ASCE

# Robot Planning for Active Collision Avoidance in Modular Construction: Pipe Skids Example

Qi Zhu, S.M.ASCE[1]; Tianyu Zhou, S.M.ASCE[2]; Pengxiang Xia, S.M.ASCE[3]; and Jing Du, Ph.D., M.ASCE[4]

**Abstract:** Robot-assisted assembly has shown great potential in modular construction for the future. However, as the complexity of prefabricated structural modules increases, the likelihood of unexpected robotic collisions also rises, such as unexpected collisions between robots and structural modules during the assembly process. Most robotic collision avoidance methods rely on known robot specifications, especially the work envelope, that is, the range profiles of movement created by a robot arm moving forward, backward, up, and down. In contrast, it is unknown what models of robots will be applied for future modular construction because of the rapid development of robot mechanical designs. The deep uncertainties related to future robot work envelopes can challenge any effort for planned robotic collision avoidance for robot-assisted modular construction. There is a pressing need for a simulation-based robot collision avoidance planning method for future robot-assisted modular construction that captures the deep uncertainties of work envelopes of future robots. As a result, this paper presents an active robotic collision avoidance method, called collision-free workspace and collision-avoidance path planning (CWCP), to tackle the unique robotic collision challenges in modular construction. First, CWCP uses an inverse kinematics Monte Carlo (IKMC) simulation in the design phase of a pipe skid module to scan a large number of possible robotic movement trajectories and the corresponding boundaries of the robotic space. This result can be used to identify a robust structural design with enough clearance for most robotic actions. Then, immediately before an operation, physics engine simulation is used to optimize the waypoints of the robotic arm movement to further eliminate the collision possibility given the established structural design and robotic specifications. The proposed method was tested with a simulated pipe skid installation task. The result showed that the proposed method helped strike a balance between a safe-tolerant design and robotic planning on site and thus is more suited for future robot applications with no prior knowledge about the work envelope specifications.
**DOI: 10.1061/(ASCE)CO.1943-7862.0002374.** 

**Author keywords:** Modular construction; Robots; Inverse kinematics; Digital twins.



## Introduction

Robot-assisted modular construction refers to the use of robotic systems to support the construction of components that can be automatically produced (Mignacca et al. 2018). It is gaining growing interest for a safer and more cost-effective assembly (Tan et al. 2017; Shukla and Karki 2016). However, when applied in complex modular construction tasks, problems pertaining to robotic collisions can be a big challenge. For early adopters of robots, various types of collision incidents have frequently been reported, including robot–human collisions, robot–infrastructure collisions, and robot–robot collisions (Bragança et al. 2019). In order to solve possible collision problems, proactive collision avoidance has been proposed (Haddadin et al. 2017). This especially indicates that through delicate design of the spatial configuration, the majority of collision incidents could be prevented (Yang et al. 2010). Compared to passive collision avoidance methods that rely on real-time sensing and reactive mechanism designs, the proactive collision avoidance method is proven to be more robust to varying work contexts and workplaces.

However, the current proactive collision avoidance methods still face challenges, especially in the context of modular construction. Construction workplaces, including offsite for modular production, are usually dynamic, uncertain, and less controlled. The mixed space of human workers and moving construction materials, objects, and equipment make the all-inclusive comprehensive spatial configuration for collision-free designs nontrivial. Even for a relatively controlled modular construction process, the underbuilt infrastructure is still evolving over the course of a project, making a one-size-fits-all collision avoidance design process almost impossible. As a result, there is a need for examining a robust proactive collision avoidance method that can tackle the deep uncertainties with respect to the selection of robots, the configuration of workplaces, and operational workflows for safer robot-assisted modular construction.

This paper proposes an active robot collision avoidance method for robotic operations in modular construction, via the digital twins

[1]Ph.D. Candidate, Informatics, Cobots and Intelligent Construction (ICIC) Lab, Dept. of Civil and Coastal Engineering, Univ. of Florida, 1949 Stadium Rd. 360 Weil Hall, Gainesville, FL 32611. Email: qizhu@ufl.edu

[2]Ph.D. Candidate, Informatics, Cobots and Intelligent Construction (ICIC) Lab, Dept. of Civil and Coastal Engineering, Univ. of Florida, 1949 Stadium Rd. 360 Weil Hall, Gainesville, FL 32611. Email: zhoutianyu@ufl.edu

[3]Ph.D. Student, Informatics, Cobots and Intelligent Construction (ICIC) Lab, Dept. of Civil and Coastal Engineering, Univ. of Florida, 1949 Stadium Rd. 360 Weil Hall, Gainesville, FL 32611. Email: xia.p@ufl.edu

[4]Associate Professor, Informatics, Cobots and Intelligent Construction (ICIC) Lab, Dept. of Civil and Coastal Engineering, Univ. of Florida, 1949 Stadium Rd. 360 Weil Hall, Gainesville, FL 32611 (corresponding author). ORCID: https://orcid.org/0000-0002-0481-4875. Email: eric.du@essie.ufl.edu

Note. This manuscript was submitted on January 25, 2022; approved on May 19, 2022; published online on August 11, 2022. Discussion period open until January 11, 2023; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Construction Engineering and Management***, © ASCE, ISSN 0733-9364.

(DT) simulation, that is, creating a digital replica of the robot and its interactions with the workspace for optimization simulations. We focus on a two-step DT simulation strategy called collision-free workspace and collision-avoidance path planning (CWCP) planning for designing a collision-free workspace (CW) and collision-avoidance path planning (CP). This two-step strategy suits well the unique needs of modular construction operations at different phases of a project. The collision-free workspace design aims to identify a structural design, along with the workspace boundary design, to create sufficient clearance for all possible robotic systems to be deployed. It is conducted in the early phase of a project before the modular structural design is determined. At this point, the team should still have the chance to affect the workspace of the prospective construction operations to eliminate most possibilities of robotic collisions via enough tolerance and clearance in the spatial configuration. The collision-avoidance path planning will then optimize the robotic movement for a given modular construction workspace and robotic platform. It happens immediately before an operation, as the final check for active collision avoidance. The remainder of this paper will introduce the point of departure, a simulation study of the proposed method, and the lessons learned.

## Literature Review

### Robots in Modular Construction

Over the last few decades, modular construction has become popular in the construction industry because of its advantages of efficiency, safety, superior quality, and flexibility (Ferdous et al. 2019). According to a previous study (Jaillon et al. 2009), modular buildings have resulted in reductions both in construction time (70%) and labor cost (43%). Despite having well-documented benefits, modular construction only occupied a few proportions of the market compared to traditional construction methods (Jaillon et al. 2009).

The challenges of applying modular construction in real-life workspaces are multifaceted, including technical, financial, and regulatory barriers (Tam et al. 2007). However, the expeditious evolution of automated and robotic systems in construction alleviates the labor needs for assembling and installation and then speeds up the uptake rate of modular construction (Delgado et al. 2019). Single-task construction robots (STCRs) (Bock and Linner 2016), such as robotic arms executing a single task in a repetitive manner, have been directly used on construction sites for fabrication, building part assembly (Dritsas and Soh 2019), concrete laying (Więckowski 2017), and so on. Especially in the offshore environment, due to the difficulty in accessing subsea installations and relatively less flexible systems, robotic equipment assembly could be a better solution to both the need for efficiency and maximum production in this industry (Chen et al. 2014). On the onshore field, building of large platforms are challenges not only because of the environmental conditions (waves, wind, and current) but also the costs of keeping humans on these platforms, like the requirements of food, shelter, and safety (Shukla and Karki 2016).

Previous studies have proved the success of remotely controlled unmanned platforms that rely on the automation of offshore facilities (Skourup et al. 2008). In recent years, more advanced remote-controlled unmanned platforms have started to be equipped with redundant manipulators (Pinosofa et al. 2010). Compared to onshore, offshore facilities are highly constrained by the shape and size of floor space (Graf and Pfeiffer 2008). Manipulators in this environment have been used in both building processes like wall painting (Zhang et al. 2020), and welding (Lee et al. 2011), as well as facility management like mobile inspection (Bengel et al. 2009). However, all these robotic applications faced challenges of collision detection. Usually, robots rely on proximity sensors to avoid nearby walls and general obstacles. However, offshore installation is extremely complex because this environment is filled with pipe flanges, tanks, steel frames, ramps, and other complex-shaped structures, which makes them more difficult to detect by proximity sensors installed in the robot (Graf et al. 2007). Therefore, an improved collision avoidance strategy is needed to improve robotic construction performance in constrained environments such as offshore platforms.

### Robotic Collision Avoidance

Collision avoidance techniques have been extensively studied for robots avoiding obstacles (Long et al. 2018). According to Haddadin et al. (2017), collision avoidance occurs following seven steps, including the precollision phase, collision detection, collision isolation, collision identification, collision classification, collision reaction, and postcollision phase. Although most efforts have been made to accomplish simultaneous collision avoidance, for example, the use of mechanical responsive design to halt the machine immediately after a collision happens (Haddadin et al. 2008; Suita et al. 1995; Takakura et al. 1989), the precollision phase, or active collision avoidance, is when the collision-avoidance influence is the maximum and the cost is the minimum; hence, this is the focus of our method.

There are three main approaches for active collision avoidance: collision-free workspace design, collision-avoidance path planning, and anticipatory collision avoidance (Haddadin et al. 2017). The collision-free workspace is a set that contains all the positions that can be achieved by a reference point on the end-effector under the condition that all kinematic constraints are satisfied and no collision occurs (Yang et al. 2010). The possible mechanical collisions can be classified into two groups. One kind of collision is related to the manipulator's architecture and takes place between different constituting parts of the manipulator. The other kind happens when obstacles exist within the manipulator's workspace that may create barriers inside the workspace (Danaei et al. 2017). Many approaches have been developed in order to determine a collision-free workspace of manipulators (Bohigas et al. 2012; FarzanehKaloorazi et al. 2014; Snyman et al. 2000; Wang et al. 2010). FarzanehKaloorazi et al. (2014) investigated the collision-free workspace of a 3-RPR (where R represents a revolute, or hinged, joint, and P is a prismatic, or sliding, joint) planar parallel mechanism. The interval analysis is used for workspace determination of Gough-type (Merlet 1999) and six-degrees-of-freedom parallel mechanisms (Bonev and Ryu 2001). However, most of them are case dependent or computationally complex or exclusively include one of the aforementioned cases of collision (Danaei et al. 2017; FarzanehKaloorazi et al. 2017).

