APPLICATION OF ROBOTIC OBSTACLE AVOIDANCE IN CRANE LIFT PATH PLANNING

Zhen Lei[1]*, Saeed Behzadipour[2], Mohamed Al-Hussein[1], Ulrich (Rick) Hermann[3]

[1] *Department of Civil & Environmental Engineering, University of Alberta, Edmonton, Canada*
[2] *Department of Mechanical Engineering, University of Alberta, Edmonton, Canada*
[3] *PCL Industrial Constructors Inc., Edmonton, Canada*
* Corresponding author (zlei@ualberta.ca)

ABSTRACT: Crane lift path planning aims to find a collision-free trajectory for the lifted object among on-site obstacles from its pick location to the final location. However, the current manual planning process is time-consuming, prone to errors, and requires the practitioners to have exceptional visual abilities, since the construction site is congested and dynamically changing. Therefore, the need for developing a decision support system, to automate the crane path planning process, is quite eminent. This paper presents a methodology based on robotic motion planning to numerically solve the crane path planning problem. The proposed methodology conducts 2D path planning for a crane lift operation and accounts for the rotation of the lifted object during its movements. The proposed methodology has been implemented into a computer module, which provides a user-friendly interface to aid the practitioners to perform a collision-free path planning, and check the feasibility of the path at different stages of the project. Two examples are described in order to demonstrate the effectiveness of the proposed methodology and illustrate the essential features of the developed module. This research project is supported by PCL Industrial Constructors Inc.

Keywords: Crane lift; Path planning; Robotics motion planning; Automation.

1. INTRODUCTION

The efficiency of construction has been improved dramatically in recent decades due to the use of cranes. Heavy lift planning maintains a central role in the success of crane operations, and errors in this process can result in extra cost and schedule delay. Meanwhile, the heavy lift planning is usually carried out in the pre-construction phase with limited data available, which makes the planning more challenging. The current heavy lift planning methods rely mostly on engineers' intuitions and experience. In order to improve the efficiency and accuracy of heavy lift planning, efforts have been made by engineers and researches. Many have focused on crane selection; for instance, Al-Hussein [1] proposed a methodology for crane selection location and on-site utilization for construction projects, and also an optimization algorithm for selection and location of mobile cranes on construction sites [2]. A fuzzy logic

approach was applied to crane type selection [3]. Also, a crane selection tool, IntelliCranes, has been developed based on probabilistic neural networks [4]. Tam et al. [5] and Tam and Tong [6] applied a genetic algorithm (GA) to analyze the operations of tower cranes. In addition, Manrique et al. [7] described a methodology used to integrate crane selection algorithms and an optimization model with 3D modeling and animation for the selection, utilization, and location of cranes. A database management system was developed to assist practitioners to manage crane-related data by Al-Hussein et al. [8]. Hasan et al. [9] presented an algorithm for analyzing the mobile crane supporting system.

Heavy lift path planning aims to arrange a moving path for the lifted object, avoiding the collision with the on-site obstacles. Due to the dynamic nature and congested space of the construction site, the current manual planning process for the heavy lift path planning is

inefficient and prone to errors. Particularly, industrial construction projects involve frequent heavy lift operations using mobile cranes, which require efficient and frequent path planning. Therefore, there is demand to develop a decision support system to automatically generate the crane path planning. Researchers like Ali et al. [10] solved cooperative crane lifts path problem using different search path methods. Reddy et al. [11] developed a system for automated path planning for single mobile crane lifts using the AutoCAD environment and AutoLisp. A virtual crane model has been built for visualization of erection processes and erection schedule by using robotic motion planning [12].

In this paper, a robotic approach is proposed to solve crane lift path planning problem. The proposed methodology conducts 2D path planning for a crane lift operation and accounts for the rotation of the lifted object during its movements. The designed system integrates a database and utilizes VB.Net programming environment in order to develop a prototype module, providing a user-friendly interface to aid the practitioners during the path planning and collision checking.

