Automating Crane Lift Path through Integration of BIM and Path Finding Algorithm

Songbo Hu[a] and Yihai Fang[a]

[a]Department of Civil Engineering, Monash University, Australia
E-mail: Songbo.Hu@monash.edu, YihaiFang@monash.edu

Abstract -

Path planning, as a primary mission in crane lift planning, has a profound and direct impact on the safe and efficient execution of lift jobs on construction sites. Typically, the main objective of path planning is to find the shortest (or a relatively short) and collision-free path from the load supply point to the demand point in a finite 3D space with the presence of obstacles, while considering the mechanical constraints of the crane. Despite collision-free being a primary criterion in path planning, other critical safety issues have not been adequately addressed in existing methods. For example, blind lifts and reduced visibility of crane operators are prevalent in construction and have been recognized as major safety concerns in crane lifts. Furthermore, complex coordination of crane motions and frequent changes of moving direction potentially lead to human errors. To address this gap in the knowledge, this paper proposes a semi-automated planning approach by using Building Information Modeling (BIM) and an intelligent path finding algorithm. First, dimension and weight data of building components to be lifted are retrieved from the BIM model. Then, a modified RRT* algorithm is used to generate a short and collision-free path that satisfies the desired level of path smoothness, visibility, and motion coordination. Finally, paths for all lifts are stored and associated with corresponding elements in BIM for easy analysis and visualization. Preliminary results show that the proposed method can effectively reduce the occurrence of blind lifts while ensuring a practical path for execution. In the future, the proposed method is expected to enable a BIM-based risk analysis tool for the safety of crane operation and its impact on other adjacent construction activities.

Keywords -

Building Information Modeling (BIM); Path planning Algorithm; Tower Cranes; Practicality; Construction Safety; Lift Planning

1 Introduction

A crane is a piece of indispensable equipment on construction sites, undertaking both vertical and horizontal transportation [1]. However, it is also one of the main contributors to the construction fatality [2]. According to various accident analysis research, these fatal accidents related to cranes were primarily caused by inappropriate positioning of the lift load, which either violates the maximum reach or has spatial conflicts with workers, machinery, existing structures, and prohibited area [3]. Therefore, planning the load position during the lift, which is also known as path planning, is an essential procedure to avert safety hazards. In conventional practice, path planning is manual and based on experience, usually leading to a time-consuming process and error-prone outcomes [4].

In recent decades, researchers have made significant efforts to optimize and automate the path planning process. Multiple pieces of literature utilized the robotic path finding algorithms to generate the shortest and collision-free path abide by the load chart [5]. This method is able to shorten the planning time and eliminate multiple safety hazards, such as crane tipping-over and collisions between the load and static obstacles [6]. However, using robotic path finding algorithms is subjected to a number of strong assumptions, leading to incomplete considerations of safety hazards. For example, it is assumed that the operator has the ability to precisely execute the planned path. In reality, however, it is impractical for a human operator to carry out over-complicated maneuvers such as blind lifts [7] or complex coordination of swinging, luffing, and hoisting [8]. Meanwhile, most algorithms assume that the obstacles are static [9]. This assumption is inconsistent with the dynamic nature of construction sites and ignores the risk of spatial conflicts between the lift load and dynamic obstacles. Although in recent years, several novel frameworks enabled the path re-planning based on real-time monitoring of moving objects [10], these works sorely rely on the crane operators to alter the path on the spot and cannot coordinate with the affected objects and workspaces to avoid spatial conflicts.

To mitigate the spatial conflicts between the crane and other workspaces, one strategy is to visualize the crane workspace in the planning stage based on crane parameters [11]. It helps the construction stakeholders to understand the spatial relationships between the crane and dynamic/static obstacles and to identify potential safety hazards proactively. This strategy usually involves data visualization and analysis, on an effective information management platform. Building Information Modelling (BIM) has been exploited as a core data generator for risk management tools to demonstrate geometric information, analyze spatial conflicts, and communicate safety risks [12]. In such an approach, the parametric crane workspace is usually over-estimated and unable to explicitly reflect the location of the lift load [13]. There is a demand for a realistic path finding algorithm that guides and predicts the trajectory of lift loads. The algorithm-planned results for each building component need to be stored in BIM for further risk analysis.

Therefore, this paper proposes a novel approach that integrates a modified RRT* path finding algorithm and BIM to create a decision support tool for proactive path planning. Firstly, the original RRT* algorithm is improved to incorporate common practicality considerations. These practicality considerations determine the operational complexity for a crane operator and profoundly impact the crane lift safety, including avoiding frequent turning, minimizing blind lifts, and limiting the coordination of crane motions. Secondly, BIM is utilized to interoperate with the modified RRT* algorithm bi-directionally, which provides information to the algorithm and visualizes its output.