The collision-avoidance path planning method identifies a safe path that can avoid most possible collisions in a task based on workplace configuration, independently of human actions that would take place (Latombe 2012). Random tree exploration (Karaman and Frazzoli 2011) and A* algorithms (Hart et al. 1968) are often used to scan a large number of possible paths until one or more common "secure" paths are identified. In the view of collision-avoidance path planning, human actions are less relevant, but the spatial configuration–the locations of the robots and surrounding objects–is more important. In contrast, anticipatory collision avoidance focuses on predicting the human operator's actions and then exerting countermeasures (e.g., a reroute) based on the prediction (Bajcsy et al. 2019). Unlike the collision-avoidance path planning method, the anticipatory collision avoidance method

highlights the importance of incorporating predictions of human actions (Mainprice and Berenson 2013; Park et al. 2013). However, anticipatory collision-avoidance techniques are computationally expensive to convert to online methods that can handle instantaneous changes, especially when human action is taken into account (Mainprice and Berenson 2013; Sisbot and Alami 2012). To simplify, in this study, we focus only on robot–infrastructure collision and assume that human operators are very stable in operation instructions, with no unexpected actions. Although previous active collision avoidance methods, that is, collision-free workspace design, collision-avoidance path planning, and anticipatory collision avoidance, have good performance in some scenarios, these methods are usually performed in an isolated manner (Haddadin et al. 2017). In most representative studies, such as Danaei et al. (2017) and Bajcsy et al. (2019), the three methods mentioned previously were tested separately. The test was controlled only to manifest the performance and benefits of each method. This study provides solid evidence on each of the collision-avoidance approaches but does not generate any clear insight into the integration of these methods. Instead of renovating each of the collision-avoidance path planning algorithms, this study aimed to explore the integration between the two main collision planning methods, that is, collision-free workspace design and collision-avoidance path planning, and test the additional benefits that could be rooted in such an integration. This unique two-step, two-phase planning design of CWCP incorporates the uncertainties in both the design and operations phases of modular construction projects.

## Collision-Free Workspace and Path Planning Method

### *Overall Analytical Flow*

The proposed CWCP method executes a stepwise simulation to design (1) a collision-free workspace; and (2) a collision-avoidance path for robots in typical robot-assisted construction operations. Without losing its generalizability, we select industrial manipulators (robotic arms) for pipe skid operation as the study case. The focus of this paper was to present an innovative analytical flow for integrating collision-free workspace design and collision-avoidance path planning. For a better demonstration of the use of the method, a simpler planning problem, that is, a fixed robotic platform, was used in the simulation. It is our future agenda to incorporate scenarios of mobile robotic platforms and examine how they affect the proposed method. Fig. 1 illustrates the workflow of the CWCP method, consisting of two major steps: inverse kinematics Monte Carlo (IKMC) simulation and movement waypoint optimization.

As shown in Fig. 1, the collision-free workspace needs to try out a large number of possible robotic movement trajectories, with randomly assigned specification parameters, to identify the common boundaries of a safe work zone. The first step of CWCP was to determine the motion trajectories of the robotic arm based on the poses of the starting and target points. Then, cyclic coordinate descent inverse kinematics (CCDIK) with different assigned specification parameters was executed multiple times to draw a large number of probable robotic movement trajectories and the corresponding boundary of the robotic workspace. Eventually, the area containing a certain percentage of probable trajectories was selected as the work area. This result can be used to identify a robust structural design with enough clearance for most robotic actions.

The reason for using stochastic features and the simulation approach was mainly to address the uncertain work envelopes of future industrial manipulators (robotic arms), that is, the unique work clearance space profiles of each robot model due to the distinct locomotion specifications. At present, industrial manipulators are not widely available for testing in modular construction. In addition, it is too early to assume the values or probability density functions (PDFs) of important specifications about robot locomotion and kinematic features of yet-to-come robotic models, especially the work envelope of the robotic arm. For example, for typical industrial manipulators, the rotation degree of the end-effector joints could range

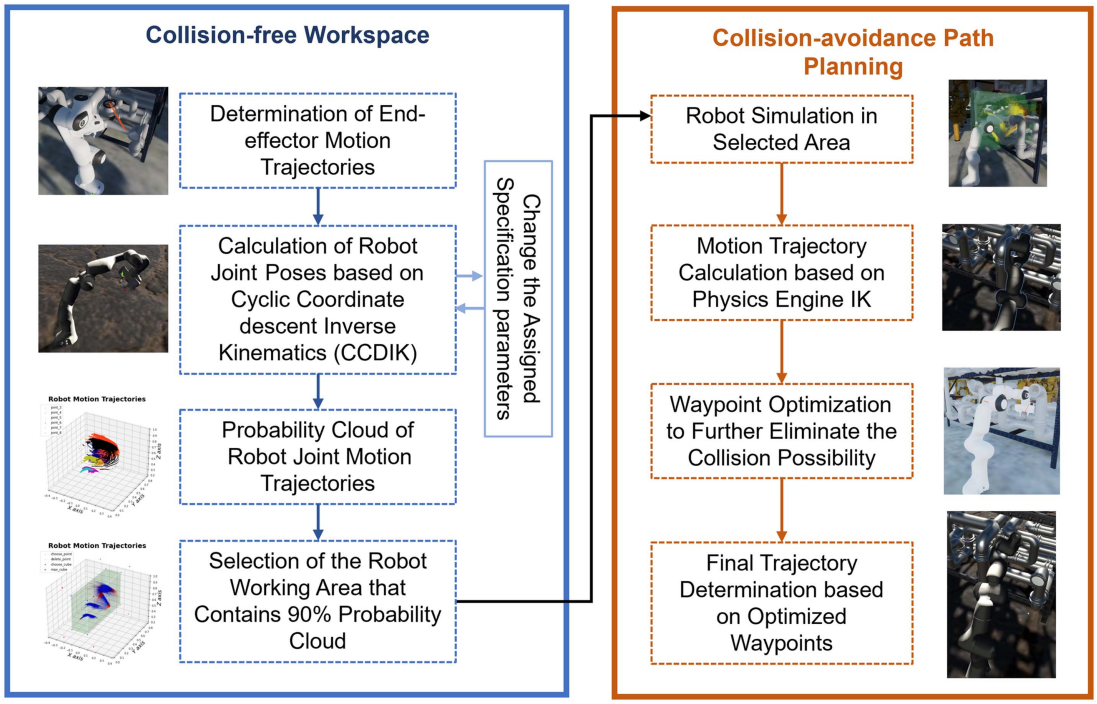



**Fig. 1.** Workflow of CWCP method.



J. Constr. Eng. Manage., 2022, 148(10): 04022114

from ±120° (KUKA LBR iiwa 14 R820) to ±400° (ABB IRB 120), whereas the base joint could range from ±165° (ABB IRB 120) to ±170° (KUKA LBR iiwa 14 R820). Different robotic arms will generate distinct work envelopes, causing different workspace clearance issues. With the rapid development of new industrial manipulators, we anticipate that there will be a deep uncertainty pertaining to the locomotion constraints of each of the joints of future robots. As a result, when planning for clearance in important modular construction processes, it is less reasonable to use a set of deterministic values for robot specifications. Because of the same consideration, we used uniform distributions as the PDFs. Previous literature, such as Quade and Carter (1989) and Walker et al. (2012), recommended using uniform distributions instead of any other predetermined assumptions of PDFs to address deep uncertainty simulation. This is the best available approach to avoid any bias in simulation parameter selection.

Before an operation, collision-avoidance path planning requires a high-fidelity reproduction of the robot specifications and established workspace to identify the best movement trajectory for robotic control. The physics engine IK was used to simulate and test the motion trajectory of the robot. Precollision prediction and active collision avoidance were executed based on the trajectory simulation. Then, CWCP optimized the waypoints of the trajectory to eliminate the collision possibility given the established structural design and robotic specifications.

### Collision-Free Workspace Design Based on Inverse Kinematics Monte Carlo Simulation

The first step of the proposed CWCP is the collision-free workspace design (CW simulation). The collision-free workspace design aims to identify a structural design, along with the workspace boundary design, to create sufficient clearance for all possible robotic systems to be deployed. CW is only about clearance planning without any ad hoc obstacles. In this study, CW was performed before collision-avoidance path planning to design enough clearance to work for most scenarios, whereas once the robot is placed on site, collision-avoidance path planning is performed to further reduce the chance of collisions given the specific job site conditions. CW simulation relies on an inverse kinematics–based Monte Carlo simulation. To control a robot arm, IK, that is, the use of kinematic equations to determine the joint parameters of a manipulator (Aristidou et al. 2018), is used to move the end-effector to the desired position. A variety of IK methods have been developed, such as the analytical method (Craig 2009), numerical method (Buss 2004), and artificial neural networks (El-Sherbiny et al. 2018). The mathematic efficiency and reliability of IK have been well validated in robotics (Sciavicco and Siciliano 2012) and computer science literature (Aristidou et al. 2018). Among existing IK algorithms, CCDIK (Luenberger and Ye 1984; Wang and Chen 1991) is an iterative heuristic technique and is suitable for interactive control of an articulated body. The viability for creating and controlling highly articulated characters of CCDIK was examined by Kenwright (2012). CCDIK provides a numerically stable solution, and it has linear-time complexity in the number of degrees of freedom, which leads to low computational cost per iteration (Welman 1993). Compared to other methods, CCDIK is flexible in that it allows constraints to be placed at each step and is easy to program, conceptually simple, and computationally fast (Canutescu and Dunbrack 2003), showing better performance in DT simulation. In this research, the robot locomotion analysis was based on CCDIK.

We defined the starting and target positions of the end-effector of a robotic arm as $(X_0, Y_0, Z_0)$ and $(X_t, Y_t, Z_t)$. Then, the poses (positions and rotations) of the joints of the robotic arm can be recovered based on the CCDIK. In order to solve the IK problem, the joint angles must be set so that the resulting configuration moves the end-effectors as close as possible to the target position (Aristidou et al. 2018). Let $\mathbf{s_1}, \ldots, \mathbf{s_k}$ be the positions of the end-effectors, where $k$ is the number of end-effectors. Let $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ be the scalars that describe the complete joint configuration of the multibody, where $n$ is the number of joints. Each $\boldsymbol{\theta}_j$ value is the joint angle, and each $s_i$ can be expressed as a function of the joint angles. The target positions are defined as $\mathbf{t_1}, \ldots, \mathbf{t_k}$, where $\mathbf{t_i}$ is the target position for the $i$th end-effector. The desired change in position of the $i$th end-effector is given by $\mathbf{e_i} = \mathbf{t_i} - \mathbf{s_i}$. The vector of these parameters can be formed as $\boldsymbol{s} = (\mathbf{s_1}, \ldots, \mathbf{s_k})^{\mathbf{T}}$, $\mathbf{t} = (\mathbf{t_1}, \ldots, \mathbf{t_k})^{\mathbf{T}}$, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k)^{\mathbf{T}}$, $\mathbf{e} = \mathbf{t} - \mathbf{s}$. The end-effector positions can be expressed as functions of the joint angles

$$\mathbf{s} = f(\boldsymbol{\theta}) \tag{1}$$

On the other hand, the goal of IK is to find a vector $\boldsymbol{\theta}$ such that $\mathbf{s}$ is equal to a given desired configuration $\mathbf{s_d}$

$$\boldsymbol{\theta} = f^{-1}(\mathbf{s_d}) \tag{2}$$

The CCDIK method attempts to minimize the pose errors by transforming one joint variable at a time (Aristidou et al. 2018). The main idea of CCDIK is to align each joint position with the end-effector and the target at each step, starting from the end-effector and moving inward toward the manipulator base, and each joint angle is transformed so that the last one of the chain gets closer to the target. Assume a kinematic chain consists of $n$ joints, where $\mathbf{j_1}$ is the root joint, $\mathbf{j_n}$ is the end-effector, and $\mathbf{T}$ is the target position. First, CCDIK finds the angle $\boldsymbol{\theta}_{n-1}$ defined by $\mathbf{T}$, $\mathbf{j}_{n-1}$, and $\mathbf{j_n}$. Then, update the end-effector's position by rotation $\mathbf{j_n}$ so that $\boldsymbol{\theta}_{n-1}$ is set to 0. Similarly, find the angle $\boldsymbol{\theta}_{n-2}$ defined by $\mathbf{T}$, $\mathbf{j}_{n-2}$, and $\mathbf{j_n}$, and update the $\mathbf{j}_{n-2}$ and $\mathbf{j_n}$ positions so that $\boldsymbol{\theta}_{n-2}$ is set to 0 (Wang and Chen 1991). An iteration is completed when all joints are updated. This procedure is repeated until the end-effector is satisfactorily close to the target position.