2. ROBOTIC MOTION PLANNING

In robotics, the motion planning problem consists of path planning and trajectory planning. Path planning is the process in which a collision free path is developed between the starting and ending configurations of the robot; Trajectory planning deals with the timing of the motion along the designed path. Therefore, the robotic motion planning is to ensure that the movements of the robot(s) are safe, efficient, and with respectively short moving distance. The problem of crane lifting planning is very similar to path planning problem in robotics. The crane has the role of the robotic manipulator and the lifted object can be considered as the payload of the robot. Lozano-Pérez's [13] proposed a method called obstacle growth by using Minkowski point-set operations to solve 2D the motion planning problem. The core concept of this method is to shrink the shape of the robot into one single representing point, and grow the obstacles to configuration space obstacles (C-Obstacles). Instead of

arranging the trajectory considering the entire shape of the robot, the path planning problem is simplified to search the path for the chosen representing point. After generating the C-Obstacles, there exist various methods to find the path for robots. Latombe [14] reviewed the common methods: roadmap, cell decomposition, and potential field. Fig.1 shows an example of creating C-Obstacle based on a triangular lifted object without considering the rotations. The top vertex of the triangle A is chosen as the representing point, and the obstacle B is grown by the shape of triangle A, generating a new area (C-Obstacle). After the generation of the C-Obstacle, the problem of finding a path for A avoiding collision with B is equivalent to finding a path for the representing point avoiding the generated C-Obstacle.
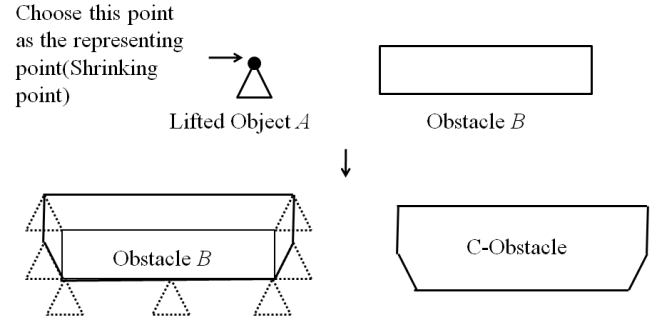
Fig. 1 Obstacle Growth Approach

The shapes of the generated C-Obstacles change as the lifted object rotates. So it is ideal to create the C-Obstacles while considering the rotations of the lifted object in a continuous manner. As shown in Fig. 2 (a), for every possible rotation of the lifted object, the C-Obstacles are created and consequently a volume is obtained. However, analytical creation and handling of such volumes, seems unfeasible. Therefore, to obtain a computationally feasible solution, it is assumed that a finite rotation step is used and the volume is presented by a finite number of layers, as shown in Fig.2 (b). In this case, for each rotation of the lifted object, the C-Obstacles are generated and laid on one layer, and layers corresponding to various rotations of the lifted object are generated (Fig.2 b and c). The connections for the nodes on the same layer or between layers are checked and

stored in a connection network (as shown in Fig. 3), and based on that, the optimal path for the lifted object can be searched.
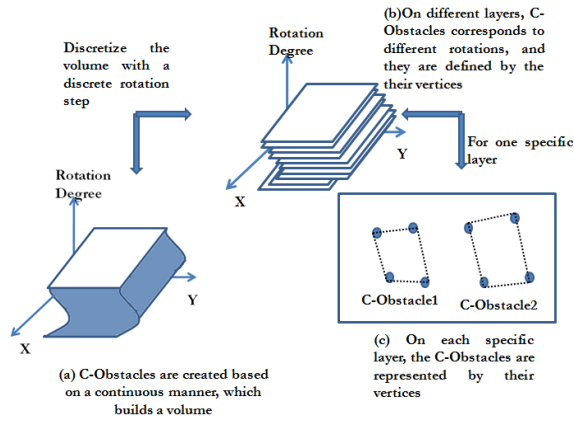

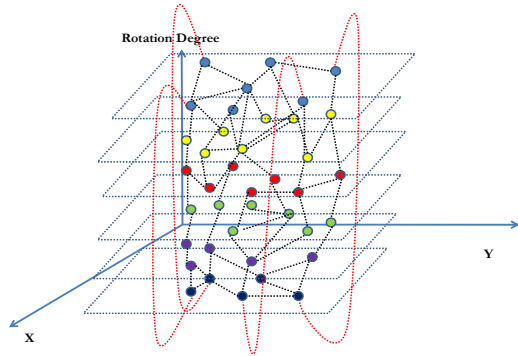Fig. 2 Generation of C-Obstacles for Different Rotations

Fig. 3 Connection Network

3. DESIGNED METHODOLOGY AND ALGORITHMS

This research combined the robotic motion planning with the current industrial construction crane practice, and the system is designed in VB.Net. The robotic motion planning is the main topic of this paper.

