF has not been researched yet. This study identifies the current state of research and gaps in the literature regarding these AI techniques while providing directions for future research. We conduct a systematic review that evaluates existing literature on ATP in terms of environmental characteristics, modeling languages, ATP techniques, and results. We searched the ACM, IEEE, Scopus, WOS, and SpringerLink databases for papers published in the last 20 years (2002-2022) that discuss symbolic AI methods used in task planning within the CF fields. Our findings indicate that research on automated planning is currently limited regarding the characteristics of CF environments. Only a few papers have utilized symbolic languages, AI planners, and ATP techniques. No paper has evaluated their planning system in an on-site CF process. As a result, many symbolic languages, planners, and ATP techniques remain unexplored.

CCS Concepts: • **Computing methodologies → Planning and scheduling**.

Additional Key Words and Phrases: AI, artificial intelligence, construction, fabrication, plan, TAMP, civil engineering, systematic review, modeling languages, symbolic languages, reinforcement learning, symbolic AI

[1]This research was supported by the Ministry of Science, Research and the Arts Baden-Wuerttemberg in the Artificial Intelligence Software Academy (AISA) and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2120/1 - 390831618.

Authors' addresses: Shermin Sherkat, shermin.sherkat@icd.uni-stuttgart.de, Institute for Computational Design and Construction (ICD). Department for Computing in Architecture (ICD/CA) and Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart, Seidenstrasse 36,70174, Stuttgart, Baden Württemberg, Germany, 70174; Thomas Wortmann, Institute for Computational Design and Construction (ICD). Department for Computing in Architecture (ICD/CA), University of Stuttgart, keplerstrasse 11, 70174, Stuttgart, Baden Württemberg, Germany, thomas.wortmann@icd.uni-stuttgart.de; Andreas Wortmann, Institute for Control Engineering of Machine Tools and Manufacturing Units (ISW), University of Stuttgart , Seidenstrasse 36, Stuttgart, Baden Württemberg, Germany.

Manuscript submitted to ACM

ACM Reference Format:
Shermin Sherkat, Thomas Wortmann, and Andreas Wortmann. 2018. Two Decades of Automated AI Planning Methods in Construction and Fabrication: A Systematic Review. *J. ACM* 37, 4, Article 111 (August 2018), 21 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Most of what is considered AI today falls under the category of subsymbolic AI, e.g., machine learning, which is a powerful tool for processing large amounts of data [44]. However, the field of AI includes two paradigms: symbolic (logic-based) AI and subsymbolic (data-based) AI. Symbolic AI uses logic, reasoning, and search to make inferences based on explicit, human-readable symbol manipulation. In contrast, subsymbolic AI uses statistical learning algorithms to identify patterns and relationships based on numbers/features and the distance between them [6, 44]. Symbolic AI, especially for logical problems like planning, has proven highly effective and has developed various techniques for different types of planning problems since its downfall during the AI winter[1]. Automated planning, a subfield of symbolic AI planning, focuses on the computational aspect of the reasoning involved in generating a sequence of tasks that will lead from an initial state to a desired goal state [22]. Automated planning techniques have been successfully applied to different fields such as robotics and production planning [46] to automate task planning, path planning, and more.

Currently, most projects in the construction industry rely heavily on manual planning and scheduling workflows [2]. This lack of automation can be due to (1) the way knowledge is modeled, stored and maintained, which requires constant heavy manual input; (2) the limited application of current efforts to real-life construction processes; and (3) decoupled automated planning and scheduling optimization techniques [2].

Current reviews have conducted scientometric analysis [8, 12, 32] and systematic mapping [1, 38] on AI methods in the CF [2] field. Also, systematic reviews have been done on automation methods for construction planning [2] and scheduling [15]. However, while automated AI planning techniques for task planning in CF can potentially address current issues regarding planning automation, none of the existing reviews have addressed them in detail.

We have conducted a systematic review using the PRISMA statement [36] to identify the current state of the use of ATP techniques in CF and pinpoint research gaps. As several symbolic AI methods exist for task planning, we included only automated task planning techniques[3] and excluded expert systems and CBR because published review papers have already explored them in detail [15, 55]. We searched the last twenty years of the ACM, IEEE, Scopus, WOS, and SpringerLink databases, leading to six relevant papers. This paper aims to answer questions about (1) planning environments, (2) languages and tools, (3) ATP techniques, and (4) evaluation of results.

Our findings show that (1) current research does not adhere to the full environmental requirements of CF processes, which makes the results difficult to use in a real-world process planning, (2) current research does not deploy the variety of languages designed to solve planning problems for different environments and consequently is not able to leverage powerful AI planners that work with these languages, (3) very few aspects of existing ATP techniques have been explored, and some have not been explored at all, and (4) most papers have not applied the methods to real-world CF processes.

Consequently, our contributions are:

[1]AI winter is a period of reduced funding and interest in artificial intelligence research [42]
[2]Acronyms used in this paper are listed in Table 13
[3]Automated planning refers to [22], defined as domain-independent AI planning techniques that rely on abstract models of actions and the relevant environment

(1) Categorization of papers on AI planning from the last twenty years according to their planning environments, modeling languages and tools, and ATP techniques
(2) Identification of limitations, challenges, gaps, and best practices for ATP in CF
(3) A roadmap for future research on ATP in CF

2 BACKGROUND: BASIC CONCEPTS AND CORRESPONDING TERMINOLOGIES

Automated planning has been a significant area of research in AI since its early days. Planning is considered a demonstration of intelligence, a deliberation process that selects and organizes actions by anticipating their expected outcomes [22]. ATP involves exploring computational models and techniques to generate, evaluate, control, and implement task plans [25]. ATP is a form of symbolic AI where we define a knowledge base and utilize logic and reasoning to create a plan [22].

We first briefly explain some common AI terms used in ATP based on [42]:

- A 'state' is a description of problem elements at a given moment.
- The 'initial state' is the state the agent starts in.
- the 'state space' is the set of all states reachable from the initial state.
- the 'goal state' is the desired final state.
- 'Actions' are state transitions and change one state to another.
- A 'path' is a sequence of states connected by a sequence of actions.

Planners search for a path from the initial to the goal state while minimizing the cost or time. In an ATP problem, this path is called the plan [42]. As a symbolic AI method, ATP requires a knowledge base that includes a model of the planning domain and the planning problem defined in a planning language. The domain model includes **predicates** that describe a state using relations and **actions** that describe the state transition. The problem model includes an **initial state**, a **goal state**, and **elements** that exist in the relevant world. Actions are defined in terms of preconditions and effects (Figure 1). The planner then uses the domain and problem models and the action definitions to find a plan using various algorithms, such as GraphPlan in the Metric-FF planner [21, 27].

To assess the plans created with ATP techniques, we need to know about the planning environment, the language with which the knowledge base has been modeled, and the ATP planning techniques (Figure 2). Finally, we need to evaluate the results. In the following sections, we explain all of these criteria.

Fig. 1. Inputs, outputs, and the process of AI planning [22]

Table 1. Environmental characteristics definitions for AI tasks [42]

Environmental characteristics	Definition
Observable	The sensors receive all data relevant to the choice of action.
Partially observable	Part of the data is missing due to noisy or inaccurate sensors.
Not observable	There is no sensor.
Deterministic	The effect of an executed action is always as expected.
Stochastic	Actions have probabilistic outcomes.
Episodic	The next state does not depend on the actions taken in previous. states.
Sequential	Each action could affect all future actions.
Static	The environment is static while an agent is deliberating.
Dynamic	The environment can change while an agent is deliberating.
Continuous	The state, time and actions are described as continuous. variables.
Discrete	The state, time and actions are described as discrete variables.
Multi agent	Agents behavior performance measure depends on each other.
Single agent	There can be only one agent.

2.1 Environmental Characteristics

AI task environments can be categorized based on some key characteristics listed in Table 1. These characteristics are important in determining the most effective design for an AI agent and in implementing the appropriate techniques for each task environment [42].