Without losing its generality, we consider it the most popular industrial manipulator with seven degrees of freedom. There exist several ways to improve the performance and increase the realism of the animation; one of these is to incorporate constraints (Aristidou and Lasenby 2009). The simplest way of incorporating constraints can be achieved by weighting the moves of the individual joints (Meredith and Maddock 2005). In CW simulation, we focused on the most influential factor in IK, that is, the weight of the joints. It shows the relative level of flexibility of the motion freedom of a particular joint in comparison with other joints. It was also used to fine-tune the recovered joint locations or rotations for a given application (Meredith and Maddock 2005). In CCDIK, the weight refers to the priority in the locomotion controls of the industrial manipulator. When weighting the moves of the individual joints in CCDIK, it indicates the percentage of the angles $\boldsymbol{\theta}$ each joint can move. Let $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k)^{\mathbf{T}}$ be the angle change for each joint every iteration; $\mathbf{W} = (\mathbf{w_1}, \ldots, \mathbf{w_k})$ be the weight of each joint; and $\mathbf{W\theta} = (\mathbf{w_1}\boldsymbol{\theta}_1, \ldots, \mathbf{w_k}\boldsymbol{\theta}_k)$ be the update angle change for each iteration. For example, if Joint 2 has a weight of 0.5, Joint 2 can move $0.5 \times \boldsymbol{\theta}_2$ for each iteration. In this way, a weight of 1 means that the movement of the joint has not been adjusted, and a weight of 0 means that the joint cannot move.

Fig. 2 shows an example of different received joint postures with different priority values for the same target point. In other words, it shows snapshots of the continuous motion of IK-driven robotic arm motions. The group weights $\mathbf{W} = (\mathbf{w_1}, \ldots, \mathbf{w_k})$ for each robot are (1,0,1,1,1,1,1,1) for Fig. 2(b), (1,1,1,1,1,1,1,1) for Fig. 2(c), and

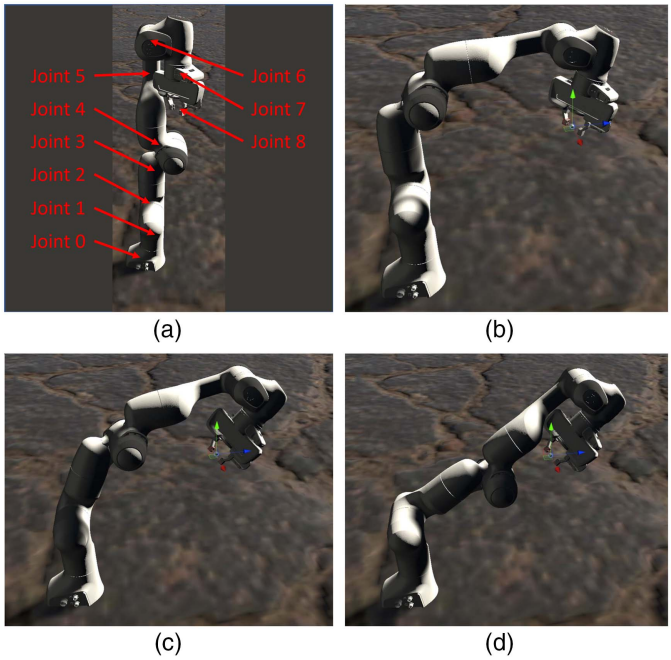



**Fig. 2.** Different joint postures for the same target point based on three different weights.

(1,1,1,0,1,1,1,1) for Fig. 2(d). The robot's original pose and order of each joint are illustrated in Fig. 2(a). As illustrated in Fig. 2(b), the weight of Joint 2 is 0, so Joint 2 cannot be moved in CCDIK, and it remains in the same pose as the original one. The same is true for Fig. 2(d). Joint 4 cannot be moved and stays in the original pose. If all the weights are 1.0, the priority of each joint is the same, and every joint can be fully moved, as shown in Fig. 2(c).

The scope of this study is to propose a method to capture the uncertainties and unknown specifications of future modular construction robots instead of selecting from any known robot. With that said, the goal is to design a facility that is tolerant of any possible robot specifications. As such, the key is to perform an uncertainty analysis, such as a Monte Carlo simulation, to estimate the chance of collision given a possible range of robotic locomotion features. Given the changing priority, we define a Monte Carlo simulation as follows:

In order to find a collision-free workspace, the maximum area that the robot can reach needs to be determined first, and then part of the area is selected as the working area according to the minimum allowable requirements of locomotion errors of the robotic systems. This is because no matter how well the robotic locomotion controls are designed, there must be uncontrolled errors in such controls. According to the most popular robotic manufacturers, the pose accuracy and repeatability of the robotic arm are $\pm 0.5$ mm based on ISO 9283 (ISO 1998) and ANSI/RIA R15.06 (Institute 2012). Therefore, this was incorporated into our simulation. The job of the robot in our research was to move from the starting point to the designated valve and turn the valve. All the areas that the robot joints can reach during the movement are the maximum boundaries. To find the possible robotic movement trajectories and boundaries, we randomly selected the weight for each joint and executed it multiple times. Different weights for each joint meant that the joint's position was different in an IK calculation. It also meant that different weights produced different trajectories. The continuous uniform distribution was used for the weight selection of each joint. The weights and trajectories of all joints were recorded to establish boundaries based on the Monte Carlo simulation.

The reason for using uniform distribution is that the study aimed to address future robot-assisted modular construction works, where most of the parameters of the future robot are subject to changes or not yet known. For the so-called "deep uncertainty" problems, the literature has suggested the use of uniform distribution to avoid any biased assumptions on the PDFs. As Quade and Carter (1989) wrote: "Stochastic uncertainties are therefore among the least of our worries; their effects are swamped by uncertainties about the state of the world and human factors for which we know absolutely nothing about probability distributions and little more about the possible outcomes." These kinds of uncertainties are now referred to as deep uncertainty (Lempert 2003) or severe uncertainty (Ben-Haim 2006; Walker et al. 2012). For example, Quade and Carter (1989) suggested that when there was an absence of data about the prior probability distributions for the critical parameters, a uniform distribution should be assumed.

Finally, the boundary of a safe, collision-free workspace $\mathbf{R}$ can be defined as

$$\mathbf{R}' \in \mathbf{R} \tag{3}$$

In other words, the structural design should allow the minimal clearance that covers the simulated boundaries of all seven joints and connections of the robotic system from the CW Monte Carlo simulation.

### *Collision-Avoidance Path Planning Based on Physics Engine Simulation*

The second step of the CWCP method is to plan out a collision-avoidance robotic movement path via the physics engine simulation. We consider the work sequences in actual construction operations, where out-of-sequence work often occurs, which can hardly be predicted or planned. As a result, our goal is to capture the vast variability and variation of work sequence orders in real operations. Thus, the most robust decision should be built upon all possible sequence orders. In our case, we consider it the situation in which all possible obstacles in the middle of the planned path are present. Collision-avoidance path planning aims to optimize the robotic movement for a given workspace and a given robotic platform. In this study, we demonstrate how physics engine simulation can be used to identify one or more paths for robot–infrastructure collision avoidance. We define the starting and target points as $(X_0, Y_0, Z_0, \phi_0, \theta_0, \psi_0)$ and $(X_t, Y_t, Z_t, \phi_t, \theta_t, \psi_t)$. Then the path of the robotic arm and all seven joints are based on the given parameters of IK as follows.

The Jacobian method is used to calculate the IK solution for CP. Differentiation of Eq. (1) gives the forward dynamics equation (Aristidou et al. 2018; Buss 2004)

$$\dot{\mathbf{s}} = \mathbf{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} \tag{4}$$

The Jacobian matrix $\mathbf{J}$ can be described as a function of the $\boldsymbol{\theta}$ values and is given by

$$\mathbf{J}(\boldsymbol{\theta})_{\mathbf{q}} = \left(\frac{\partial \mathbf{s_i}}{\partial \boldsymbol{\theta}_\mathbf{j}}\right) \tag{5}$$

where $i = 1, \ldots, k$; and $j = 1, \ldots, n$ (where $k$ is the number of end-effectors, and $n$ is the number of joints). Thus, $\mathbf{J}$ would be a $k \times n$ matrix with vector entries. In practice, this would be converted to a $3k \times n$ matrix of scalar entries. Then, we calculate the entries of $\mathbf{J}$ using quantities $v_j$, which are the unit vectors pointing along the rotation axis of the $j$th joint

$$\frac{\partial \mathbf{s_i}}{\partial \boldsymbol{\theta}_\mathbf{j}} = \mathbf{v_j} \times (\mathbf{s_i} - \mathbf{p_j}) \tag{6}$$

where $\mathbf{p_j}$ = position of the joint.

Suppose the target position for end-effector $i$ is $\mathbf{t_i}$; then, attempt to find the values $\boldsymbol{\theta}$, which minimize the errors $\mathbf{e_i}$ between the actual end-effector and target positions

$$\mathbf{e_i} = \mathbf{t_i} - \mathbf{s_i}(\boldsymbol{\theta}) \tag{7}$$

A small change $\Delta\boldsymbol{\theta}$ is made in the joint angles, and the consequent change in end-effector positions is approximated as

$$\Delta\mathbf{s} \approx \mathbf{J}\Delta\boldsymbol{\theta} \tag{8}$$

$\mathbf{J}$ can be calculated from the current values of $\mathbf{s}$ and $\boldsymbol{\theta}$. Because we are looking for a value of $\Delta\mathbf{s}$ that is as close as possible to the error $\mathbf{e}$ (the error term $e$ should be clamped to avoid instabilities in convergence), we can estimate the change in $\boldsymbol{\theta}$ **to be** $\Delta\boldsymbol{\theta} \approx \mathbf{J}^{-1}\mathbf{e}$.

After calculating the total change $\Delta\boldsymbol{\theta}$ of each joint, we also divide $\Delta\boldsymbol{\theta}$ into certain segments $\Delta\boldsymbol{\theta}_1, \ldots, \Delta\boldsymbol{\theta}_n$, where $n$ is the number of divisions. The pose of each joint is calculated for each $\Delta\boldsymbol{\theta}_k$, and the trajectory is built based on the position of each joint.