In the rest of this paper, Section 2 introduces the background information about the practicality considerations in path finding algorithms with respect to crane safety, as well as the potential application of BIM in this paper. Section 3 introduces the proposed methods, including the overall framework and the development of a prototype system. The proposed methods are validated and demonstrated in a case study in Section 4. Section 5 concludes the paper and indicates future directions.

2 Background

2.1 Path Finding Algorithm

Planning the lift path is a critical task in the lift planning process. In reality, an experienced human path planner not only avoids collisions between the lift load and obstacles but balances the operational complexity and path length [14]. A number of studies have been proposed to automate this task with path finding algorithms, including A* [15], Genetic Algorithm (GA) [16], Probability Roadmap [7], and Rapid Random-exploring Tree algorithm (RRT) [17]. These algorithms were designed to find a collision-free path within its kinematic ranges and comply with capacity limits. However, rather than applying these algorithms directly, these studies had to scrutinize the characteristics of crane lifting and modified the algorithms accordingly to generate a practical path with acceptable operational complexity. These unique considerations are referred to as practicality considerations in this study.

Three practicality considerations have been discussed in the related work, including the smoothness of a path, the blind lifts, and the motion coordination. First, a smooth path benefits construction safety since it requires gentle maneuvers and reduces the likelihood of human errors [18]. However, “smoothness” is a vague description and researchers have divergent opinions on what it embodies. For example, Ali et al. [5] defined a roughness index, which accumulated the angular displacements for each point on the paths. Together with the path length, the roughness index was integrated into the objective function for optimization. The description of operability using path roughness was later adopted by Zhang and Hammad in an RRT-based path planning method [18]. This interpretation of smoothness yields a continuous path and prevents abrupt changes in moving direction. Similar efforts can be found in [17], which employs a spline function to remove acute angles in the path. These efforts smoothen a path to prevent abrupt turnings, but it requires constant and complicated adjustment of moving directions. To address this issue, other researchers defined a smooth path as a path with fewer motion switches and consequentially fewer way-points. To remove redundant way-points, two strategies have been applied, which either adds operation switching cost to the objective function [19] or post-process a candidate path via the “straight-line strategy” [20]. These two strategies have both strengths and weaknesses. Using an operation switching cost can guide the path searching process to find a smooth path, but the definition of cost is subjective [20]. Also, the “straight-line strategy” is effective but it cannot compare the “smoothened” path with other alternatives generated during the planning process.

Motion coordination is another factor determining the operational complexity and thus researchers attempted to limit the number of coordinated motions in path planning. For example, Olearczyk et al. [21] avoided the coupling of crane rotation (i.e., swinging) and translation (i.e., luffing, hoisting, or both) to decrease the operational complexity. Chi et al. [7]assumed that crane operators can at most coordinate two motions simultaneously and applied this assumption to the generation of the roadmap using the probabilistic roadmap (PRM) algorithm. Cai et al. [16] adopted similar measures of prohibiting three simultaneous motions out of four degrees of freedom (i.e., swinging, luffing, hoisting and load rotation) and eliminated the complex motion coordination in post-processing. While these pieces of research followed an overwhelmingly stringent restriction on motion coordination, Hung et al. [22] allowed the combination of all crane motions and designed a user-defined parameter to limit the maximum number of coordinated motions.

In addition to being smooth and with a minimum level of motion coordination, a practical path should avoid blind lifts and ensures clear visibility to the load. Good visibility is essential for the crane operator to gain acute situational awareness, which helps to recognize and mitigate the severe safety risks [23]. In addition to safety concerns, blind lifts often compromise efficiency as the operator has to maneuver slowly with extra caution while communicating with the signal person [24]. To account for the extra time due to lifts in blind areas, researchers often estimate the total lift time using a mathematical model where poor visibility incurs a time penalty [25]. Abundant studies were devoted to enhancing the vision of crane operators [4]. However, there still lacks precaution against blind lifts in the planning stage [26]. Among the literature reviewed, only Chi et al. [7]examined the visibility of the path generated by a PRM algorithm. If the candidate path has any invisible segment, they execute PRM for an extra iteration until the path is fully visible. Although this method ensures the result to be visible, it did not modify the mechanism of PRM itself. Thus, the computational efficiency and success rate are subject to randomness due to the nature of PRM.