2.2 Languages

Two paradigms of programming languages exist: declarative and imperative [35]. Declarative programming defines the results of a program without describing a step-by-step process, while imperative programming describes a step-by-step process for a program's execution. The main difference between the two is that imperative programming is focused on how to achieve a task, while declarative programming is focused on what the task is [35]. ATP uses declarative languages to model the domain, initial state, and goal states. Then, AI planners can handle planning problems, ranging from complex domains with uncertainty, partial observability, and temporal constraints to simple deterministic domains with known actions and outcomes. Declarative languages differ depending on the planning technique and the planners. Modeling the domain, language requirements, and planners are closely linked. For instance, a planner might only accept a domain modeled using a specific version of planning domain definition language (PDDL), which filters out all the other languages. Similarly, insisting on a particular language version limits the supporting planners.

2.3 ATP Methods

Planning techniques include classical planning, temporal and numerical planning, probabilistic planning, hierarchical task networks, online/offline planning, or a combination of these techniques (hybrid planning) to tackle more complex problems. In this section, we define the planning techniques based on the environmental characteristics outlined in Section 2.1. Classical planning involves planning in deterministic, fully observable, and discrete environments [6]. While classical planning answers what tasks to do and in which order, temporal planning focuses on when a task occurs and the duration required for executing each task [6]. Temporal planning only considers time as the continuous variable, while all other variables are discrete [44]. However, numerical planning allows for additional continuous variables besides time [44]. Probabilistic planning, on the other hand, deals with planning in stochastic environments [6] and online planning in stochastic and dynamic environments. Hierarchical Task Networks (HTNs) are methods for solving planning problems that consist of abstract tasks and their decomposition methods [11], making large problems more manageable.

Motion planning is another form of planning, which is relevant to robotic assembly problems. The challenge of motion planning is finding a feasible trajectory in space and time [22]. It includes (1) identifying a path within the environment to move a mobile system from its initial position to the desired goal position (2) considering the control law along the path while accounting for the dynamic limitations of the mobile system, such as speed, kinematics, and acceleration [22]. Task planning and motion planning differ in their level of abstraction, language, and goal. Task planning handles high-level tasks, while motion planning deals with low-level geometrical planning such as path-finding or collision checking [20].

2.4 Result Evaluation

The technology readiness level (TRL) scale is widely used as a tool to assess technology maturity across various industries [37]. This scale starts with a technology in its early scientific form (TRL 1) and progresses toward a proven technology in an operational environment (TRL 9). We use this scale, as evaluating a technology's position on the TRL scale enables a better understanding of its readiness for eventual deployment [37].

To facilitate analysis and interpretation, we clustered TRL levels into three main categories: research (TRL 1-3), development (TRL 4-6), and deployment (TRL 5-9) [37].

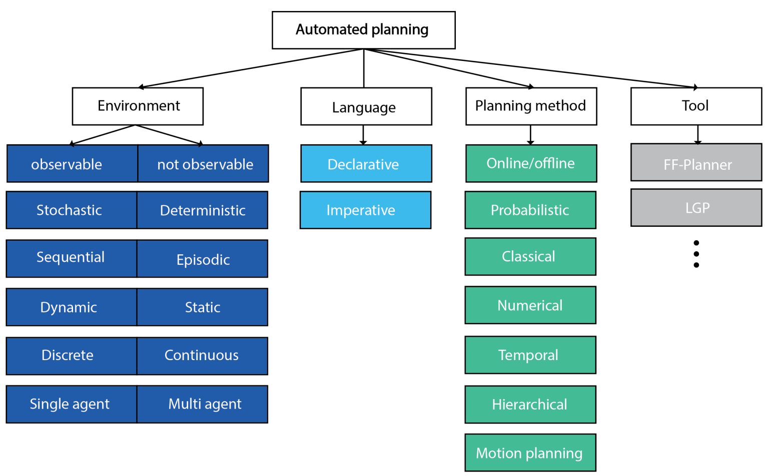

Fig. 2. The categories used in this paper to analyse ATP techniques based on concepts presented in [22, 35, 44]

3 RELATED WORKS

To determine the current state of research on symbolic AI planning methods in CF, we analyzed three categories of review papers: (1) systematic reviews, (2) systematic mapping, and (3) scientometric analysis. A systematic review summarizes and synthesizes existing evidence on a specific research question or topic. Systematic mapping provides an overview or "map" of the existing literature on a broad topic or research area. The scientometric analysis focuses on analyzing the quantitative aspects of scientific publications, such as citations, co-authorship networks, and publication trends. In the following paragraphs, we discuss the state of the art of literature on symbolic AI in CF, focusing on methods used for planning and scheduling: CBR, expert systems, and automated planning.

Automated methods and systems for construction planning and scheduling have been reviewed by [2]. This paper identifies the following major challenges with automation in planning with current methods: lack of flexibility in the storage of construction knowledge within existing templates, (2) reliance on manually created and maintained work templates and lack of learning techniques, (3) limited validation of the applicability of existing automated planning systems in real-life construction projects, and (4) disconnected research approaches in automated planning and schedule optimization.

This paper also explores expert systems as the earliest attempt to automate construction planning and scheduling while addressing knowledge-based systems' limitations: formal representation, automation in structuring and reusing previous projects, and rigidity of work templates [2]. Regarding knowledge formalization, current methods use unstructured text or object-oriented representation for activity descriptions. Unstructured methods lack semantic representation and specificity, while in object-oriented methods, activities are part of a hierarchy of classes that share similar attributes [2].

Paper [15] categorized the automation approaches for construction scheduling into CBR, knowledge-based (KB), model-based, genetic algorithms, expert systems, and neural networks. CBR and expert systems were analyzed for their use cases in scheduling in CF, including tools and applications. Paper [55] also looked into the use cases of CBR for safety and risk assessment and management. However, these reviews did not cover ATP techniques and task planning.

AI in the construction industry has been reviewed in [1], which outlines applied subfields of AI along with their potentials and limits in CF fields. Knowledge-based systems, including expert systems, CBR, and intelligent tutoring systems, are briefly described and introduced. According to [1] ML techniques have surpassed KB systems in popularity over the last decade (2010-2020). Paper [38] provides an overview of AI adaptation in construction, engineering, and management while categorizing developed AI techniques into four groups: expert systems, fuzzy logic, ML, and optimization algorithms. The paper concludes that the research focus has shifted from expert systems to building information modeling (BIM) and digital twin technology. Automated planning is not a part of current or future research directions.

Paper [12] conducted a scientometric analysis to determine the state of current AI research in the CF field. They found that genetic algorithms, neural networks, and fuzzy logic have the most interest rates. AI has primarily been used for optimization, simulation, uncertainty, and project management in the CF field. However, planning and expert systems have low interest. A list of under-researched keywords was provided, without a mention of automated planning, which is again not included in a table called "AI Techniques Applicable to the CF". Paper [8] conducted a bibliometric review on AI in construction engineering and management and identified construction project management, construction engineering, and estimation of cost control as the major fields of AI application in construction. The most applied techniques were evolutionary computation, fuzzy set theory, and CBR. Automated planning was not included in the keywords. Paper [32] "Intelligent Systems Research in the Construction Industry" covered the use of CBR and expert systems in the construction industry.

To summarize, from symbolic AI planning methods (expert systems, CBR, and ATP), expert systems and CBR have been explored in detail, along with other knowledge-based tools and techniques for planning in CF. However, the reviews have yet to acknowledge and search for automated planning methods, and thoroughly reviewing ATP techniques in the CF is crucial.

4 METHODOLOGY

We investigate the current state of the use of ATP in CF. To this end, we answer the following research questions (RQs) (Section 1):

(1) In which type of environment planning has been done?
(2) What languages and tools have been used to define the construction domain?
(3) Which AI planning techniques have been used to automate the planning?
(4) What are the use cases of automated task planning in CF, and how can the results be evaluated?
(5) What can be understood from the analytic data such as country, year, and authors of the papers?

To answer these questions, we conduct a systematic literature review (SLR) using the preferred reporting items for systematic reviews and meta-analysis (PRISMA) method [36] (Figure 3).