### CW-CP Weighting

We also recognize that the constraints for CW and CP simulations may change in practice. For CW, although CW simulation gives a clearance space of $R$, it may be beyond the allowable space from a structural design or economic perspective. For example, the spanning of a pipe rack structure usually has an upper limit for structural integrity purposes. In this case, the identified collision-free workspace may not be 100% achievable. Similarly, the CP optimization of the path may fail to identify a path that is completely collision free. As a result, the results of both CW and CP should be adjusted according to how much flexibility can be granted to the design changes and operational changes. We define an allowance factor $\alpha$ to give an adjusted CW result as

$$R_{\text{adj}} = R_\alpha \tag{9}$$

In other words, a realistic workspace design may not be perfectly aligned with the CW simulation results, still causing a certain chance of collision in rare cases. It is the cost of satisfying other factors such as structural design requirements. Similarly, there may be cases where CP cannot find a solution for a collision-avoidance path. As a result, we adopt a midwaypoint approach by adding one or more transition waypoints $(X_t', Y_t', Z_t')$ that add deviations to the original target point

$$(X_t', Y_t', Z_t') = (X_t, Y_t, Z_t) \pm (\Delta x, \Delta y, \Delta z) \tag{10}$$

The determination of deviation levels $(\Delta x, \Delta y, \Delta z)$ can be found with reinforced learning, such as A* search (Hart et al. 1968) and random tree exploration (Karaman and Frazzoli 2011). A* starts from a specific starting node and aims to find a path to the given goal node with the smallest cost. At each iteration of its main loop, A* needs to determine which of its paths to extend. It does so based on the cost of the path and an estimate of the cost required to extend the path all the way to the goal (Hart et al. 1968)

$$f(n) = g(n) + h(n) \tag{11}$$

where $n$ is the next node on the path; $g(n)$ is the cost of the path from the start node to $n$; and $h(n)$ is a heuristic function that estimates the cost of the cheapest path from n to the goal. A* terminates when the path it chooses to extend is a path from the start to goal or if there are no paths eligible to be extended.

It is natural that the selection of $\alpha$ will affect the levels of $(\Delta x, \Delta y, \Delta z)$. Specifically, given a greater value of $\alpha$, the deviation of $(\Delta x, \Delta y, \Delta z)$ should be greater. Therefore, we assume a linear relationship $f(\cdot)$ between the two

$$(\Delta i) = f(\alpha) \ldots \text{Eq}(x) \tag{12}$$

$\text{Eq}(x)$ will then be used as an important input for the reinforced learning algorithm.

## Test Case Study

### Overview

We used an industrial robot-assisted pipe skid modular construction as the test case for the proposed CWCP method. The model included a pipe skid system and a Franka Panda robot in the virtual reality (VR) system. The robot was required to install the valves and pipelines for the pipe skid. The interactive VR system was developed based on our previous systems (Du et al. 2016, 2017, 2018a, b; Shi et al. 2018; Zhou et al. 2020; Zhu et al. 2021). Fig. 3 shows the pipe skid system and the Franka Panda robot in the virtual environment. The system collected data on robot joints' trajectories at a frequency of 90 Hz. After each VR experiment, the developed VR system automatically recorded the raw data and saved it into a CSV file.

The main purpose of the VR system in this research was to collect data to test the proposed CWCP models. The VR environment was built in a way to trigger similar robotic responses as seen in real work settings due to the same IK algorithms used and was thus reasonable for the model development. We also recognize and agree that the model, once tested, will require real-world data collection techniques to be implemented in real-world settings.

The modular structure was relatively deterministic in our test case, but the interaction between the robot and the structure was something that needed to be optimized. We relied on two factors to drive the simulation-based optimization: the simulated robot work envelope (as discussed earlier) and the clearance spacing of the modular structure that had direct contact with the robot. In our test case, the robot arm was used to install a valve during assembly. To reach the valve, the robot arm needed to go through a confined space between Points A and B, as shown in Fig. 4. The goal was to minimize the spacing between Points A and B for cost and performance considerations while still allowing a safe (i.e., collision-free) work interface between the robot arm and structure. To provide more details about the structure and robot used in the optimization simulation, we show dimension information in Table 1 and Fig. 5. Fig. 5(a) shows the kinematic chain and parameters of the simulated robot. Figs. 5(b and c) illustrate the side and top views of the robot's work envelope. This will help other scholars reproduce the findings of this study. In addition, we have shared the VR models of both the structure and the robot in a GitHub project (Link: https://osf.io/jcz2y/). Interested scholars are welcome to rebuild the scene with the shared models.

Fig. 3 shows the relative spatial arrangement between the robot and the facility.

### CW Simulation for Collision-Free Workspace Design

We selected the path between the starting point and target valve as the design factor. IK-based Monte Carlo simulation was used to find the collision-free workspace. A total of nine joint positions


**Fig. 3.** The pipe skid system and Franka Panda robot in the virtual environment.

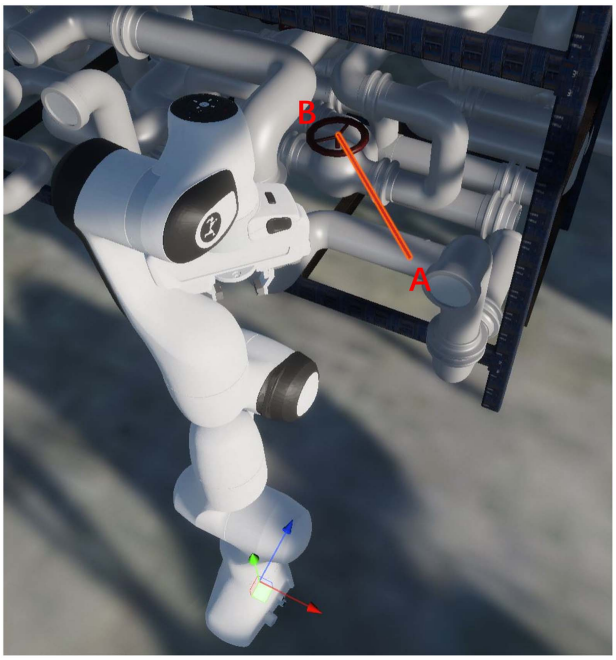



**Fig. 4.** The task path of the robot in the pipeline system.

**Table 1.** Specifications of robot used in the optimization simulation

| Joint | $a$ (m) | $d$ (m) | $\alpha$ (rad) | $\theta_{\min}$ (rad) | $\theta_{\max}$ (rad) |
|---|---|---|---|---|---|
| Joint 1 | 0 | 0.333 | 0 | −2.8973 | 2.8973 |
| Joint 2 | 0 | 0 | $-\frac{\pi}{2}$ | −1.7628 | 1.7628 |
| Joint 3 | 0 | 0.316 | $\frac{\pi}{2}$ | −2.8973 | 2.8973 |
| Joint 4 | 0.0825 | 0 | $\frac{\pi}{2}$ | −3.0718 | −0.0698 |
| Joint 5 | −0.0825 | 0.384 | $-\frac{\pi}{2}$ | −2.8973 | 2.8973 |
| Joint 6 | 0 | 0 | $\frac{\pi}{2}$ | −0.0175 | 3.7525 |
| Joint 7 | 0.088 | 0 | $\frac{\pi}{2}$ | −2.8973 | 2.8973 |
| Flange | 0 | 0.107 | 0 | 0 | 0 |

were recorded, where Joint 0 was the base, Joints 1–7 were seven joints of the robot, and Joint 8 was the grabber mounted on the Joint 7 (end-effector). Because the robot base was not moveable, we did not need to set the weight for Joint 0, and the position for this joint (JointPos0) only needed to be recorded once.

Considering that the dimension, weight, and payload of the end-effector may also affect the collision-free workspace, an additional boundary is added to the target position of the end-effector. The payload and end-effector dimension boundary were determined to cover the maximum range of the majority of the possible robotic arms. We relied on the specifications of the popular small-scale industrial manipulators, such as the Franka Panda robot, for estimating the boundary. The payload was defined as 3 kg, and the dimension of the end-effector was set to $200 \times 50 \times 75$ mm $(L \times W \times H)$. The specification of the valves was set to $100 \times 10$ mm (Radius × Height) based on the possible specifications of valves. The weight of the Franka Panda hand was 0.73 kg. The payload of the valve was set to 2 kg to avoid unexpected movement of the robot. These values were large enough to cover the uncertainties of the majority of the robotic systems used. In order to consider the space required by the robot in the process of turning the valve, we added a boundary to the robot's end-effector based on the size of the robot's hand, valve, and pipes. The boundary we set was ±200 mm for the $x$-direction according to the length $\frac{L}{2} + R$, ±125 mm for the $z$-direction according to the width $\frac{W}{2} + R$, and +85 mm for the $y$-direction according to the height $H_{\text{Hand}} + H_{\text{Valve}}$. Then, each joint's randomly selected weight from a continuous uniform distribution $x \in [0, 1]$, and 1,000 sets of different weights corresponding to each group of joints (Joints 1–8) were selected to calculate the robot motion boundary. The weight of each joint $(W1, W2, W3, W4, W5, W6, W7, W8)$ and the position of each joint (JointPos1, JointPos2, JointPos3, JointPos4, JointPos5, JointPos6, JointPos7, JointPos8) were collected. The raw data can be found at: https://osf.io/jcz2y/.

The trajectory of each joint was then calculated based on CCDIK. Because CCDIK only calculated the position of each joint and did not include path planning, we manually created the path in

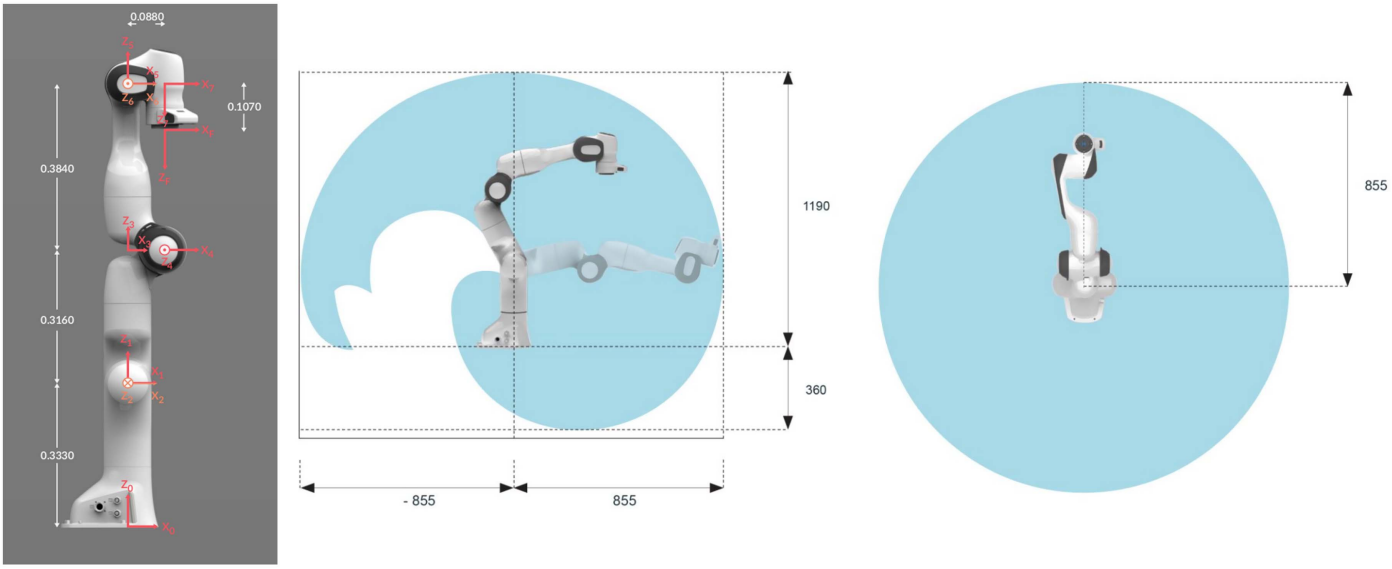



**Fig. 5.** Robot specifications: (a) Panda's kinematic chain; (b) arm workspace side view; and (c) arm workspace top view.

the starting point and target valve. We set the path between the two points to be a straight line and divided this straight line evenly into 100 points. CCDIK was used to calculate the position of all joints corresponding to each point on the trajectory to calculate an approximate path. The weight of each group $(W1–W8)$ was constant when calculating a trajectory and changed to another group of random numbers when one trial was completed. The data collection video can be found at: https://osf.io/jcz2y/.