To accommodate these three practicality considerations, the RRT* algorithm is selected as the base for further modifications. RRT*, which is a variant of the classic RRT algorithm, is designed to efficiently search a high-dimensional space and converges towards the shortest collision-free path [27]. The optimality in the path length comes from two searching procedures, “Choose Parent” and “Rewire”. These two procedures dynamically reduce redundancy in the path set and remove unnecessary movements along every alternative path, which not only shorten the path length but smoothen the path. Additionally, the searching mechanism of the RRT* algorithm is highly adaptable, which allows modifications to consider visibility and motion coordination.

2.2 Building Information Modelling (BIM)

BIM is an emerging research focus in construction risk management [12]. In this task, BIM serves as two fundamental roles: the core data reservoir which provide baseline data to BIM-based risk management tools; and a visualization platform that enhances the identification, communication, and prevention of safety hazards [28]. Specific to the crane, BIM can both provide component information for automated lift planning [13], and visualize the paths and workspace of the crane to help stakeholders identify potential spatial conflicts [29]. For example, Ji and Leite [13] proposed an automatic rule-based checking system for reviewing a lift plan by using a 4D BIM model as the information source to reduce tedious manual input. Wang et al. [30] used BIM as an information source for a location optimization algorithm (i.e., firefly algorithm) and further visualized the algorithm-planned crane locations.

Despite some attempts, path planning, as a critical planning task, has not leveraged full benefit of BIM yet. Integrating BIM and the path finding algorithms is expected to not only eliminate tedious manual inputs, but explicitly predict the risks of cranes by analyzing the spatial relationship between an algorithm-planned path and existing structures in the pre-construction phase.

3 Methodology

3.1 Framework

This paper proposes a novel approach to improve the safety of lifting activities in the pre-construction phase via the integration of a practicality-and-safety-aware RRT* algorithm (PSRRT*), and BIM. Figure 1 presents the overall structure of the framework, which has three components: information retrieval, path generation, and path visualization. This framework starts with a query in the BIM model to retrieve information relevant to the building components to be lifted, the site area, existing structures, and the crane specifications. The retrieved information formulates the search space for PSRRT* who generates a path that satisfies the practicality considerations specified. The geometry of the path is then stored and visualized in the BIM model with invisible parts highlighted.

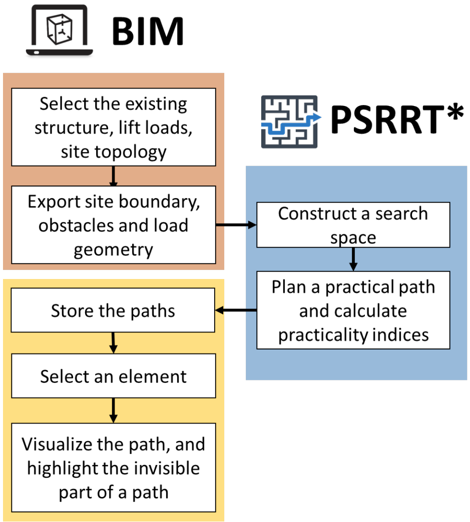


Figure 1. Research framework

3.2 Prototype Development

To validate the proposed framework, a prototype system was developed using Autodesk Revit, Dynamo Studio, and Python programming language. Python was employed to realize the PSRRT* algorithm while Revit and Dynamo were used to enable the data exchange between the BIM model and PSRRT*.

3.2.1 Information Retrieval

The first step in the prototype system involves the search for and retrieve of lift-related information from a Revit model using Dynamo. The information to be retrieved includes (1) the geometry and location of existing structures; (2) the ID, geometry, location, and weight of the lift load; and (3) the geometry of site topology. In addition to the building information, crane specifications are also required. In some cases, there exist detailed crane information models, which include the crane specifications such as the capacity, the boom length, and location of the crane and its cabin. However, more often, the crane information has a source other than the BIM model, such as a spreadsheet from crane manufacturers. Therefore, this step focuses on general cases and retrieves building information only.

Based on the retrieved information, the search space of PSRRT* can be constructed, with the location and dimension of obstacles and the space boundary. First, building structures existing prior to the lift are given axis-aligned minimum bounding boxes (AABB), which are further buffered with the length and height of the lift load. The buffered boxes are used to represent obstacles in the search space. To minimise computational complexity, existing structures with a top constraint lower than the base constraint of the lifted load in the BIM model are consolidated to be one single obstacle. Then, the search space boundary is determined by the site topology, which is presented by toposurface in Revit. Figure 2 shows the Dynamo code overview for retrieving necessary information from Revit models to construct the search space for PSRRT*. As a result, three .csv files are exported, which contain information on obstacles, the lift load, and the site boundary.