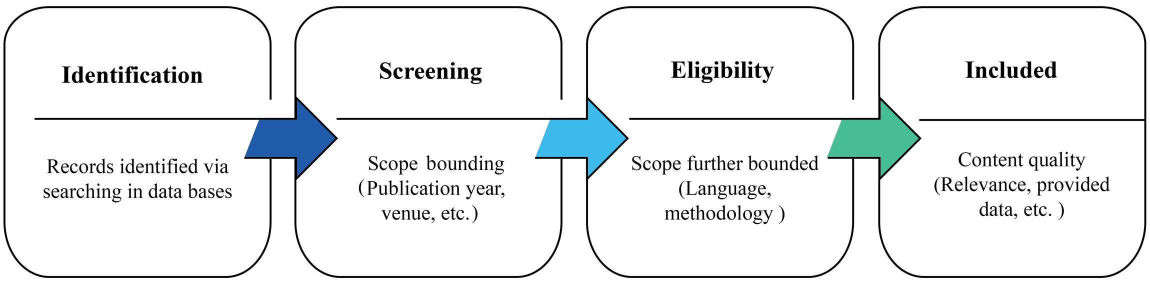

Fig. 3. Main steps of the PRISMA method [36]

To establish a clear scope, we explain the topics we included and excluded. We included the ATP use cases in CF planning or scheduling, specifically ATP techniques according to [22], as explained in the background and introduction. Additionally, we include task and motion planning (TAMP),which is a field in automated planning and robotics that combines task planning and motion planning. We exclude other AI methods for planning and scheduling in construction, such as deep learning, neural networks, CBR, and expert systems, because these topics, along with their limitations and state of the art, have been explored in other review papers (Section 3). We searched the data bases of WOS, Scopus, SpringerLink, ACM digital library, and IEEE. We also used the snowballing technique in the references of included papers but found no additional papers to include. The search query is shown in Table 2 and Figure 4:

Technique keywords	((pddl* OR hddl OR htn OR mdp OR pomdp OR rddl OR mapddl OR anml OR rmpl OR nddl OR ppddl OR tamp OR graphplan) OR ((Plan* OR schedul*) AND (AI OR “artificial intelligence”)))
domain keywords	(“civil engineering” OR “construction management” OR “construction engineering” OR “construction industry*” OR “construction plan*” OR “construction project*” OR fabrication)
search Query	(((pddl* OR hddl OR htn OR mdp OR pomdp OR rddl OR mapddl OR anml OR rmpl OR nddl OR ppddl OR tamp OR graphplan) OR ((Plan* OR schedul*) AND (AI OR “artificial intelligence”))) AND (“civil engineering” OR “construction management” OR “construction engineering” OR “construction industry*” OR “construction plan*” OR “construction project*” OR fabrication))

Table 2. Search query used in data bases for acquiring papers

We selected specific keywords, including common planning language names as well as general keywords like "plan*" and "AI" to ensure comprehensive coverage of relevant papers. We also utilized the search filters to ensure the relevance of the results (Table 3).

Data bases	*Categories*	*Publication type*	*Language*	*Year*	
WOS	*Included:* *Engineering Civil* *Engineering Electrical Electronic* *Computer Science Artificial Intelligence* *Multidisciplinary Sciences* *Construction Building Technology* *Automation Control Systems* *Robotics* *Computer Science Interdisciplinary Applications* *Engineering Multidisciplinary* *Computer Science Theory Methods* *Engineering Manufacturing* *Engineering Industrial* *Computer Science Information Systems* *Engineering Mechanical* *Computer Science Software Engineering* *Computer Science Hardware Architecture* *Architecture* *Mechanics* *Engineering Environmental* *Regional Urban Planning* *Art* *Geriatrics Gerontology* *Gerontology*	*article* *proceeding paper* *review article*	*English*	*01.01.2002-31.12.2022*	*title,abstract,keywords*
SCOPUS	*Included:* *Engineering* *Computer Science* *Energy* *Multidisciplinary* *Decision Sciences*	*conference paper* *article* *review*	*English*	*01.01.2002-31.12.2022*	*title,abstract,keywords*
IEEE	-	*conferences* *journals*	*English*	*01.01.2002-31.12.2022*	*metadata*
ACM	-	*proceedings* *journals*	*English*	*01.01.2002-31.12.2022*	*abstract, title*
Springerlink	*Included:* *subdecipline: civil engineering*	*article* *conference paper*	*English*	*01.01.2002-31.12.2022*	-

Table 3. Filters used in each data base

We removed duplicates, papers with less than four pages, and excluded irrelevant titles after obtaining search results from 3710 papers (Figure 4). We then reviewed abstracts and excluded irrelevant papers. Finally, we reviewed introductions and conclusions to identify relevant papers. Six papers were relevant in the end. Figure 4 displays the search results and the number of papers excluded at each step.

5 FINDINGS

This section analyzes the remaining papers based on environmental characteristics, language, tool, and planning techniques.

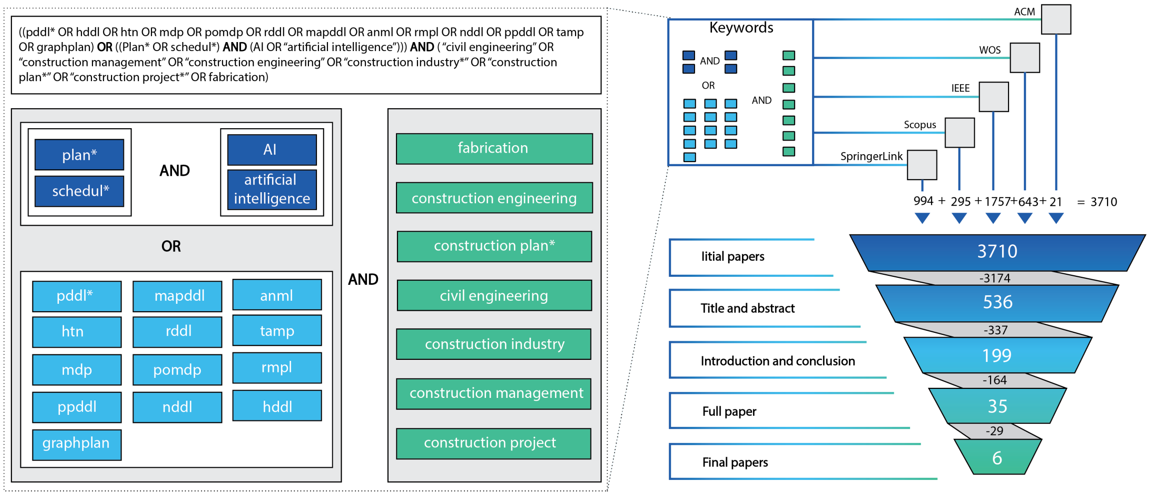

Fig. 4. Keywords, the data bases, and the filters used for paper exclusion

5.1 Environmental Characteristics

This section answers the RQ1 regarding the environmental characteristics of the reviewed papers.

5.1.1 Observability. Papers [9, 23, 24, 28] have used a simulation environment in which the agent can receive the relevant data from the environment. Hence, they have all been done in an observable environment.

Papers [23, 24] are also observable because they did planning in a simulation environment, and no noise was present in the movement of all robotic agents. Paper [9] also develops a simulation model to create a virtual environment showing the work site, construction structures, and two virtual cranes.

The paper [29] uses a simulation platform to address robot reachability analysis and collision-free motion (e.g., collision checking, forward kinematics, and visualization during motion planning).

Paper [29] has also done a real-world case study in a fabrication environment. In their case study, a task involves the robotic arm connecting and disconnecting a clamp on a structure. This movement can be difficult because the clamp may shift during the clamping process. To address this, they installed a camera on the robot's flange to capture an image and identify any misalignment. The case study environment is observable, as noises are not mentioned in the camera input. On the other hand, papers [28, 40] are not observable because no simulation environment, sensor, or agent was introduced, and the initial required data exists offline.

5.1.2 Deterministic/ stochastic. The environment is deterministic in [23, 24], as these papers use a simulation environment. Although [23] discusses stochastic restarts to overcome local optima and avoid infeasible configurations, the overall environment is still deterministic. Papers [28, 40] are also deterministic because no agent is involved, and the actions in these papers (converting data for data fusion and disassembly sequence of building components) are always successful. However, in paper [9], the environment is stochastic, and an agent in the environment communicates action results to the cranes. Paper [29] introduces a deterministic simulation environment, but manual intervention was required in the real-world case study due to encountered challenges during the building and plan execution. Therefore, we consider the environment in [29] stochastic.

5.1.3 Episodic/ sequential. All the six papers utilize sequential planning, as each action is dependent on the previous one. Specifically, papers [23, 24, 28, 29], involve assembling a structure, where each action heavily relies on the one preceding it. Paper [40] is also focused on finding the correct sequence of actions for data fusion, and paper [9] emphasizes the importance of sequential planning, as two agents must communicate and collaborate.