Fig. 6 shows 1, 10, 100, and 1,000 trials of all the robot joint trajectories. As illustrated in Fig. 6(a), the robot arm model and one of the robot motion trials are shown in three-dimensional (3D) space. The trajectories of each joint of the robot are shown in a different color. At the same time, the joint corresponding to each trajectory is also marked. Figs. 6(b–d) show the results of 10, 100, and 1,000 trials, respectively. The bottom three points in these three pictures are the trajectories of Joints 0, 1, and 2. The trajectories of these three joints converge to a point because the IK calculation suggests that the movements of these three joints are neglectable. Because the positions of Joints 0, 1, and 2 were almost unchanged, we only recorded their positions but did not apply them to the data analysis of the Monte Carlo simulation. The rest of the joints (Joints 3–8) were used to calculate the boundary in the Monte Carlo simulation, as shown in Fig. 7(a). We defined a cuboid area as the robot motion boundary. The positions $(X, Y, Z)$ of the eight vertices of the cuboid were determined by the maximum and minimum $X$, $Y$, and $Z$ positions of the trajectories. Fig. 7(b) shows the cuboid area of the maximum robot motion workspace. However, the allowable space may be beyond the robot motion space from a structural design or economic perspective. In this case, we identified the collision-free workspace, which includes 90% of the trajectories as an example. As illustrated in Fig. 8, the translucent cuboid area shows a collision-free workspace; the trajectories outside the workspace are considered collisions, and those inside the workspace are collision-free. After identifying the collision-free workspace, the pipe skid system was designed based on the workspace. Fig. 9 shows the collision-free workspace in the pipe skid system; the cuboid area is between two pipes and two railings. Figs. 9(a–c) show the collision-free workspace without a robot arm. The lines are the moving trajectories of the robot's joints. The cuboid area is the workspace, the trajectories inside the workspace are collision-free, and the trajectories outside the workspace are considered collisions. We also added a robot arm with an example pose to make the image clearer. As illustrated in Fig. 9(d–f), Joints 3–8 of the robot arm were working in the collision-free workspace.

Fig. 6 shows the example results from the IK Monte Carlo simulation. The curves represent the motion trajectories of all joints. The density of curves shows the cumulative results from the Monte Carlo simulation.

Fig. 7 shows the example analysis between the allowable space and work zone envelope of the robotic arm. The curves represent the trajectories of the joints of the robotic arm, whereas the cube indicates the allowable space defined by the facility design. As such, all curves beyond the boundary of the cube are considered collisions.

Fig. 8 shows the example analysis between the allowable space and the work zone envelope of the robotic arm. The curves represent the trajectories of the joints of the robotic arm, whereas the cube indicates the allowable space defined by the facility design. As such, all curves beyond the boundary of the cube are considered collisions.

Fig. 9 illustrates how the collision analysis looks in the context of the facility. All curves beyond the boundary of the cube are considered collisions.

### *CP Optimization for Collision-Avoidance Path Planning*

We then controlled the robotic arm to move from another test point to the target valve. The Jacobian method was used to control the robotic arm for CP. We divided $\Delta\boldsymbol{\theta}$ into 100 even segments $(\Delta\boldsymbol{\theta}_{\mathbf{k}})$ and recorded the position of all joints when the robot rotated each segment angle $\Delta\boldsymbol{\theta}_{\mathbf{k}}$. The trajectories of each joint were calculated based on the position recorded.

The test point, valve target, and trajectories were in the collision-free workspace determined in CW. However, the CP optimization failed to identify a path that was completely collision free due to the spatial limitation. In other words, the realistic workspace design was not aligned with the CW simulation results, still causing a

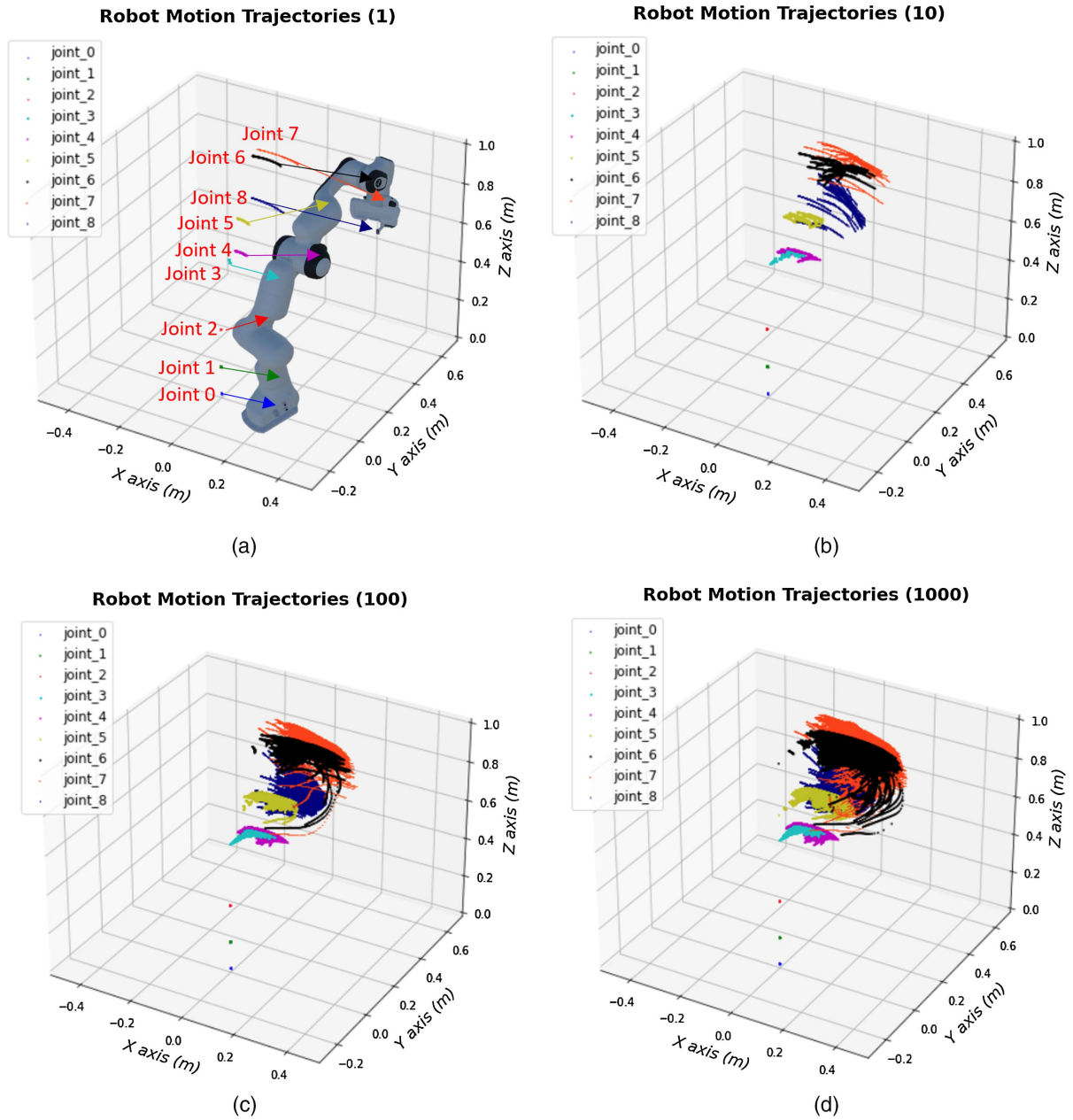



**Fig. 6.** The schematic diagram of the number of trajectories of robot joints: (a) 1; (b) 10; (c) 100; and (d) 1,000.

collision. As illustrated in Fig. 10, we set a movement from a point in the collision-free workspace to the target valve. During the movement, the collision happened between robot Joint 6 and the left pipe. As a result, the midwaypoint approach was achieved by adding one transition waypoint $(X_t', Y_t', Z_t')$. We used the A* search algorithm to scan a large number of possible paths until one or more collision-avoidance paths were identified. The workspace of the robot was defined as a cuboid with $L$: [0.25,1.08], $W$: $[-0.21, 0.17]$, and $H$: [0.00, 0.80]. We set a point for every 0.01 scale in this cuboid area, and a total of $83 \times 38 \times 80 = 252{,}320$ points were set. The A* search algorithm was used to find the paths based on these points. The average computation time for path planning was about 4.5 s. The path with the shortest length and the fewest midwaypoints was selected as the final path.

As illustrated in Fig. 11, a midwaypoint was found based on the A* search. The robot moved from the starting point to the midwaypoint and then moved to the target valve. The lines represent the trajectory of each joint. Fig. 11(a) shows which joint each trajectory corresponds to. The midwaypoint is the turning point in the trajectory of Joint 8. Because the positions of Joints 0, 1, and 2 were almost unchanged, we did not draw the trajectories of them. Fig. 11(b) shows the pose of the robot passing the midwaypoint. Each joint is passing its corresponding trajectory. The trajectories of the robot joints confirm that the midwaypoint avoided the collisions. The pose of each joint at the end of the trajectory is shown in Fig. 12. The joint avoided collision with the pipe. The midwaypoint video can be found at: https://osf.io/jcz2y/.

Fig. 11 shows the robotic arm trajectories in one specific motion.

## Discussion and Conclusions

Robots are being deployed in a variety of industrial applications to promote safety and productivity in dynamic workplaces.