3.2.2 Path Generation

As an integral component in this framework, PSRRT* is devised to find a practical path given the search space, obstacles, and load supply and demand points. Although the algorithm can easily adapt to other types of cranes, in this study, PSRRT* is designed for cranes with three DoFs (i.e., swinging, luffing, and hoisting), such as the luffing tower crane or the truck mobile crane. As discussed in Section 2.1, a practical path is defined as a smooth and visible path with an acceptable level of motion coordination. As the original RRT* algorithm can produce a smooth path, PSRRT* mainly aims to improve the outcomes' practicality in visibility and motion coordination.

The flowchart in Figure 3 illustrates how PSRRT* explores the search space and makes adjustments to reflect the practicality considerations. The searching mechanism relies on a "tree" with the "root" at the initial point of the lift load. The tree is defined as vertices and edges that connect the parent vertices and its child vertices. Since one child vertex only has one parent vertex, a path connecting the initial point to the goal point is found when the tree grows to reach the goal. To expand the tree, PSRRT* randomly samples a node (q_{rand}) in the search space and attempts to connect q_{rand} to the nearest node ($q_{nearest}$) on the existing tree rooted in the initial node ($q_{initial}$). If the distance between $q_{nearest}$ and q_{rand} is longer than a given resolution for tree expansion (Δq), a new node q_{new} is created Δq away from $q_{nearest}$ in the direction towards q_{rand}.

If q_{new} is visible and collision-free, the nearby nodes (q_{near}) are found. For each q_{near}, if the edge between q_{new} and q_{near} is visible and collision-free, the cumulate distance from $q_{initial}$ to q_{new} via q_{near} and the distance between q_{near} to q_{goal} are calculated. These two distances are summed up as a total distance to reach q_{new}. The q_{near} with the smallest total distance is considered as the parent of q_{new}. This procedure is known as "Choose Parent". It is followed by the "Rewire", which measures the cumulative distance from $q_{initial}$ to q_{near} via q_{new} if the in-between edge is visible and collision-free, and changes the parent of q_{near} to q_{new}. This loop iterates until any tree

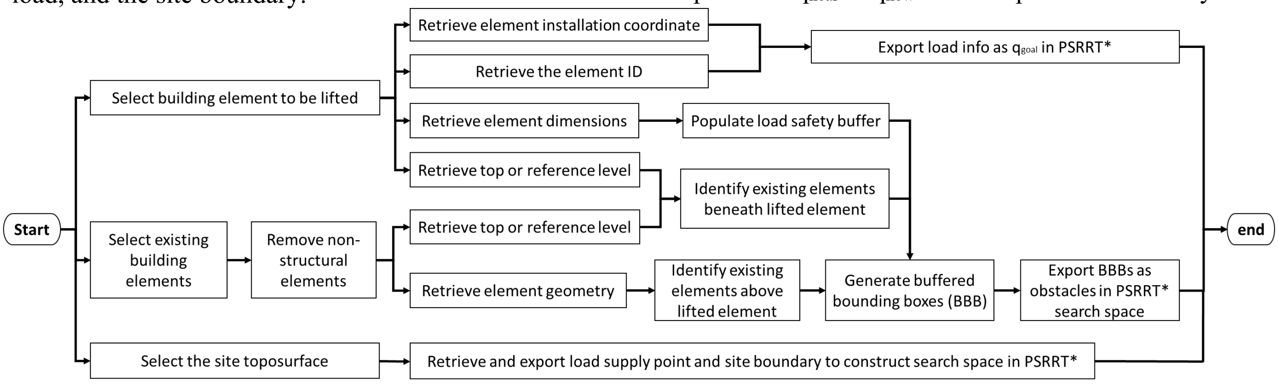


Figure 2. Flowchart for information retrieval from BIM model to PSRRT*

node can be connected to q_{goal} with a collision-free and visible edge.

In this process, the motion coordination is limited by redefining distance in the algorithm. Usually, distance is defined as the Euclidean distance, which neglects the complex coordination of motions in the configuration space (C-space). As a result, PSRRT* uses a weighted Manhattan distance to model the cost between any two configurations:

$$\text{Cost} = w_1 \Delta \alpha + w_2 \Delta \theta + w_3 \Delta \mathrm{L}$$

Where α indicates the swing angle, θ indicates the luffing angle, L indicates the length of the hoist line, which connects the tip of the crane boom and the lift load; weights (i.e., w_1, w_2, w_3) are determined according to the operating speed of each crane motion.

On the other hand, blind lifts are avoided via visibility checks. The visibility check uses the AABB collision detection technique to examine the spatial relationship between the points along the vision line and the physical obstacles. The vision lines are straight lines in the Cartesian space, starting from the crane cabin. It is worth noting that a search space usually has two representations: a Cartesian space and a C-space, and these two representations are transformable. In this algorithm, tree expansion is conducted in the C-space since it is convenient to identify motion coordination, while the collision detections and visibility checks are implemented in the Cartesian space.