5.1.4 Static/dynamic. Regarding dynamic environments, paper [9] focuses on planning using real-time information updates. The paper also explores the possibility of searching for new paths to re-plan when necessary. On the other hand, paper [24] proposes embedding planning in space-time [20, 46] to enable planning in dynamic and uncertain environments. This approach is particularly useful in assembly tasks with multiple agents where agents may block goal regions and multiple keyframes need to be sampled at different time intervals. However, the remaining papers in the literature review mainly address planning in static environments. In these papers, planning either happens in a controlled or simulated environment [23, 29] or in a static environment [28, 40].

5.1.5 Single/ multi agent. Papers [9, 23, 24] have planned using multiple agents. In Paper [23], two robots - a crane and a mobile robot - are coordinated for final placement. The crane agent is responsible for lifting the parts and moving them to the handover position, and the mobile robot then positions and places the parts (Figure 5). Paper [9] describes a system with three agents: two cranes and one site-state agent, each one capable of sending messages, receiving messages, and making decisions.

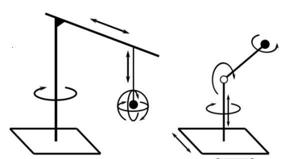

Fig. 5. The crane and the mobile robot with six degrees of freedom in paper [23]

The site-state agent monitors the on-site situation and collects information from the crane agents and the work site environment model (Figure 6) . In paper [24], the authors discuss how multiple robots can be utilized in planning

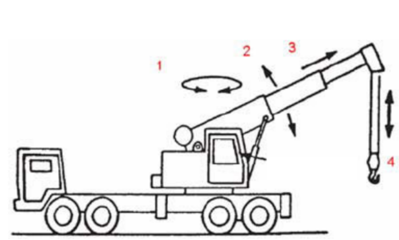

Fig. 6. The crane agent and its degrees of freedom in paper [9]

for building constructions. They showcase their approach by using three types of robots: a mobile manipulator, a KUKA-arm mounted on a mobile base, and a crane (Figure 7). The authors demonstrate their methodology through various scenarios, such as a well (using six agents), a pavilion (using eight agents), and a tower. In the example of the tower, the KUKA-arms are unable to reach the top of the tower. To solve this issue, a tower crane is added. However,

Papers	Observability	Deterministic (D)/ Stochastic (S)	Episodic (E) Sequential (S)/	Static (S)/ Dynamic (D)	Continuous (C)/ Discrete (D)	Agents
[23]	O	D	S	S	D & C	M
[40]	NO	D	S	S	D & C	no agent
[28]	NO	D	S	S	D	no agent
[9]	O	S	S	D	D & C	M
[29]	O	D and S	S	S	D & C	S
[24]	O	D	S	D	D & C	M

Table 4. The environmental characteristics of the reviewed paper. Observability: NO = not observable, O = observable. Agents: M = multi agent, S = single agent

the crane cannot access the pieces lying on the floor, which makes handover sequences necessary to place the last parts in their position. Papers [28, 40] have not introduced an agent and paper [29] uses a single industrial robotic arm.

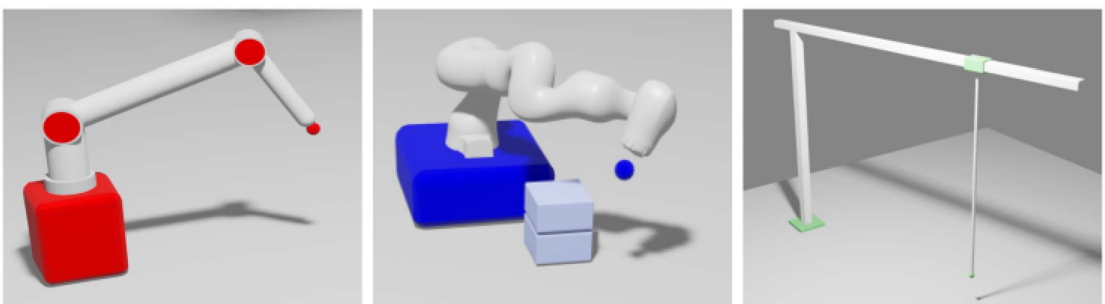

Fig. 7. A KUKA arm, a mobile base and a crane used as agents in paper [24]

5.1.6 Continuous/ discrete. In paper [28], only actions with discrete variables were used, as the variables represented building parts. However, papers [9, 23, 24, 29, 40] utilize both continuous and discrete variables. In papers [23, 24], task and motion planning were integrated into a single framework (TAMP). They used logic geometric programming, which introduces a discrete variable $s \in S$ (the state of a symbolic domain S) that parametrizes the costs and constraints of a nonlinear trajectory optimization problem over the (continuous) path x. On the other hand, paper [29] used a flowchart-based method for task planning and then called motion planners utilizing the Compass framework. However, manual changes in the tasks during plan execution were possible. Paper [9] combines path and task planning where task planning is first conducted, followed by path planning. The path planner reads the goal description and searches for possible paths while adhering to predefined rules to avoid obstacles and meet engineering or safety constraints. Paper [40] utilizes both discrete and continuous variables, with the latter taking the form of temporal data. Table 4 gives a summary of discussed environmental characteristics.

5.2 Language and Tool

This section answers the RQ2 regarding languages. Paper [23] utilizes a "first-order language similar to PDDL." They have also provided a list of the actions and predicates that have been used in the planning process. The actions that can be performed include pick, place, retract, and handover. The same language has also been used in [23] to define the set S(R, O). This set represents all legitimate state sequences for both robots (R) and objects (O). The language mentioned corresponds to the LGP planner that they are developing. This planner utilizes an optimization-based approach for long horizon combined task and motion planning [49].

The LGP method and tool apply optimization techniques to solve task and motion planning (TAMP) problems. It is a non-linear constrained program that considers the full world trajectory, where the symbolic state-action sequence defines the (in-)equality constraints [49]. One challenge of this method is its ability to scale up to long-horizon planning problems, but this is being improved in papers [5, 23, 24]. However, LGP can tackle specific sets of problems, such as maximizing the height of a physically stable construction [5]. The inputs of this tool include a cost function and a set of symbolic, symbolic goal state, initial state, initial kinematic state and a horizon. The output of this tool is a sequence of robot manipulations that can be used to reach the goal [5].

Paper [9] uses knowledge query and manipulation language (KQML) [17] for agents to exchange messages transmitted between the crane agents and the site agents. The movements of a crane are broken down into a sequence of actions. Using a hydraulic crane as an example, the crane movement consists of the following actions: BaseMove, BaseStop for base movement; BoomRaise, BoomLower, BoomExtend, BoomRetract, BoomSwing, BoomStop for boom movement; and HookHoist, HookLower, HookStop, HookGrip, HookRelease for hook movement.

The cranes use a traditional AI technique called coordination-by-planning, which involves planning and execution phases to achieve a shared plan. In the planning phase, a set of plans is produced, including a set of actions to be carried out by agents to achieve a goal [16]. Also, a genetic algorithm is used to find the near-optimal plan for individual crane agents. In the execution phase, the shared plan is executed. The cranes work together through messages: one crane has priority in proposing actions, while the other can accept or reject the proposals to create a shared plan. The process of using cranes to lift heavy objects involves generating a sequence of actions based on inputs like goal description, knowledge base (crane model, engineering constraints, criteria for plan generation, and rules for actions), and environment model (static and dynamic objects and data collected by sensors). In this method, real-time coordination and multi-agent planning are advantages, but autonomy varies between agents [9].

In paper [29], the robot and tool descriptions are created using the unified robot description format (URDF) and meshes (.obj and .stl), while the assembly description, robotic configuration, and geometry classes are based on the Compass framework [7, 19]. Python language was used to develop the task planning, action to movement decomposition, and compiler.

Creating a task sequence is done through a flowchart, which can be created either manually or using heuristics. Flowchart tasks are user-defined and can consist of one or more atomic movements. The flowchart also includes conditional statements to trigger different actions based on specific properties of each step. This method has the advantage of focusing on high-level actions before dealing with their low-level implementation. However, while the flowchart makes it easy for fabricators to adjust and sequence tasks, planning languages like PDDL are not used, and planners are not used for task planning.