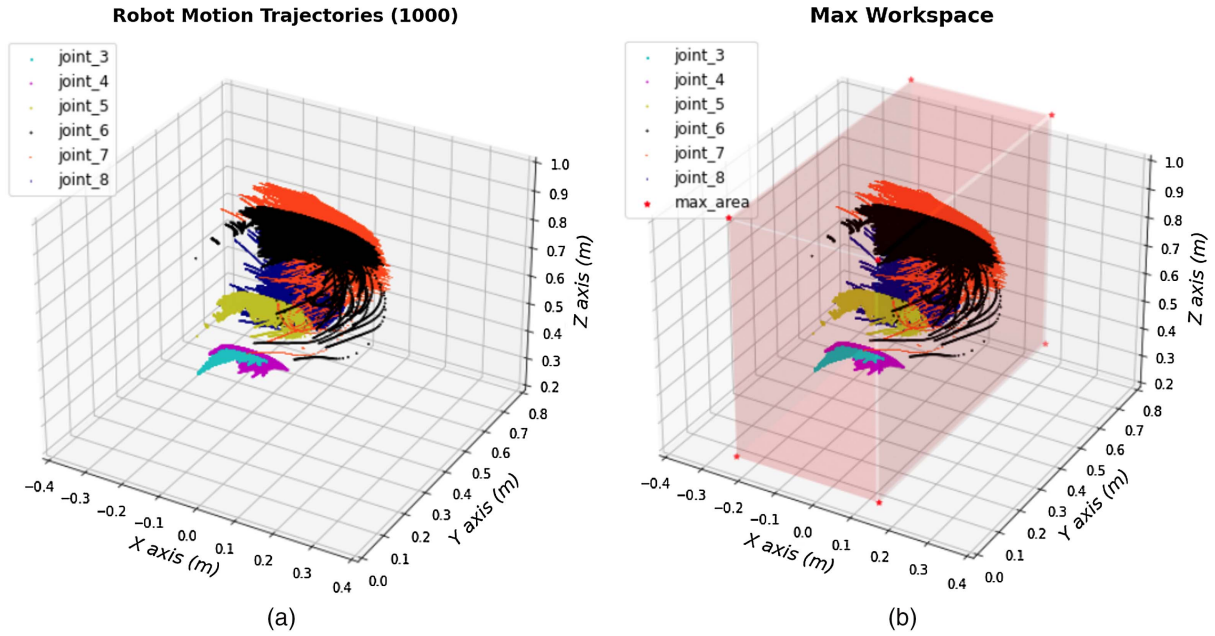



**Fig. 7.** The trajectories of (a) Joints 3–8; and (b) the maximum working space of the robot.

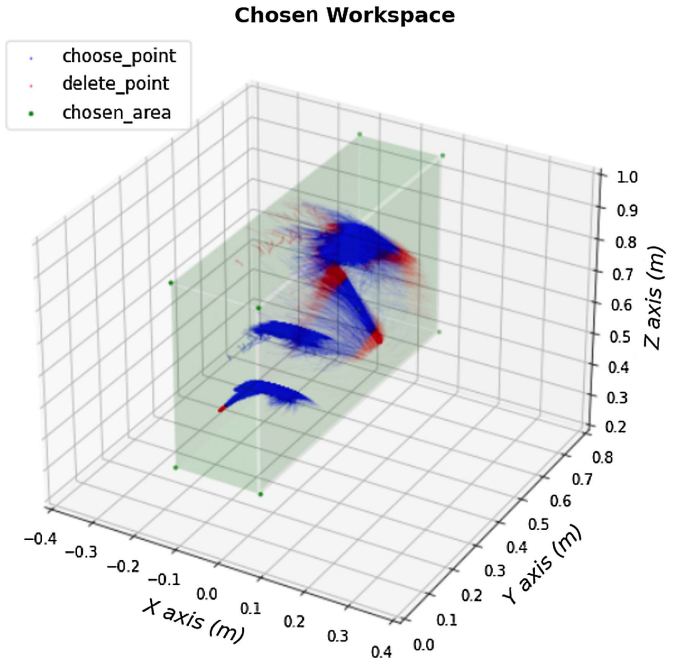



**Fig. 8.** The collision-free workspace with chosen and deleted trajectories.

For modular construction, robots show unique benefits because they allow construction workers to engage in tasks that are traditionally unsafe, inaccessible, or difficult. Successful applications of robots in the construction industry include the use of remote industrial manipulators in precision operations, such as those seen in facility installation and maintenance. Despite the apparent benefits of robots, concerns still persist. In particular, robotic collisions with human workers, with infrastructure or equipment, and with other robots on construction sites are highly possible because construction workplaces are usually dynamic and constrained. Moving workers and construction objects complicate any effort that aims to reduce collisions with proactive or reactive strategies. In particular, passive collision avoidance methods that focus on taking reactive actions when a collision occurs are not as effective in construction operations as in well-controlled workplaces because there are many influential factors on a construction job site (e.g., unexpected motions of a nearby worker) that could impair reactive actions. An active robotic collision avoidance method is needed to facilitate the design of a collision-free construction workspace with robots for the most likely cases of robotic operations and/or to plan robotic operational procedures when the site conditions are known.

This paper proposed an active robot collision avoidance method for collaborative robotic operations based on the collision-free workspace and collision-avoidance path planning. It is important to validate a new technology with a real-world case when it is available. However, the scope of this work aimed to address yet-to-come new technology: robot-assisted modular construction scenarios. At this point, it is not yet developed and deployed in real-world applications. Not only is the structure to be built unknown, but the robots that will become available are also less clear. In addition, the knowledge gap we wanted to fill concerns deep uncertainty about the future use of robots. In other words, we do not yet have a clear understanding and data about the specifications of the future robot for modular construction. However, it is important for us to understand the methods and analytical workflow to address such uncertainty when robots become available in the future. With that said, the main contribution of this work is to present a simulation-based method for analyzing, calculating, and controlling the uncertainties of a vast selection of robots that will be used for future modular construction. If we select one kind of robot to run a real-world validation, it will invalidate the point of testing and stimulating uncertainties from a much bigger number of possible robots. A validation case with a real robot, of course, will become available when the industry agrees upon the specific type of robot to be used for a specific modular construction operation. As a result, our analysis focused on providing a fuzzy but robust workspace and path to avoid possible collisions in most cases with a


**Fig. 9.** The collision-free workspace in the pipe skid system.

given confidence level. Therefore, the only measure in this case was the achieved confidence level of the planned workspace and path in avoiding future collisions, given the probability distribution of the robotic parameters. In our case study, we selected a 90% safety range as the performance measure, which means that by using the recommended clearance and path spatial design, 90% of future collisions can be avoided.

A real-world application can be inspired based on the CWCP method. First of all, the future design of modular structural components should incorporate the possibility of the application of robotic systems. Thus, the locomotion and spatial requirements of the robotic system should be considered in the design phase. The digital twin simulation informed by this study can be an effective design decision tool for future structural engineers to consider the parameters of a more complex modular structure and its corresponding spatial coordinates with the presence of robotic systems. Another real-world application that can be inspired by this study is a virtual reality-robot operating system (VR-ROS) platform that drives a real robot for validation in a testbed. There may be special situations in the operation of real robots. The lessons learned from VR-ROS simulations can help us develop better methods to improve the effectiveness of future robotic controls. The challenge is the PDFs used for the Monte Carlo simulation because the parameters of future robots can be difficult to capture or predict given the rapid development of the tech trend. As a result, we used the uniform distribution to avoid any possible bias in our simulation, but this should definitely be improved. In addition, we used a fixed robotic platform in the simulation. Incorporating scenarios of mobile robotic platforms in the future may increase the performance of the digital twin simulation.

The proposed CWCP method contributes to safer modular construction robot applications in two ways. First, it presents a collision avoidance mechanism tailored for modular construction. Most existing robotic collision avoidance methods are designed for well-controlled works and workplaces, such as manufacturing. Because the operations and environments are more predictable, anticipatory collision reactions are possible based on the prediction of the system status and human behaviors. In contrast, construction workplaces (even for modular construction) are less controlled, posing uncertainties to any efforts in prediction-based countermeasures. As a result, CWCP adopts robust design principles at the beginning of the project. Through a simulation of various design solutions, it is possible to identify a structural and workplace design that meets the need of both work safety and structural considerations. Although work zone design methods can also be seen in several studies, modular construction projects can also present deviations from master plans, such as out-of-sequence activities. As a result, CWCP uses onsite path planning simulation as the secondary

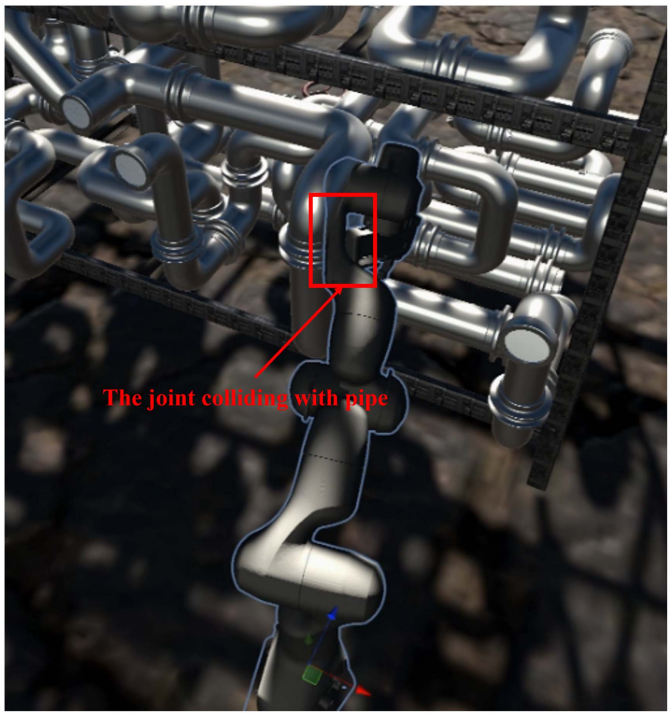
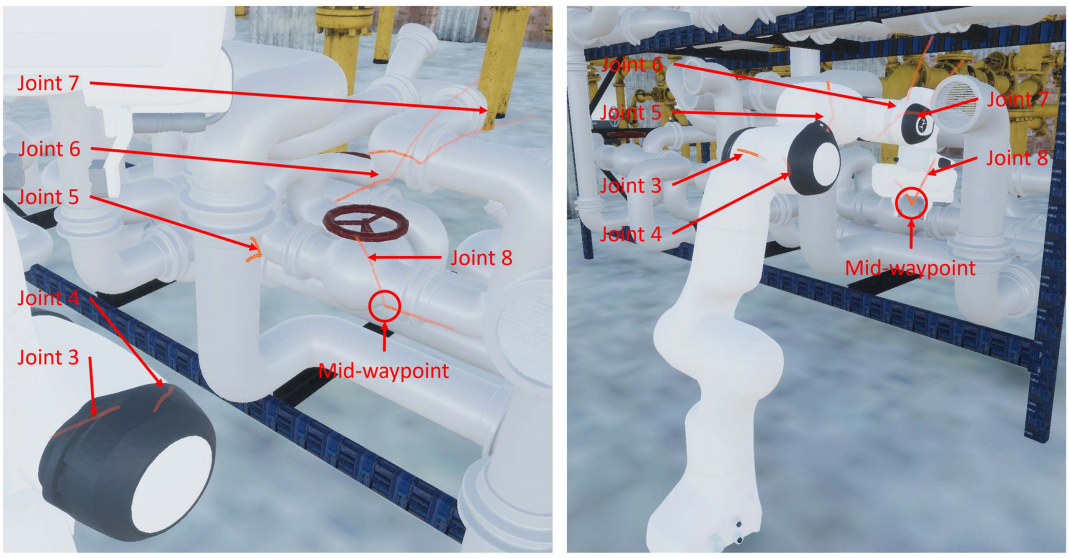



**Fig. 10.** A collision between the joint and pipe in the collision-free workspace.

and last safeguard to ensure minimized collision likelihood. This unique two-step, two-phase planning design of CWCP is the result of the uncertainties in construction projects. Second, CWCP demonstrates the potential of digital twins simulation in robot planning for future modular construction operations. Both CW and CP steps rely on the precise reproduction of the workspace and work procedures of the prospective modular construction tasks. Robotic movement simulation also builds on real mechanical algorithms (i.e., IK and physics engine) to mimic robotic motions and interactions with the surrounding environment. Our case study shows that both DT simulations in CW and CP steps can be accomplished with reasonable resources (such as the Unified Robotics Description Format [URDF] file for the robot, building information modeling (BIM) model for the building, and computer-aided design (CAD) model for the pipe skid system) and within manageable durations (20 mins for the CW step and 20 mins for the CP step). This shows that DT simulation can serve as a management support tool for future construction robotic logistics planning.