Once a valid path is found, PSRRT* executes a post-processing procedure, to further eliminate redundant motion coordination by restricting the first and last segments of the path to be vertical (i.e., only hoisting is allowed) to reflects the real practice. If they are collision-free, the algorithm returns a new path that is exported to Revit for visualization.

3.2.3 Path Visualization

Although the algorithm-planned path has already incorporated most practicality considerations, there still exist various safety concerns in path planning, depending on the site condition and the characteristics of lift loads. Therefore, it is necessary for construction stakeholders to review the algorithm-planned path in a context-rich manner. Figure 4 shows an overview of the dynamo code for path visualization. By selecting the building component to be reviewed, this component automatically reads the path from an excel and presents the path geometry with the invisible parts highlighted.

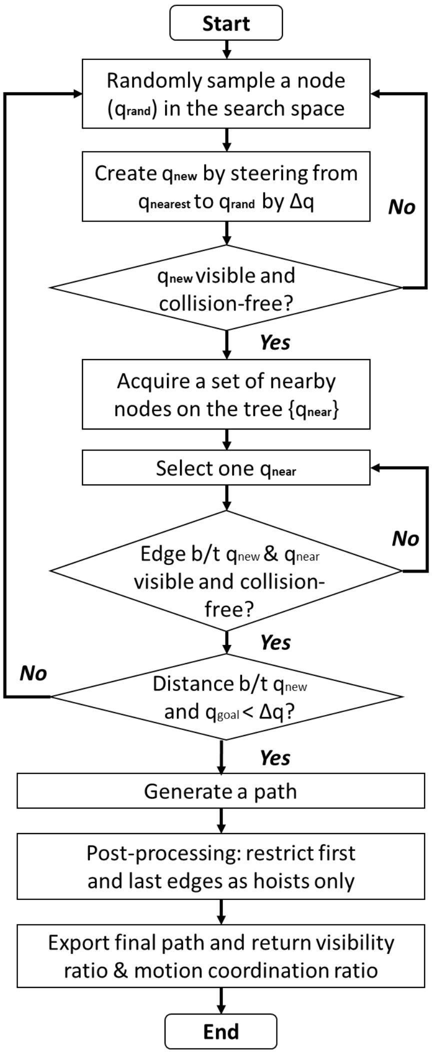


Figure 3. PSRRT* path generation mechanism

4 Case Study

The performance of the prototype system was assessed in a case study on a building project at Monash University, Melbourne, Australia. The building project consisted of a 5-story steel structure errected by three 28m-high luffing boom tower cranes with a 40m boom. The model of these tower cranes is FAVCO M390D and the deployment locations for each crane is indicated in Figure 5(a). According to the schedule, Tower Crane 3 (TC3) was required to lift three beams to level 1 in one working day, as shown in Figure 5(b). These lifting activities are selected as the test scenario in this case study. The building information was stored in a Revit model. Meanwhile, since there was no available Revit family for this particular crane model, the crane specifications and location of the supply area were stored in a .csv file.