Paper [40] utilized JavaScript and an algorithm called GraphPlan, commonly used in the ATP community. A Proposition class represents propositions, and an Action class represents actions. To capture different classifications of actions and propositions, the existing Graphplan algorithm was extended by defining new data structures (classes) because they believe the existing representation, e.g., action description language (a planning language), is insufficient. This method requires an initial state, a goal state, and a set of actions as input and produces a task plan as output. However, they created their own planner instead of using established planners that implement the GraphPlan algorithm due to the absence of a planning language.

The authors of [28] developed a graph using simple calculus that illustrates the starting point of a building, as a connection graph that displays all the necessary relationships between components. The end goal is to disassemble all prefabricated building components, represented by a set of disconnected pieces. The main goal of disassembly planning is to establish precedence relationships between disassembly operations to achieve the desired transformation from the initial state to the end goal state. Table 5 gives a summary of the languages and tools used for planning.

Papers	*Declarative*	*Imperative*	*Language*	*Tools/algorithms*
[23]	*Yes*		*Declarative, a first-order logic language (similar to PDDL)*	*LGP Planner*
[40]		*Yes*	*JavaScript*	*Graphplan*
[28]	Yes		First order logic language	Simple calculus
[9]	Yes		KQML (Knowledge Query and Manipulation Language)	Coordination by planning
[29]	Yes	Yes	Python, xml	Pybullet, Compass framework. Rhino
[24]	Yes		A first-order logic language	LGP

Table 5. Languages and the tools used for ATP in the CF

5.3 Planning Method

This section answers RQ3 regarding planning methods as following:

Probabilistic planning: None of the papers utilizing ATP techniques have incorporated or utilized any probabilistic planning techniques.

Classical Planning: Paper [28] exemplifies classical planning, as it lacks continuous variables, probabilistic techniques, online planning, and hierarchy.

Numerical and temporal: Papers [23, 24] utilize tree search and optimization techniques that involve continuous variables, such as cost function and time. Paper [40] also includes continuous variables and time. Paper [9] involves numerical variables such as the current position of crane components (e.g., boom length, boom angle, and hook location) and time (duration between start and end time of work). Paper [29] considers numerical information for high-level task planning (such as the total number of elements).

Hierarchical: Paper [40] creates a planning algorithm that utilizes hierarchical task network (HTN) planning for data fusion. The HTN planner generates an abstract plan to combine numerous data sources as the initial step. This abstract plan comprises a series of compound tasks, each corresponding to the fusion of two data sources.

Online planning: Papers [23, 24] are considered online planning as they propose an iterative limited-horizon approach to solve long-horizon LGPs, where formulating subgoals is important. Additionally, paper [9] introduces a real-time planner where a site-state agent collects information from crane agents while working and the work site environment model (including static and dynamic information).

Motion Planning: Papers [23, 24, 29] include motion planning for their robotic agents. Path planning has been used in paper [9]. Table 6 shows a summary of the planning techniques that have been used.

Papers	Probabilistic	Classical	Numerical	Temporal	Hierarchical	Online	Motion
[23]	N	N	Y	Y	N	Y	Y
[40]	N	N	N	Y	Y	N	N
[28]	N	Y	N	N	N	N	N
[9]	N	N	Y	Y	N	Y	Y
[29]	N	N	Y	N	N	N	Y
[24]	N	N	Y	Y	N	Y	Y

Table 6. The use of different ATP techniques in the reviewed papers. Y = Yes, N = No

5.4 Evaluation of Results

This section answers RQ4. Based on the TRL scale, we have categorized papers into three categories: research, development, and deployment. None of the papers have a TRL of more than six, indicating that none of the planning methods have been used on an actual construction site or for a real construction problem at the deployment level. Papers [23, 24, 28, 40] are at the research level as they have only been done in a simulation environment or at the conceptual level. Papers [9, 29] are in the development phase, with paper[29] testing the project in a fabrication setup and paper [9] testing the research in an actual lifting situation (Table 7).

Papers	Phase	TRL
[23]	construction	Research
[40]	preconstruction (planning)	Research
[28]	preconstruction (planning)	Research
[9]	construction	Development
[29]	construction	Development
[24]	construction	Research

Table 7. Evaluation of the results of the reviewed papers based on the TRL scale

5.5 Statistical Information

This section answers RQ 5 regarding statistical information of reviewd papers. Based on the data in Table 8, research interest in ATP has increased over the last two years. The country that has contributed the most to this research is the United States. However, it is worth noting that there is a five-year gap (between 2012 and 2021) and a gap from 2000 to 2005, during which no papers were published on this topic.

Papers	Year	Country	Continent	Source	Authors
[23]	2020	Stuttgart,Germany	Europe	Scopus	Hartmann V.N., Oguz O.S., Driess D., Toussaint M., Menges A.
[40]	2012	Pennsylvania, USA	America	Scopus	Pradhan A., Akinci B.
[28]	2005	Shanghai, China	Asia	Scopus	Hu W.
[9]	2007	Quebec, Canada	America	IEEE	Cheng Zhang; Amin Hammad
[29]	2021	Boston, USA Zurich, Switzerland	America	ACM	Huang Y, Leung P.Y.V, Garrett C, Gramazio F, Kohler M, Mueller C
[24]	2022	Berlin, Germany	Europe	IEEE	Hartmann, Valentin N., Andreas Orthey, Danny Driess, Ozgur S. Oguz, and Marc Toussaint

Table 8. Statistical information of the reviewed papers

6 RESULT ANALYSIS

In this section, we give a synthesis of the results mentioned in the previous sections. Table 9 highlights research gaps regarding the environment and ATP techniques. CF environments are partially observable and dynamic due to reasons like sensor noises. Moreover, actions in CF environments are usually stochastic, requiring frequent replanning. Techniques like online planning and probabilistic planning can again be used to tackle this issue.

Therefore, techniques such as online planning and probabilistic planning can be beneficial in such environments. However, while papers [9, 29] have used online planning in a stochastic environment, no paper has yet leveraged probabilistic techniques.

As CF planning is typically sequential, the gap in the table for episodic environments is irrelevant. Regarding multi-agent planning, online and probabilistic techniques are essential if agents need to do collaborative work. Papers [9, 23, 24] have used multi-agent online planning.

Table 9 shows that only paper [40] has used hierarchical techniques, showing that the projects are small.

Table 10 illustrates the environmental characteristics, ATP techniques, and papers' results. The paper [29] has results in both the research and development stages. The environment in the development phase was stochastic but not dynamic, as it was done in a fabrication setup. Regarding paper [9], the environment was both stochastic and dynamic, which are essential for the development phase (Table 10).

Environmental characteristics	Planning techniques							
	Online	Offline	Probabilistic	Classical	Numerical	Temporal	Hierarchical	Motion Planning
Observable	[23][9][9] [24]				[23] [9] [9] [24]	[23] [9] [24]		[23] [9] [9] [24]
Not observable		[2][28]		[28]		[40]	[40]	
Partially observable								
Deterministic	[23] [9] [24]	[40][28]		[28]	[9] [24]	[40][24]	[40]	[23] [9] [24]
Stochastic	[9] [9]				[9] [9]	[9]		[9] [9]
Episodic								
Sequential	[23] [9] [9] [24]	[40][28]		[28]	[9] [9] [24]	[40][9] [24]	[40]	[23] [9] [9] [24]
Static	[23] [9]	[40][28]		[28]	[9]	[40]	[40]	[23] [9]
Dynamic	[9] [24]				[9] [24]	[9] [24]		[9] [24]
Continuous								
Discrete		[28]		[28]				
Discrete and continuous	[23] [9] [9] [24]	[40]			[9] [9] [24]	[40][9] [24]	[40]	[23] [9] [9] [24]
Single agent	[9]				[9]			[9]
Multi agent	[23] [9] [24]				[23] [9] [24]	[23] [9] [24]		[23] [9] [24]
No agent		[40][28]		[28]		[40]	[40]	

Table 9. Environmental characteristics and ATP techniques of the reviewed papers

7 THREATS TO VALIDITY

We examined potential threats to the validity of our review according to the four fundamental types of validity threats, as outlined by Wohlin et al. [51, 52]: construct validity (research design), internal validity (data extraction), conclusion validity (reliability), and external validity (generalizability). In the following sections, we will discuss each of these potential threats in detail.