In practice, these two steps do not have clear boundaries and should be adjusted according to how much flexibility can be granted to the design or operational changes. These two steps have distinct requirements on DT simulation. For the construction environment, general work site planning is completed first. However, further adjustments are needed for structural design based on a collision-free workspace. A collision-free workspace design is to identify a clearance and workspace that work for the most popular robotic systems. It also requires that the spatial parameters not affect the structural integrity of the structure. As a result, such a collision-free workspace design should happen after the general work site planning is done but before the structural design (e.g., structural analysis) is performed. This is to ensure that the collision-free workspace design meets the minimum structural integrity requirement. The collision-free workspace design needs to try out a large number of possible robotic movement trajectories, with randomly assigned specification parameters, to identify the common boundaries of a safe work zone. The efficiency of the DT simulation for a high number of simulation iterations is the top priority. In contrast, collision-avoidance path planning requires high-fidelity reproduction of the robot specifications and the established workspace to identify the best movement trajectory for robotic control. The precise simulation of the physical interaction and robot locomotion is more important. Although the literature has given examples of robotic simulations based on inverse kinematics or physics engines, little effort has been made to explain how these different robotic simulation approaches fit the unique needs for identifying collision-free workspaces and collision-avoidance operations. As a result, the main contribution of this paper is to provide evidence of how the two main robotic simulation approaches can meet the needs of the proposed two-step active collision avoidance




(a) (b)

**Fig. 11.** The joint trajectories of the midwaypoint approach.

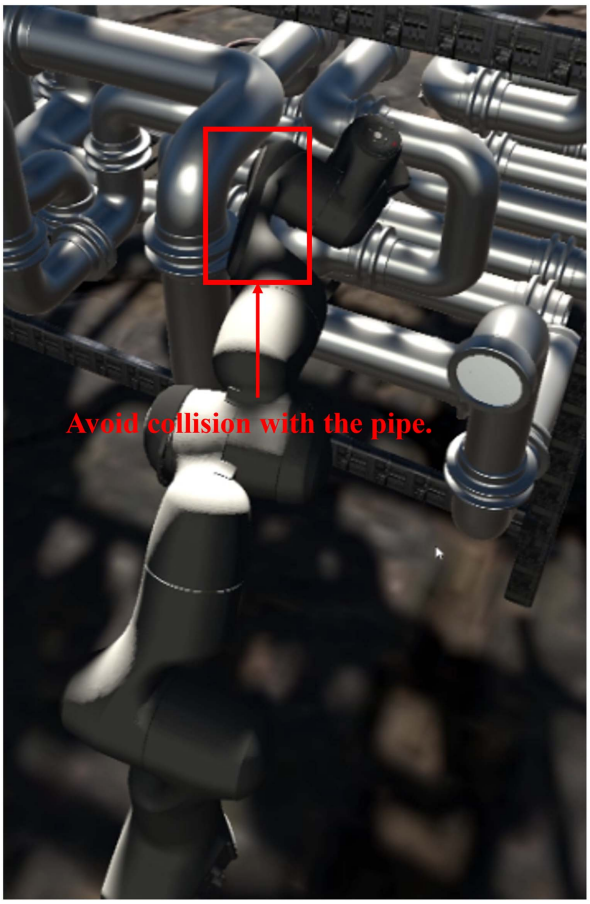



**Fig. 12.** The pose of each joint at the end of the trajectory.

planning in the early and late phases of a modular construction project.

## Limitations and Future Research

This study suffers from several limitations that require further investigations. First, further evidence is needed about how the proposed simulation tool can be used to optimize the selection, robot, and design of modular structures in the future in a more realistic setting with multiobjective requirements. In our current study, we analyzed the determination process of a single clearance spacing based on the distribution of the work envelope of an adjacent robot. In reality, such a determination can be more complicated with more clearance workspace considerations. For example, multiple clearance parameters may need to be determined instead of just one. In addition, there may be a more sophisticated consideration of the structural design and robot specifications in place that should be incorporated in the simulation, such as a maximum rack width due to spatial constraints. A more comprehensive multiobjective simulation will be performed in the future. Second, the robot platform base used in this simulation was fixed. However, there are certain scenarios that require a mobile platform to support a more flexible installation process. This means that a robot arm could be mounted on a moving platform, such as a unmanned ground robot (UGV) for more flexible locomotion in the work area. With a mobile platform, it is more complicated to identify the optimal parameters of the robot–structure work interface for reduced collision possibilities. As a result, it is our future agenda to incorporate scenarios of mobile robotic platforms and examine how they affect the proposed method. The last limitation concerns the robot-assisted automated workflow. In this study, to focus on the research scope, human workers were excluded from the workplace simulation. In addition, we did not consider dynamic obstacles such as reciprocal velocity obstacles. The incorporation of dynamic obstacles in our path planning would definitely make a great addition to our current work. As for this current paper, because we focus on facility maintenance in a fixed platform, the chance of encountering dynamic obstacles in the line of work is minimized. In addition, we found that the consideration of dynamic path planning would be another line of research that requires the velocity obstacle concept (Fiorini and Shiller 1998; Van den Berg et al. 2008). As a result, we will make it a future agenda item.

In future research, the simulation could be tested in more dynamic scenarios such as human workers and robots presenting in the same space or multiple robots collaborating with each other or with other dynamic moving objects in the range of the workspace. These dynamic influence factors would be included in the simulation to bring this method to real-world operation by working on extra prep work and remaining research. In future research, we will further test the use of reinforced learning in optimizing the weighting factors between CW and CP steps.

## Data Availability Statement

All data, models, or code generated or used during the study are available in a repository online in accordance with funder data retention policies (https://osf.io/jcz2y/).

## Acknowledgments

This material is supported by the National Science Foundation (NSF) under Grants 2024784 and 2128895 and the National Aeronautics and Space Administration (NASA) under Grant 80NSSC21K0815. Any opinions, findings, conclusions, or recommendations expressed in this article are those of the authors and do not reflect the views of the NSF or NASA.

## References

Aristidou, A., and J. Lasenby. 2009. *Inverse kinematics: A review of existing techniques and introduction of a new fast iterative solver.* Cambridge: Cambridge Univ.

Aristidou, A., J. Lasenby, Y. Chrysanthou, and A. Shamir. 2018. "Inverse kinematics techniques in computer graphics: A survey." *Comput. Graphics Forum* 37 (6): 35–58. https://doi.org/10.1111/cgf.13310.

Bajcsy, A., S. L. Herbert, D. Fridovich-Keil, J. F. Fisac, S. Deglurkar, A. D. Dragan, and C. J. Tomlin. 2019. "A scalable framework for real-time multi-robot, multi-human collision avoidance." In *Proc., 2019 Int. Conf. on Robotics and Automation (ICRA)*, 936–943. New York: IEEE.

Bengel, M., K. Pfeiffer, B. Graf, A. Bubeck, and A. Verl. 2009. "Mobile robots for offshore inspection and manipulation." In *Proc., 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 3317–3322. New York: IEEE.

Ben-Haim, Y. 2006. *Info-gap decision theory: Decisions under severe uncertainty.* Burlington: Elsevier.

Bock, T., and T. Linner. 2016. *Construction robots: Volume 3: Elementary technologies and single-task construction robots*. Cambridge: Cambridge University Press.

Bohigas, O., M. Manubens, and L. Ros. 2012. "A complete method for workspace boundary determination on general structure manipulators." *IEEE Trans. Rob.* 28 (5): 993–1006. https://doi.org/10.1109/TRO.2012.2196311.

Bonev, I. A., and J. Ryu. 2001. "A new approach to orientation workspace analysis of 6-DOF parallel manipulators." *Mech. Mach. Theory* 36 (1): 15–28. https://doi.org/10.1016/S0094-114X(00)00032-X.

Bragança, S., E. Costa, I. Castellucci, and P. M. Arezes. 2019. "A brief overview of the use of collaborative robots in industry 4.0: Human role and safety." In Vol. 202 of *Occupational and environmental safety and health. Studies in systems, decision and control*. Berlin: Springer. https://doi.org/10.1007/978-3-030-14730-3_68.

Buss, S. R. 2004. "Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods." *IEEE J. Rob. Autom.* 17 (1–19): 16.

Canutescu, A. A., and R. L. Dunbrack Jr. 2003. "Cyclic coordinate descent: A robotics algorithm for protein loop closure." *Protein Sci.* 12 (5): 963–972. https://doi.org/10.1110/ps.0242703.

Chen, H., S. Stavinoha, M. Walker, B. Zhang, and T. Fuhlbrigge. 2014. "Opportunities and challenges of robotics and automation in offshore oil & gas industry." *Intell. Control Autom.* 5 (3): 136–145. https://doi.org/10.4236/ica.2014.53016.

Craig, J. J. 2009. *Introduction to robotics: Mechanics and control, 3/E*. India: Pearson Education India.

Danaei, B., N. Karbasizadeh, and M. T. Masouleh. 2017. "A general approach on collision-free workspace determination via triangle-to-triangle intersection test." *Rob. Comput. Integr. Manuf.* 44 (Apr): 230–241. https://doi.org/10.1016/j.rcim.2016.08.013.

Delgado, J. M. D., L. Oyedele, A. Ajayi, L. Akanbi, O. Akinade, M. Bilal, and H. Owolabi. 2019. "Robotics and automated systems in construction: Understanding industry-specific challenges for adoption." *J. Build. Eng.* 26 (Nov): 100868. https://doi.org/10.1016/j.jobe.2019.100868.

Dritsas, S., and G. S. Soh. 2019. "Building robotics design for construction." *Constr. Rob.* 3 (1): 1–10. https://doi.org/10.1007/s41693-018-0010-1.

Du, J., Y. Shi, C. Mei, J. Quarles, and W. Yan. 2016. "Communication by interaction: A multiplayer VR environment for building walkthroughs." In *Proc., Construction Research Congress 2016*, 2281–2290, Reston, VA: ASCE.

Du, J., Y. Shi, Z. Zou, and D. Zhao. 2018a. "CoVR: Cloud-based multiuser virtual reality headset system for project communication of remote users." *J. Constr. Eng. Manage.* 144 (2): 04017109. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001426.

Du, J., Z. Zou, Y. Shi, and D. Zhao. 2017. "Simultaneous data exchange between BIM and VR for collaborative decision making." *Comput. Civ. Eng.* 2017 (Jun): 1–8. https://doi.org/10.1061/9780784480830.001.