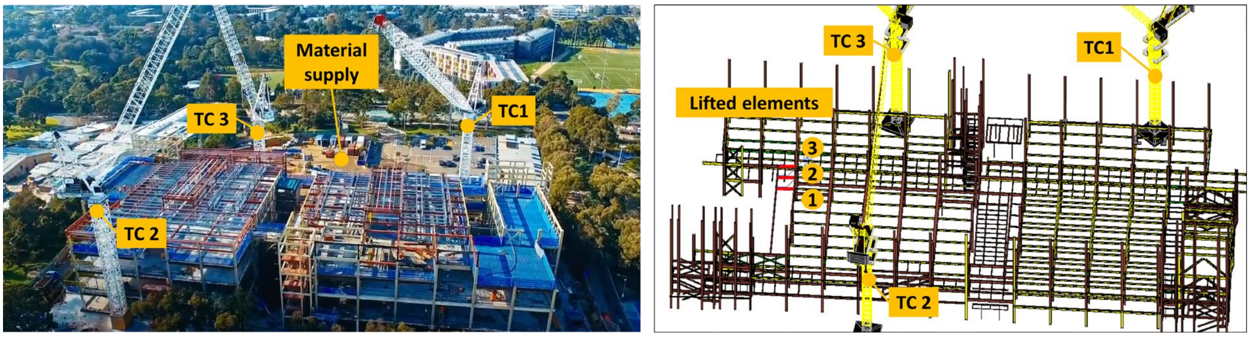
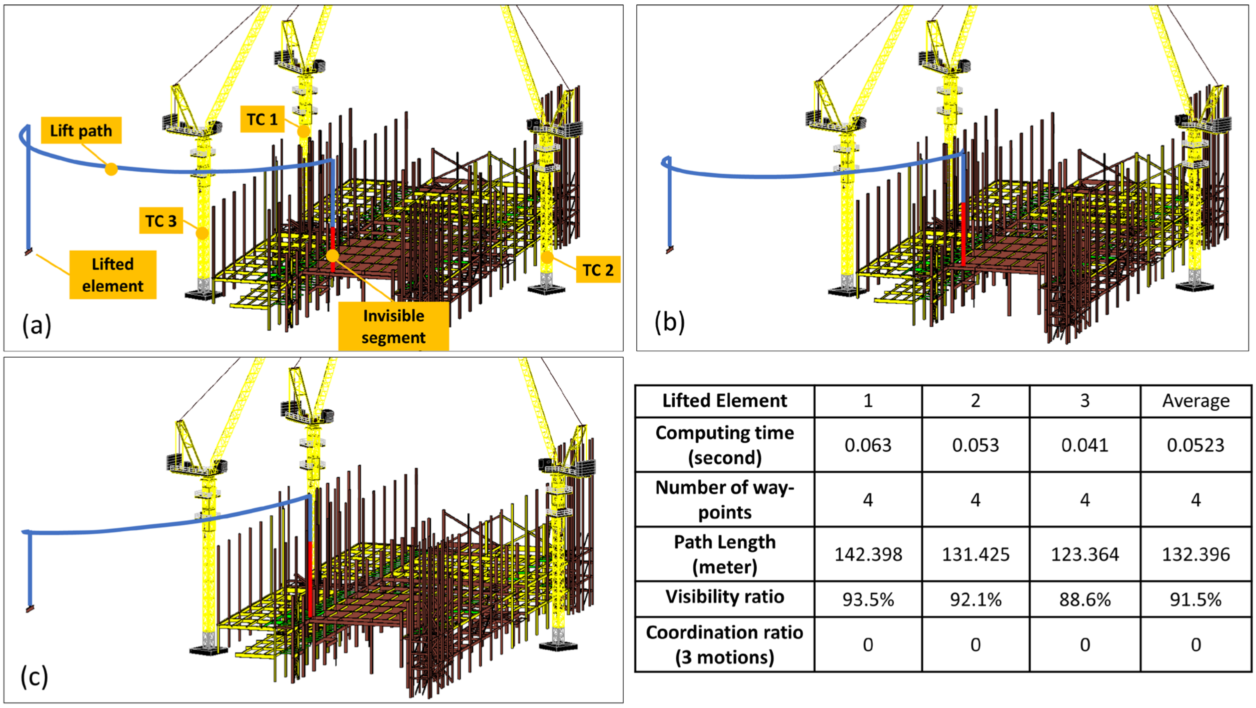


Figure 4. Overview of the project (left) and building elements to be lifted in the BIM model (right)

Firstly, the prototype system retrieved information from the Revit model. In this process, there were 1390 elements retrieved as structures existing prior to the lifts. These elements were exported as 503 obstacles in a 150m x 140m area. Based on this information, PSRRT* calculated the paths for the three lift tasks, as shown in Figure 6(a-c). All three paths have similar trajectories, where the load is vertically lifted from the material supply area, then moved to above the demand points through a combination of swinging and luffing, and finally lower the loads to the target elevation for installation. As illustrated in Figure 6(d), although the number of obstacles is high, the computing times of PSRRT* are relatively low. The average computing time is 0.0523s for the paths. For a project with 3000 elements to be installed, the proposed method is anticipated to finish the planning task within 3 minutes. This prediction requires further verification since RRT* is stochastic and its computing time is influenced by random sampling. Meanwhile, to quantitively evaluate the improvement of the proposed path finding algorithm in smoothness, visibility, and motion coordination, Figure 6(d) also tabulated the number of way-points, the ratio of the visible path to the entire path, and the ratio of the path with three coordinated motions to the entire path. It is observed that the average visibility ratio is 91.5%, and the invisible parts for each path are visible to the beam connector (i.e., the last segment). Furthermore, the coordination ratio for these three paths is zero, indicating an optimal complexity of operations.