Construct validity (research design): To ensure that our findings are widely applicable, we utilized five commonly used databases: WOS, Scopus, IEEE, Springerlink, and ACM digital library. We excluded Google Scholar's digital library as it includes non-peer-reviewed papers. Our search keywords include related terms such as the names of prominent planners and planning languages and algorithms, in addition to publications mentioning more general keywords like "Planning or scheduling" and "AI".

Environmental characteristics	TRL			Planning techniques	TRL		
	Research	Development	Deployment		Research	Development	Deployment
Observable	[23] [24]	[9] [9]		Online	[23][24]	[9][9]	
Not observable	[40][28]			Offline	[40][28]		
Partially observable				Probabilistic			
Deterministic	[23] [24] [40][28]	[9]		Classical	[28]		
Stochastic		[9] [9]		Numerical	[23] [24]	[9] [9]	
Episodic				Temporal	[23] [24] [40]	[9]	
Sequential	[23] [24] [40] [28]	[9] [9]		Hierarchical	[40]		
Static	[23] [40][28]	[9]		Motion planning	[23] [24]	[9] [9]	
Dynamic	[24]	[9]					
Continuous							
Discrete	[28]						
Discrete and continuous	[23] [24] [40]	[9] [9]					
Single agent		[9]					
Multi agent	[23] [24]	[9]					
No agent	[40][28]						

Table 10. Reviewed papers based on their their environmental characteristics, ATP techniques, and results

Internal validity (data extraction): During our data extraction process, we only included metadata such as title, abstract, and keywords. This was necessary to avoid irrelevant results due to the various meanings of terms such as "planning," "scheduling," and "AI" in the CF field. Additionally, to prevent the exclusion of relevant publications, we temporarily included papers we were uncertain of during the abstraction screening process, and ultimately decided on inclusion or exclusion after reading the full papers. We excluded papers less than 4 pages as they did not contribute sufficiently to our research and only considered conference papers and articles since other formats are not peer-reviewed. Note that our study is biased towards institutions and journals publishing in English, which may affect conclusions about the most active institutions and relevant journals. Additionally, our review is subject to publication bias as we only report on published results.

Conclusion validity (reliability): In terms of drawing incorrect conclusions, we have explored several factors that could potentially result in such conclusions within the framework of threats to internal validity. To ensure replicability, we have provided a comprehensive description of our research methodology in Section 4, which allows replication of every stage of this study.

External validity (generalizability): External validity is related to three criteria of the research: interaction of selection and treatment, interaction of setting and treatment, and interaction of history and treatment, all of which are not applicable to our research.

8 DISCUSSION

Based on the findings of the paper, it appears that the utilization of ATP techniques over the last twenty years is limited. The few papers that were available on the topic presented challenges with obtaining significant results and avoiding generalizations. However, these papers did highlight the significant gaps in the use of ATP techniques in the construction and fabrication industry.

Construction site environments are usually partially observable (due to noises in the sensors or obstacles obstructing the cameras or sensors), non-deterministic, sequential, and dynamic. Hence, none of the reviewed papers' results can be practically used in onsite construction. The prefabrication environment has the same environment except for dynamic, which can become static as fabrication usually happens in a controlled environment. Among the evaluated papers, paper [9] is the closest to meeting construction environment criteria, but it only deals with a small subproblem: picking an object with two cranes.

We also found that commonly used domain-independent languages, such as PDDL, have yet to be utilized in defining construction and fabrication domain problems, resulting in many unused generic planners. Subsequently, many planning techniques and corresponding languages and planners have not been explored.

There are several potential reasons for this lack of exploration: Symbolic AI methods, or "good old fashioned" AI methods, are less popular than subsymbolic AI methods, such as ML, which means many CF practitioners might not be aware of ATP techniques and their benefits. Additionally, many CF practitioners are not familiar with declarative planning languages like PDDL. The lack of a user-friendly interface for editing domains and problem models has probably also contributed to the challenge (the planning community has made progress in creating a user-friendly editor, see [20]).

We provide a roadmap for researchers to explore the use of ATP techniques. To this end, researchers first need to identify the problem environment, select the appropriate technique and language, test existing languages and techniques [45], and choose or develop a planner accordingly to solve their problem. Table 11 categorizes related environment characteristics, languages, techniques, and tools. While this table is not comprehensive, it provides a starting point, including valuable resources and keywords for several research roads.

Additionally, research can be done on symbolic AI planning together with learning methods (subsymbolic) [3, 41]. To fully unravel the potentials of ATP, future research should investigate the advantages and limitations of existing ATP techniques, tools, and languages and address them in the context of CF planning processes.

Environment characteristics	*Declarative languages*	*Technique*	*Tools (Planners)*
Partially observable Dynamic Stochastic Online/offline	PPDDL[53] RDDL[41, 43]	Probabilistic/conditional planning: Partially observable markov decision process (POMDP) [21, 54] Markov decision process (MDP)[21]	SPUDD[26] PROST[33, 34]
Static Observable Deterministic Discrete	PDDL[25] STRIPS[42] ADL[39]	Classical planning [25]	Metric-FF[27]
Continuous Observable Deterministic	PDDL 2.1 [25]	Numerical and temporal planning [42]	VHPOP [53] CRIKEY[10] Metric-FF [27] TLPlan [13]
Static Observable Deterministic Discrete	HDDL [31]	Hierarchical planning, Hierarchical task networks (HTN)[14]	SHOP2 [4] PANDA [30]
Multi-agent Online/offline	MA-PDDL [18]	Multi-agents planning [47]	MAPlan [56] PSM planner [50] FMAP [48]

Table 11. A roadmap for future research showing related environment characteristics, languages, techniques, and tools

9 CONCLUSION

This paper presents a systematic review of the use of AI automated planning techniques for task planning in the field of CF to evaluate their current state of research and provide a research roadmap and gaps for future research. The automation of construction planning faces challenges regarding knowledge flexibility, knowledge formalization and maintenance, and the decoupled nature of planning and scheduling. ATP techniques offer various languages and algorithms to model different problems, and planning and scheduling are not separated. However, these techniques have not been evaluated for the CF processes. This review focuses on automated task planning techniques, the important aspects to consider when using them, and specific research gaps.

The current literature on the use of AI in the CF industry can be divided into three categories: systematic mappings, scientometric analysis, and systematic reviews. However, none of these reviews have focused specifically on ATP techniques in CF field. Therefore, this paper aims to fill this gap by reviewing papers on AI task planning techniques and answering important questions related to the (1) planning environment, (2) modeling languages and tools, (3) ATP techniques, and (4) results. Following paragraphs summarize the answers to the research questions of this paper:

- Our survey highlights that all of the papers have been planned in an observable environment except for the ones with no agent in which observability is irrelevant. The majority of the papers are planned in a deterministic and static environment with consideration for both continuous and discrete variables. All papers were planned in a sequential environment, and most introduced multiple agents.
- Regarding planning languages, KQML and a first-order logic language like PDDL have been used as declarative languages, while JavaScript and Python were used as imperative languages. LGP and Compass are the only tools that have been used, with LGP being the only ATP planner.
- Probabilistic planning has not been explored in any paper. Only one paper has used classical planning. Half of the papers have used numerical, temporal, and online planning simultaneously. All the papers introducing an agent have also used motion or path planning, and only one paper has used hierarchical planning.
- Most of the papers are in the research phase, and no paper has made it to the deployment phase. Additionally, there has been an increase in the use of ATP techniques in the CF, as half of the papers have been published in the past two years.