Du, J., Z. Zou, Y. Shi, and D. Zhao. 2018b. "Zero latency: Real-time synchronization of BIM data in virtual reality for collaborative decision-making." *Autom. Constr.* 85 (Jan): 51–64. https://doi.org/10.1016/j.autcon.2017.10.009.

El-Sherbiny, A., M. A. Elhosseini, and A. Y. Haikal. 2018. "A comparative study of soft computing methods to solve inverse kinematics problem." *Ain Shams Eng. J.* 9 (4): 2535–2548. https://doi.org/10.1016/j.asej.2017.08.001.

FarzanehKaloorazi, M., M. T. Masouleh, and S. Caro. 2014. "Collision-free workspace of 3-RPR planar parallel mechanism via interval analysis." In *Advances in robot kinematics*, 327–334. Cham, Switzerland: Springer.

FarzanehKaloorazi, M., M. T. Masouleh, and S. Caro. 2017. "Collision-free workspace of parallel mechanisms based on an interval analysis approach." *Robotica* 35 (8): 1747–1760. https://doi.org/10.1017/S0263574716000497.

Ferdous, W., Y. Bai, T. D. Ngo, A. Manalo, and P. Mendis. 2019. "New advancements, challenges and opportunities of multi-storey modular buildings–A state-of-the-art review." *Eng. Struct.* 183 (Mar): 883–893. https://doi.org/10.1016/j.engstruct.2019.01.061.

Fiorini, P., and Z. Shiller. 1998. "Motion planning in dynamic environments using velocity obstacles." *Int. J. Rob. Res.* 17 (7): 760–772. https://doi.org/10.1177/027836499801700706.

Graf, B., and K. Pfeiffer. 2008. "Mobile robotics for offshore automation." In *Proc., EURON/IARP Int. Workshop on Robotics for Risky Interventions and Surveillance of the Environment*. Chicago, IL: International Association of Rehabilitation Professionals.

Graf, B., K. Pfeiffer, and H. Staab. 2007. "Mobile robots for offshore inspection and manipulation." In *Proc., IPTC 2007: Int. Petroleum Technology Conf.* Bunnik, Netherlands: European Association of Geoscientists & Engineers.

Haddadin, S., A. Albu-Schaffer, A. De Luca, and G. Hirzinger. 2008. "Collision detection and reaction: A contribution to safe physical human-robot interaction." In *Proc., 2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 3356–3363. New York: IEEE.

Haddadin, S., A. De Luca, and A. Albu-Schäffer. 2017. "Robot collisions: A survey on detection, isolation, and identification." *IEEE Trans. Rob.* 33 (6): 1292–1312. https://doi.org/10.1109/TRO.2017.2723903.

Hart, P. E., N. J. Nilsson, and B. Raphael. 1968. "A formal basis for the heuristic determination of minimum cost paths." *IEEE Trans. Syst. Sci. Cybern.* 4 (2): 100–107. https://doi.org/10.1109/TSSC.1968.300136.

Institute, A. N. S. 2012. *ANSI/RIA R15.06-2012 American national standard for industrial robots and robot systems—Safety requirements*. Ann Arbor, MI: Robotic Industries Association.

ISO. 1998. *Manipulating industrial robots—Performance criteria and related test methods*. Genève: International Organization of Standards.

Jaillon, L., C.-S. Poon, and Y. H. Chiang. 2009. "Quantifying the waste reduction potential of using prefabrication in building construction in Hong Kong." *Waste Manage.* 29 (1): 309–320. https://doi.org/10.1016/j.wasman.2008.02.015.

Karaman, S., and E. Frazzoli. 2011. "Sampling-based algorithms for optimal motion planning." *Int. J. Rob. Res.* 30 (7): 846–894. https://doi.org/10.1177/0278364911406761.

Kenwright, B. 2012. "Inverse kinematics—Cyclic coordinate descent (CCD)." *J. Graphics Tools* 16 (4): 177–217. https://doi.org/10.1080/2165347X.2013.823362.

Latombe, J.-C. 2012. *Robot motion planning*. Berlin, Heidelberg: Springer Science & Business Media.

Lee, D., N. Ku, T.-W. Kim, J. Kim, K.-Y. Lee, and Y.-S. Son. 2011. "Development and application of an intelligent welding robot system for shipbuilding." *Rob. Comput. Integr. Manuf.* 27 (2): 377–388. https://doi.org/10.1016/j.rcim.2010.08.006.

Lempert, R. J. 2003. *Shaping the next one hundred years: New methods for quantitative, long-term policy analysis*. Santa Monica, CA: RAND.

Long, P., T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan. 2018. "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning." In *Proc., 2018 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 6252–6259. New York: IEEE.

Luenberger, D. G., and Y. Ye. 1984. *Linear and nonlinear programming*. Cham, Switzerland: Springer.

Mainprice, J., and D. Berenson. 2013. "Human-robot collaborative manipulation planning using early prediction of human motion." In *Proc., 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 299–306. New York: IEEE.

Meredith, M., and S. Maddock. 2005. "Adapting motion capture data using weighted real-time inverse kinematics." *Comput. Entertainment* 3 (1): 5. https://doi.org/10.1145/1057270.1057281.

Merlet, J.-P. 1999. "Determination of 6D workspaces of Gough-type parallel manipulator and comparison between different geometries." *Int. J. Rob. Res.* 18 (9): 902–916. https://doi.org/10.1177/02783649922066646.

Mignacca, B., G. Locatelli, M. Alaassar, and D. C. Invernizzi. 2018. "We never built small modular reactors (SMRs), but what do we know about modularization in construction?" In *Proc., Int. Conf. on Nuclear Engineering*. New York: American Society of Mechanical Engineers.

Park, C., J. Pan, and D. Manocha. 2013. "Real-time optimization-based planning in dynamic environments using GPUs." In *Proc., 2013 IEEE Int. Conf. on Robotics and Automation*, 4090–4097. New York: IEEE.

Pinosofa, A., A. Ramirez, O. S. Cortazar Cruz, Y. Ravelo, and G. Yermagaliyeva. 2010. "Unmanned offshore platforms: Automation kit." In *Proc., Trinidad and Tobago Energy Resources Conf.* Dallas, TX: OnePetro.

Quade, E. S., and G. M. Carter. 1989. *Analysis for public decisions*. Cambridge: MIT Press.

Sciavicco, L., and B. Siciliano. 2012. *Modelling and control of robot manipulators*. London: Springer Science & Business Media.

Shi, Y., J. Du, E. Ragan, K. Choi, and S. Ma. 2018. "Social influence on construction safety behaviors: A multi-user virtual reality experiment." In *Proc., Construction Research Congress*, 147–183. Reston, VA: ASCE.

Shukla, A., and H. Karki. 2016. "Application of robotics in offshore oil and gas industry—A review Part II." *Rob. Auton. Syst.* 75 (Jan): 508–524. https://doi.org/10.1016/j.robot.2015.09.013.

Sisbot, E. A., and R. Alami. 2012. "A human-aware manipulation planner." *IEEE Trans. Rob.* 28 (5): 1045–1057. https://doi.org/10.1109/TRO.2012.2196303.

Skourup, C., J. Pretlove, N. Stembridge, and M. Svenes. 2008. "Enhanced awareness for offshore teleoperation." In *Proc., Intelligent Energy Conf. and Exhibition*. Dallas, TX: OnePetro.

Snyman, J., L. Du Plessis, and J. Duffy. 2000. "An optimization approach to the determination of the boundaries of manipulator workspaces." *J. Mech. Des.* 122 (4): 447–456. https://doi.org/10.1115/1.1289388.

Suita, K., Y. Yamada, N. Tsuchida, K. Imai, H. Ikeda, and N. Sugimoto. 1995. "A failure-to-safety 'Kyozon' system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence." In *Proc., 1995 IEEE Int. Conf. on Robotics and Automation*, 3089–3096. New York: IEEE.

Takakura, S., T. Murakami, and K. Ohnishi. 1989. "An approach to collision detection and recovery motion in industrial robot." In *Proc., 15th Annual Conf. of IEEE Industrial Electronics Society*, 421–426. New York: IEEE.

Tam, V. W., C. M. Tam, S. Zeng, and W. C. Ng. 2007. "Towards adoption of prefabrication in construction." *Build. Environ.* 42 (10): 3642–3654. https://doi.org/10.1016/j.buildenv.2006.10.003.

Tan, Y., Y. Song, X. Liu, X. Wang, and J. C. Cheng. 2017. "A BIM-based framework for lift planning in topsides disassembly of offshore oil and gas platforms." *Autom. Constr.* 79 (Jul): 19–30. https://doi.org/10.1016/j.autcon.2017.02.008.

Van den Berg, J., M. Lin, and D. Manocha. 2008. "Reciprocal velocity obstacles for real-time multi-agent navigation." In *Proc., 2008 IEEE Int. Conf. on Robotics and Automation*, 1928–1935. New York: IEEE.

Walker, W. E., R. J. Lempert, and J. H. Kwakkel. 2013. "Deep uncertainty." In *Encyclopedia of operations research and management science*. 3rd ed., edited by S. Gass and M. Fu. Berlin: Springer. https://doi.org/10.1007/978-1-4419-1153-7.

Wang, L.-C., and C.-C. Chen. 1991. "A combined optimization method for solving the inverse kinematics problems of mechanical manipulators." *IEEE Trans. Rob. Autom.* 7 (4): 489–499. https://doi.org/10.1109/70.86079.

Wang, Z., S. Ji, Y. Li, and Y. Wan. 2010. "A unified algorithm to determine the reachable and dexterous workspace of parallel manipulators." *Rob. Comput. Integr. Manuf.* 26 (5): 454–460. https://doi.org/10.1016/j.rcim.2010.02.001.

Welman, C. 1993. "Inverse kinematics and geometric constraints for articulated figure manipulation." Theses, School of Computing Science, Simon Fraser Univ.

Więckowski, A. 2017. "'JA-WA'-A wall construction system using unilateral material application with a mobile robot." *Autom. Constr.* 83 (Nov): 19–28. https://doi.org/10.1016/j.autcon.2017.02.005.

Yang, X., H. Wang, C. Zhang, and K. Chen. 2010. "A method for mapping the boundaries of collision-free reachable workspaces." *Mech. Mach. Theory* 45 (7): 1024–1033. https://doi.org/10.1016/j.mechmachtheory.2010.02.002.

Zhang, S., Z. Li, Y. Wang, and Z. Zhang. 2020. "Design on the inner wall crawling and inspecting robot for offshore platform leg." *Recent Pat. Mech. Eng.* 13 (2): 109–117. https://doi.org/10.2174/2212797613666200117121403.

Zhou, T., Q. Zhu, and J. Du. 2020. "Intuitive robot teleoperation for civil engineering operations with virtual reality and deep learning scene reconstruction." *Adv. Eng. Inf.* 46 (Oct): 101170. https://doi.org/10.1016/j.aei.2020.101170.

Zhu, Q., J. Du, Y. Shi, and P. Wei. 2021. "Neurobehavioral assessment of force feedback simulation in industrial robotic teleoperation." *Autom. Constr.* 126 (Jun): 103674. https://doi.org/10.1016/j.autcon.2021.103674.