Despite the encouraging outcome from the proposed approach, several limitations of the proposed methods are also identified. For example, the proposed method assumes the structures below the lift load as a rectangular



Lifted Element	1	2	3	Average
Computing time (second)	0.063	0.053	0.041	0.0523
Number of way-points	4	4	4	4
Path Length (meter)	142.398	131.425	123.364	132.396
Visibility ratio	93.5%	92.1%	88.6%	91.5%
Coordination ratio (3 motions)	0	0	0	0

Figure 5. Planned paths for the lifted elements (1) to (3) in (a) to (c), respectively, and results for computing time, path length, visibility ratio and coordination ratio for each path

obstacle to decrease computing complexity, which is only valid for building projects with a rectangular geometry. This limitation can be overcome by using advanced collision detection algorithms (e.g., OBB) or voxelizing the search space. Furthermore, the proposed approach requires a comprehensive BIM model, which is not always available. Essential information for path planning, including lift sequences and the crane specifications are usually stored in other software, causing difficulty in interoperability. Therefore, there is a demand to create and standardize customized parameters and family types to store the necessary information for lift planning. Thirdly, the proposed approach focuses on the BIM model and may ignores environmental obstacles (e.g., adjacent buildings). In congested environments, the BIM model is expected to fuse with reality capturing technologies, such as laser scanning, to acquire information on site conditions.

5 Conclusion and Future Research

This paper proposes an approach to automatically plan and visualize a safety and practical lifting path using a novel path finding algorithm and BIM. Two main contributions can be highlighted: (1) the modified RRT* algorithm successfully incorporates the practicality considerations to generate a feasible and realistic path. (2) the proposed approach exploited the BIM model to visualize the planning results to facilitate coordination between stakeholders.

Furthermore, this paper demonstrated the potential of the proposed approach as a 4D risk management system where engineers can proactively plan the activities and workspaces under the impact of crane lift activities. In the future, the proposed method is expected to automatically plan paths for the entire building project for lifting experts to review and adjust. By analyzing adjusted paths, the hazardous zone of crane swing-over is generated to support safety-related decisions, such as coordinating active workspace and crane operations.

References

[1] R. Li, Y. Fu, G. Liu, C. Mao, P. Xu, An algorithm for optimizing the location of attached tower crane and material supply point with BIM, in: ISARC 2018 - 35th International Symposium on Automation and Robotics in Construction and International AEC/FM Hackathon: The Future of Building Things, 2018. https://doi.org/10.22260/isarc2018/0057.

[2] E. Gharaie, H. Lingard, T. Cooke, Causes of fatal accidents involving cranes in the australian construction industry, Construction Economics and Building. 15 (2015) 1–12. https://doi.org/10.5130/AJCEB.v15i2.4244.

[3] J.E. Beavers, J.R. Moore, W.R. Schriver, R. Rinehart, Crane-Related Fatalities in the Construction Industry, Journal of Construction Engineering and Management. 132 (2006) 901–910.

[4] Y. Fang, Y.K. Cho, J. Chen, A framework for real-time pro-active safety assistance for mobile crane lifting operations, Automation in Construction. (2016). https://doi.org/10.1016/j.autcon.2016.08.025.

[5] M.S.A.D. Ali, N.R. Babu, K. Varghese, Collision Free Path Planning of Cooperative Crane Manipulators Using Genetic Algorithm, (2005) 182–193.

[6] J. An, M. Wu, J. She, T. Terano, Re-optimization strategy for truck crane lift-path planning, (2018). https://doi.org/10.1016/j.autcon.2018.02.029.

[7] H.L. Chi, S.C. Kang, S.H. Hsieh, X. Wang, Optimization and evaluation of automatic rigging path guidance for tele-operated construction crane, in: 31st International Symposium on Automation and Robotics in Construction and Mining, ISARC 2014 - Proceedings, 2014: pp. 738–745.

[8] W.H. Hung, C.W. Liu, C.J. Liang, S.C. Kang, Strategies to accelerate the computation of erection paths for construction cranes, Automation in Construction. 62 (2016) 1–13. https://doi.org/10.1016/j.autcon.2015.10.008.

[9] X. Wang, Y.Y. Zhang, D. Wu, S. De Gao, Collision-Free Path Planning for Mobile Cranes Based on Ant Colony Algorithm, Key Engineering Materials. 467–469 (2011) 1108–1115. https://doi.org/10.4028/www.scientific.net/KEM.467-469.1108.

[10] J. Zhang, F. Yu, D. Li, Z. Hu, Development and implementation of an industry foundation classes-based graphic information model for virtual construction, Computer-Aided Civil and Infrastructure Engineering. (2014). https://doi.org/10.1111/j.1467-8667.2012.00800.x.

[11] J.K.W. Yeoh, J.H. Wong, L. Peng, Integrating crane information models in BIM for checking the compliance of lifting plan requirements, in: ISARC 2016 - 33rd International Symposium on Automation and Robotics in Construction,

2016. https://doi.org/10.22260/isarc2016/0116.