To summarize, regarding classical planning, paper [28] has utilized simple calculus as their language and have dealt with deterministic, static, not observable, single agent, and discrete environments. On the other hand, papers [9, 23, 24, 29] have employed numerical planning techniques, all of which utilize declarative languages in observable, multi-agent environments with both continuous and discrete variables. Finally, hierarchical planning has only been conducted using JavaScript [40]. Examples of online task planning are conducted in observable environments utilizing declarative languages (Table 12).

papers	ATP techniques	Language	Environmental characteristics
[23]	Numerical,temporal,online,motion planning	declarative	observable, deterministic, sequential, static, discrete & continuous, multi agent
[40]	Temporal, hierarchical	imperative	not observable, deterministic, sequential, static, discrete & continuous, no agent
[28]	Classical	declarative	not observable, deterministic,sequential, static, discrete, no agent
[9]	Numerical, temporal,online,motion planning	declarative	observable,stochastic, sequential, dynamic, discrete & continuous, multi agent
[29]	Numerical,motion planning	declarative and imperative	observable, deterministic and stochastic, sequential, static, discrete & continuous, single agent
[24]	Numerical,temporal,online,motion planning	declarative	observable, deterministic, sequential, dynamic, discrete & continuous, multi agent

Table 12. Summarizing the results of the reviewed papers based on the ATP techniques they employed, the languages used, and the environmental characteristics

Acronym	Full phrase	Acronym	Full phrase
AI	Artificial Intelligence	TAMP	Task and Motion Planning
ATP	Automated Task Planning	ACM	Association for Computing Machinery
CF	Construction and Fabrication	WOS	Web Of Science
CBR	Case-Based Reasoning	IEEE	Institute of Electrical and Electronics Engineers
ML	Machine Learning	TRL	Technology Readiness Level

Table 13. Acronyms used in the paper

ACKNOWLEDGMENTS

This research was funded by the Ministry of Science, Research and the Arts Baden-Wuerttemberg in the Artificial Intelligence Software Academy (AISA) and supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2120/1 - 390831618.

REFERENCES

[1] Sofiat O. Abioye, Lukumon O. Oyedele, Lukman Akanbi, Anuoluwapo Ajayi, Juan Manuel Davila Delgado, Muhammad Bilal, Olugbenga O. Akinade, and Ashraf Ahmed. 2021. Artificial intelligence in the construction industry: A review of present status, opportunities and future challenges. *Journal of Building Engineering* 44 (Dec. 2021), 103299. https://doi.org/10.1016/j.jobe.2021.103299

[2] Fouad Amer, Hui Yi Koh, and Mani Golparvar-Fard. 2021. Automated Methods and Systems for Construction Planning and Scheduling: Critical Review of Three Decades of Research. *Journal of Construction Engineering and Management* 147, 7 (July 2021), 03121002. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002093

[3] Masataro Asai and Alex Fukunaga. 2018. Classical Planning in Deep Latent Space: Bridging the Subsymbolic-Symbolic Boundary. *Proceedings of the AAAI Conference on Artificial Intelligence* 32, 1 (April 2018). https://doi.org/10.1609/aaai.v32i1.12077

[4] T. C. Au, O. Ilghami, U. Kuter, J. W. Murdock, D. S. Nau, D. Wu, and F. Yaman. 2011. SHOP2: An HTN Planning System. (2011). https://doi.org/10.48550/ARXIV.1106.4869

[5] Cornelius V. Braun, Joaquim Ortiz-Haro, Marc Toussaint, and Ozgur S. Oguz. 2021. RHH-LGP: Receding Horizon And Heuristics-Based Logic-Geometric Programming For Task And Motion Planning. (2021). https://doi.org/10.48550/ARXIV.2110.03420

[6] Erik Cambria, Yang Li, Frank Z. Xing, Soujanya Poria, and Kenneth Kwok. 2020. SenticNet 6: Ensemble Application of Symbolic and Subsymbolic AI for Sentiment Analysis. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. ACM, Virtual Event Ireland, 105–114. https://doi.org/10.1145/3340531.3412003

[7] Gonzalo Casas, Romana Rust, Beverlylytle, Anton Tetov, Matteo Pacher, Kathrin Dörfler, Chen Kasirer, Inés Ariza, Edvard Bruun, Yijiang Huang, ZhaoMA, Stefana Parascho, Achilleas Xydis, Feihln, DavidJenny, G.A. Vd. Hoorn, ClaimHF, and Kunal Chadha. 2023. compas-dev/compas_fab: COMPAS FAB v0.28.0. https://doi.org/10.5281/ZENODO.3469478

[8] Xiao Chao and Liu Yang. 2018. Bibliometric Review of Artificial Intelligence (AI) in Construction Engineering and Management.

[9] Cheng Zhang and Amin Hammad. 2007. Agent-based simulation for collaborative cranes. In *2007 Winter Simulation Conference*. IEEE, Washington, DC, USA, 2051–2056. https://doi.org/10.1109/WSC.2007.4419836

[10] Andrew Coles, Maria Fox, Keith Halsey, Derek Long, and Amanda Smith. 2009. Managing concurrency in temporal planning using planner-scheduler interaction. *Artificial Intelligence* 173, 1 (Jan. 2009), 1–44. https://doi.org/10.1016/j.artint.2008.08.003

[11] Susan Craw. 2017. Case-Based Reasoning. In *Encyclopedia of Machine Learning and Data Mining*, Claude Sammut and Geoffrey I. Webb (Eds.). Springer US, Boston, MA, 180–188. https://doi.org/10.1007/978-1-4899-7687-1_34

[12] Amos Darko, Albert P.C. Chan, Michael A. Adabre, David J. Edwards, M. Reza Hosseini, and Ernest E. Ameyaw. 2020. Artificial intelligence in the AEC industry: Scientometric analysis and visualization of research activities. *Automation in Construction* 112 (April 2020), 103081. https://doi.org/10.1016/j.autcon.2020.103081

[13] Tomas de la Rosa and Sheila McIlraith. 2011. Learning domain control knowledge for TLPlan and beyond. In *ICAPS 2011 Workshop on Planning and Learning*. 36–43.

[14] Mohamed Elkawkagy, Bernd Schattenberg, and Susanne Biundo. 2010. Landmarks in Hierarchical Planning. IOS Press 2010. https://doi.org/10.3233/978-1-60750-606-5-229

[15] Vahid Faghihi, Ali Nejat, Kenneth F. Reinschmidt, and Julian H. Kang. 2015. Automation in construction scheduling: a review of the literature. *The International Journal of Advanced Manufacturing Technology* 81, 9-12 (Dec. 2015), 1845–1856. https://doi.org/10.1007/s00170-015-7339-0

[16] Jacques Ferber. 1999. *Multi-agent systems: an introduction to distributed artificial intelligence.* Addison-Wesley, Harlow.

[17] Tim Finin, Richard Fritzson, Don McKay, and Robin McEntire. 1994. KQML as an agent communication language. In *Proceedings of the third international conference on Information and knowledge management - CIKM '94*. ACM Press, Gaithersburg, Maryland, United States, 456–463. https://doi.org/10.1145/191246.191322

[18] Daniel Fišer and Antonín Komenda. 2018. Concise Finite-Domain Representations for Factored MA-PDDL Planning Tasks:. In *Proceedings of the 10th International Conference on Agents and Artificial Intelligence*. SCITEPRESS - Science and Technology Publications, Funchal, Madeira, Portugal, 306–313. https://doi.org/10.5220/0006539503060313

[19] Philippe Fleischmann and Gonzalo Casas. 2021. compas-rrc/compas_rrc: COMPAS RRC v1.1.0. https://doi.org/10.5281/ZENODO.4639418

[20] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. 2021. Integrated Task and Motion Planning. *Annual Review of Control, Robotics, and Autonomous Systems* 4, 1 (May 2021), 265–293. https://doi.org/10.1146/annurev-control-091420-084139

[21] Hector Geffner and Blai Bonet. 2013. *A Concise Introduction to Models and Methods for Automated Planning*. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-031-01564-9

[22] Malik Ghallab, Dana S. Nau, and Paolo Traverso. 2004. *Automated planning: theory and practice*. Elsevier/Morgan Kaufmann, Amsterdam ; Boston.

[23] Valentin N. Hartmann, Ozgur S. Oguz, Danny Driess, Marc Toussaint, and Achim Menges. 2020. Robust Task and Motion Planning for Long-Horizon Architectural Construction Planning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Las Vegas, NV, USA, 6886–6893. https://doi.org/10.1109/IROS45743.2020.9341502

[24] Valentin N. Hartmann, Andreas Orthey, Danny Driess, Ozgur S. Oguz, and Marc Toussaint. 2023. Long-Horizon Multi-Robot Rearrangement Planning for Construction Assembly. *IEEE Transactions on Robotics* 39, 1 (Feb. 2023), 239–252. https://doi.org/10.1109/TRO.2022.3198020

[25] Patrick Haslum (Ed.). 2019. *An Introduction to the Planning Domain Definition Language*. Number 42 in Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan et Claypool Publishers, S.l.

[26] Jesse Hoey, Robert St-Aubin, Alan Hu, and Craig Boutilier. 2013. SPUDD: Stochastic Planning using Decision Diagrams. (2013). https://doi.org/10.48550/ARXIV.1301.6704

[27] J. Hoffmann. 2003. The Metric-FF Planning System: Translating "Ignoring Delete Lists" to Numeric State Variables. *Journal of Artificial Intelligence Research* 20 (Dec. 2003), 291–341. https://doi.org/10.1613/jair.1144

[28] Wenfa Hu. 2005. Automatic Construction Process of Prefabricated Buildings on Geometric Reasoning. In *Construction Research Congress 2005*. American Society of Civil Engineers, San Diego, California, United States, 1–10. https://doi.org/10.1061/40754(183)11

[29] Yijiang Huang, Pok Yin Victor Leung, Caelan Garrett, Fabio Gramazio, Matthias Kohler, and Caitlin Mueller. 2021. The new analog: A protocol for linking design and construction intent with algorithmic planning for robotic assembly of complex structures. In *Symposium on Computational Fabrication*. ACM, Virtual Event USA, 1–17. https://doi.org/10.1145/3485114.3485122

[30] Daniel Höller, Gregor Behnke, Pascal Bercher, and Susanne Biundo. 2021. The PANDA Framework for Hierarchical Planning. *KI - Künstliche Intelligenz* 35, 3-4 (Nov. 2021), 391–396. https://doi.org/10.1007/s13218-020-00699-y

[31] Daniel Höller, Gregor Behnke, Pascal Bercher, Susanne Biundo, Humbert Fiorino, Damien Pellier, and Ron Alford. 2020. HDDL: An Extension to PDDL for Expressing Hierarchical Planning Problems. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 06 (April 2020), 9883–9891. https://doi.org/10.1609/aaai.v34i06.6542

[32] Zahir Irani and Muhammad Mustafa Kamal. 2014. Intelligent Systems Research in the Construction Industry. *Expert Systems with Applications* 41, 4 (March 2014), 934–950. https://doi.org/10.1016/j.eswa.2013.06.061

[33] Thomas Keller and Patrick Eyerich. 2012. PROST: Probabilistic Planning Based on UCT. *Proceedings of the International Conference on Automated Planning and Scheduling* 22 (May 2012), 119–127. https://doi.org/10.1609/icaps.v22i1.13518

[34] Thomas Keller and Malte Helmert. 2013. Trial-Based Heuristic Tree Search for Finite Horizon MDPs. *Proceedings of the International Conference on Automated Planning and Scheduling* 23 (June 2013), 135–143. https://doi.org/10.1609/icaps.v23i1.13557

[35] John W. Lloyd. 1994. Practical advantages of declarative programming. In *Joint Conference on Declarative Programming*, Vol. 1. Servicio de Publicaciones de la Universidad Politécnica de Valencia, 3 – 17. https://www.programmazionelogica.it/wp-content/uploads/2015/12/GP1994-I-000-031.pdf

[36] David Moher. 2009. Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement. *Annals of Internal Medicine* 151, 4 (Aug. 2009), 264. https://doi.org/10.7326/0003-4819-151-4-200908180-00135

[37] Alison Olechowski, Steven D. Eppinger, and Nitin Joglekar. 2015. Technology readiness levels at 40: A study of state-of-the-art use, challenges, and opportunities. In *2015 Portland International Conference on Management of Engineering and Technology (PICMET)*. IEEE, Portland, OR, USA, 2084–2094. https://doi.org/10.1109/PICMET.2015.7273196

[38] Yue Pan and Limao Zhang. 2021. Roles of artificial intelligence in construction engineering and management: A critical review and future trends. *Automation in Construction* 122 (Feb. 2021), 103517. https://doi.org/10.1016/j.autcon.2020.103517

[39] Edwin P. D. Pednault. 1989. ADL: exploring the middle ground between STRIPS and the situation calculus. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 324–332.

[40] Anu Pradhan and Burcu Akinci. 2012. Planning-Based Approach for Fusing Data from Multiple Sources for Construction Productivity Monitoring. *Journal of Computing in Civil Engineering* 26, 4 (July 2012), 530–540. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000155

[41] Dongning Rao and Zhihua Jiang. 2015. Learning Planning Domain Descriptions in RDDL. *International Journal on Artificial Intelligence Tools* 24, 03 (June 2015), 1550002. https://doi.org/10.1142/s0218213015500025

[42] Stuart J. Russell, Peter Norvig, and Ernest Davis. 2010. *Artificial intelligence: a modern approach* (3rd ed ed.). Prentice Hall, Upper Saddle River.

[43] Scott Sanner. 2011. Relational Dynamic Influence Diagram Language (RDDL): Language Description. (2011). http://users.cecs.anu.edu.au/~ssanner/IPPC_2011/RDDL.pdf

[44] Md Kamruzzaman Sarker, Lu Zhou, Aaron Eberhart, and Pascal Hitzler. 2021. Neuro-Symbolic Artificial Intelligence: Current Trends. (2021). https://doi.org/10.48550/ARXIV.2105.05330

[45] Shermin Sherkat, Lior Skoury, Andreas Wortmann, and Thomas Wortmann. 2023. Artificial Intelligence Automated Task Planning for Fabrication. In *Advances in Architectural Geometry 2023*, Kathrin Dörfler, Jan Knippers, Achim Menges, Stefana Parascho, Helmut Pottmann, and Thomas Wortmann (Eds.). De Gruyter, 249–260. https://doi.org/10.1515/9783111162683-019

[46] Shermin Sherkat, Andreas Wortmann, and Thomas Wortmann. 2022. Potentials of Symbolic AI Planning for Construction. In *Proceedings of 33. Forum Bauinformatik*. Munich,Germany, 309–316. https://doi.org/10.14459/2022md1686600

[47] Alejandro Torreño, Eva Onaindia, Antonín Komenda, and Michal Štolba. 2018. Cooperative Multi-Agent Planning: A Survey. *Comput. Surveys* 50, 6 (Nov. 2018), 1–32. https://doi.org/10.1145/3128584

[48] Alejandro Torreño, Eva Onaindia, and Óscar Sapena. 2014. FMAP: Distributed cooperative multi-agent planning. *Applied Intelligence* 41, 2 (Sept. 2014), 606–626. https://doi.org/10.1007/s10489-014-0540-2

[49] Marc Toussaint. 2015. Logic-Geometric Programming: An Optimization-Based Approach to Combined Task and Motion Planning. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15)*. AAAI Press, 1930–1936. https://doi.org/10.5555/2832415.2832517 Place: Buenos Aires, Argentina.

[50] Jan Tožička, Jan Jakubův, Antonín Komenda, and Michal Pěchouček. 2016. Privacy-concerned multiagent planning. *Knowledge and Information Systems* 48, 3 (Sept. 2016), 581–618. https://doi.org/10.1007/s10115-015-0887-7

[51] Claes Wohlin, Per Runeson, Martin Höst, Magnus C. Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in Software Engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-29044-2

[52] Andreas Wortmann, Olivier Barais, Benoit Combemale, and Manuel Wimmer. 2020. Modeling languages in Industry 4.0: an extended systematic mapping study. *Software and Systems Modeling* 19, 1 (Jan. 2020), 67–94. https://doi.org/10.1007/s10270-019-00757-6

[53] H.L.S. Younes, M. L. Littman, D. Weissman, and J. Asmuth. 2005. The First Probabilistic Track of the International Planning Competition. *Journal of Artificial Intelligence Research* 24 (Dec. 2005), 851–887. https://doi.org/10.1613/jair.1880

[54] Shiqi Zhang, Mohan Sridharan, and Jeremy L. Wyatt. 2015. Mixed Logical Inference and Probabilistic Planning for Robots in Unreliable Worlds. *IEEE Transactions on Robotics* 31, 3 (June 2015), 699–713. https://doi.org/10.1109/TRO.2015.2422531

[55] Zhipeng Zhou, Yang Miang Goh, and Qiming Li. 2015. Overview and analysis of safety management studies in the construction industry. *Safety Science* 72 (Feb. 2015), 337–350. https://doi.org/10.1016/j.ssci.2014.10.006

[56] Michal Štolba, Antonín Komenda, and Daniel Kovacs. 2016. Competition of Distributed and Multiagent Planners (CoDMAP). *Proceedings of the AAAI Conference on Artificial Intelligence* 30, 1 (March 2016). https://doi.org/10.1609/aaai.v30i1.9870

Received 03 October 2023; accepted 09 April 2025