[12] Y. Zou, A. Kiviniemi, S.W. Jones, A review of risk management through BIM and BIM-related technologies, Safety Science. 97 (2017) 88–98. https://doi.org/10.1016/j.ssci.2015.12.027.

[13] Y. Ji, F. Leite, Automated tower crane planning: leveraging 4-dimensional BIM and rule-based checking, Automation in Construction. 93 (2018) 78–90. https://doi.org/10.1016/j.autcon.2018.05.003.

[14] J. Olearczyk, A. Bouferguène, M. Al-Hussein, U. Hermann, Automating motion trajectory of crane-lifted loads, Automation in Construction. 45 (2014) 178–186. https://doi.org/10.1016/j.autcon.2014.06.001.

[15] H.R. Reddy, K. Varghese, Automated Path Planning for Mobile Crane Lifts, Computer-Aided Civil and Infrastructure Engineering. 17 (2002) 439–448. https://doi.org/10.1111/0885-9507.00005.

[16] P. Cai, Y. Cai, I. Chandrasekaran, J. Zheng, Parallel genetic algorithm based automatic path planning for crane lifting in complex environments, Automation in Construction. 62 (2016) 133–147. https://doi.org/10.1016/j.autcon.2015.09.007.

[17] D. Wu, Y. Sun, X. Wang, X. Wang, An improved RRT algorithm for crane path planning, International Journal of Robotics and Automation. (2016). https://doi.org/10.2316/Journal.206.2016.2.206-4180.

[18] C. Zhang, A. Hammad, Improving lifting motion planning and re-planning of cranes with consideration for safety and efficiency, Advanced Engineering Informatics. 26 (2012) 396–410.

[19] X. Wang, Y.Y. Zhang, D. Wu, S. De Gao, Collision-free path planning for mobile cranes based on ant colony algorithm, Key Engineering Materials. 467–469 (2011) 1108–1115. https://doi.org/10.4028/www.scientific.net/KEM.467-469.1108.

[20] J. An, M. Wu, J. She, T. Terano, Re-optimization strategy for truck crane lift-path planning, Automation in Construction. 90 (2018) 146–155. https://doi.org/10.1016/j.autcon.2018.02.029.

[21] J. Olearczyk, A. Bouferguène, M. Al-Hussein, U. Hermann, Automating motion trajectory of crane-lifted loads, Automation in Construction. 45 (2014) 178–186. https://doi.org/10.1016/j.autcon.2014.06.001.

[22] W.-H. Hung, C.-W. Liu, C.-J. Liang, S.-C. Kang, Strategies to accelerate the computation of erection paths for construction cranes, Automation in Construction. 62 (2016) 1–13. https://doi.org/10.1016/j.autcon.2015.10.008.

[23] Y. Fang, Y.K. Cho, F. Druso, J. Seo, Assessment of operator's situation awareness for smart operation of mobile cranes, Automation in Construction. (2018). https://doi.org/10.1016/j.autcon.2017.10.007.

[24] Y. Fang, J. Chen, Y.K. Cho, K. Kim, S. Zhang, E. Perez, Vision-based load sway monitoring to improve crane safety in blind lifts, Journal of Structural Integrity and Maintenance. (2018). https://doi.org/10.1080/24705314.2018.1531348

[25] C. Huang, C.K. Wong, C.M. Tam, Optimization of tower crane and material supply locations in a high-rise building site by mixed-integer linear programming, Automation in Construction. 20 (2011) 571–580. https://doi.org/10.1016/j.autcon.2010.11.023.

[26] H. Guo, Y. Yu, M. Skitmore, Visualization technology-based construction safety management: A review, Automation in Construction. 73 (2016) 135–144. https://doi.org/10.1016/j.autcon.2016.10.004.

[27] S. Karaman, E. Frazzoli, Sampling-based algorithms for optimal motion planning, in: International Journal of Robotics Research, 2011. https://doi.org/10.1177/0278364911406761.

[28] R. Sacks, C. Eastman, G. Lee, P. Teicholz, BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors. 2018. https://doi.org/10.1002/9781119287568.

[29] L. Peng, D.K.H. Chua, Decision Support for Mobile Crane Lifting Plan with Building Information Modelling (BIM), in: Procedia Engineering, Elsevier, 2017: pp. 563–570. https://doi.org/10.1016/j.proeng.2017.03.154.

[30] J. Wang, X. Zhang, W. Shou, X. Wang, B. Xu, M.J. Kim, P. Wu, A BIM-based approach for automated tower crane layout planning, Automation in Construction. 59 (2015) 168–178. https://doi.org/10.1016/j.autcon.2015.05.006.