C-Obstacle Creation: The core idea of creating the C-Obstacles is to shrink the lifted object into a representing point and grow the obstacles by the shape of the lifted object. Also, different C-Obstacles corresponding to different rotations of the lifted object are presented on various layers based on a discrete rotation step. Fig .4 gives one example of the algorithm for generating a C-Obstacle for a lifted object. In step one, node 1 from the lifted object is chosen as a representing point, and two vectors starting from other vertices of the lifted object toward the representing point are created; In second step,

two generated vectors are added to each vertex of the obstacles; In third step, a boundary (shown as the dash lines) is created to embrace all the generated nodes. If the representing point (node 1) moves outside the C-Obstacle ("A" in Fig. 4) or on the edge of the C-Obstacle ("B" in Fig. 4), it indicates that the lifted object is traveling without collision with the obstacle or just touching the obstacle. "C" in Fig. 4 represents a collision when the representing point steps into the C-Obstacle. However, the shapes of the C-Obstacles also vary as the lifted object rotates. The key to build the C-Obstacles with different rotations is to find the geometrical shapes of the lifted object according to different rotations. The algorithm to calculate the geometrical shape of the lifted object for different rotations is based on a 2D rotation matrix.

Connection Checking: A connection between two nodes simply implies that a motion between the two positions represented by the nodes is feasible. For instance, a connection between two nodes from the same layer implies one can get from one node to the other using a straight line translation; a connection between two nodes from two different layers represents the rotation of the lifted object, and a translation if these two nodes have different x and y coordinates. Note that, the initial and final orientations of the lifted object do not cause collision with obstacles since they are obstacle free configurations on their corresponding layer.

The connections between nodes on the same layer are detected by a visibility graph [15]. In a visibility graph, any two nodes from the same layer can be connected if they do not collide with the existing C-Obstacles. The connection for two nodes from successive layers is checked to guarantee the movement between these two layers does not collide with the existing obstacle. All the calculated connections are stored in an adjacency matrix.

Path Finding: In this research, Dijkstra's algorithm is applied to find the shortest path for the crane lift. For this purpose one needs to define a metric for the length of path. In this work, we used the physical length of motion. Other metrics such as energy and time can be also used. The basic idea of Dijkstra's algorithm is to calculate the

minimum distance from start point to each other nodes and store them in a distance array. Then the shortest path is searched based on the array.

4. EXAMPLES

Example One: As illustrated in Fig. 5, example one provides a scenario with three obstacles. Eight steps are shown to depict the optimal path.

Example Two: Example two presents a congested scenario with four obstacles (Fig. 6). The found path proves that the designed system works for a quite congested layout in which the lifted object has to be moved through a narrow trajectory. Also, the developed

algorithm successfully calculates the path with relatively shorter moving distance
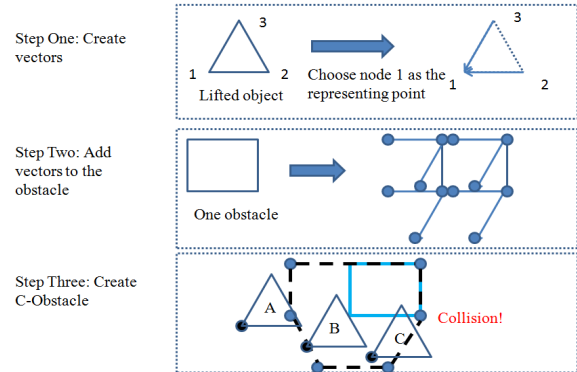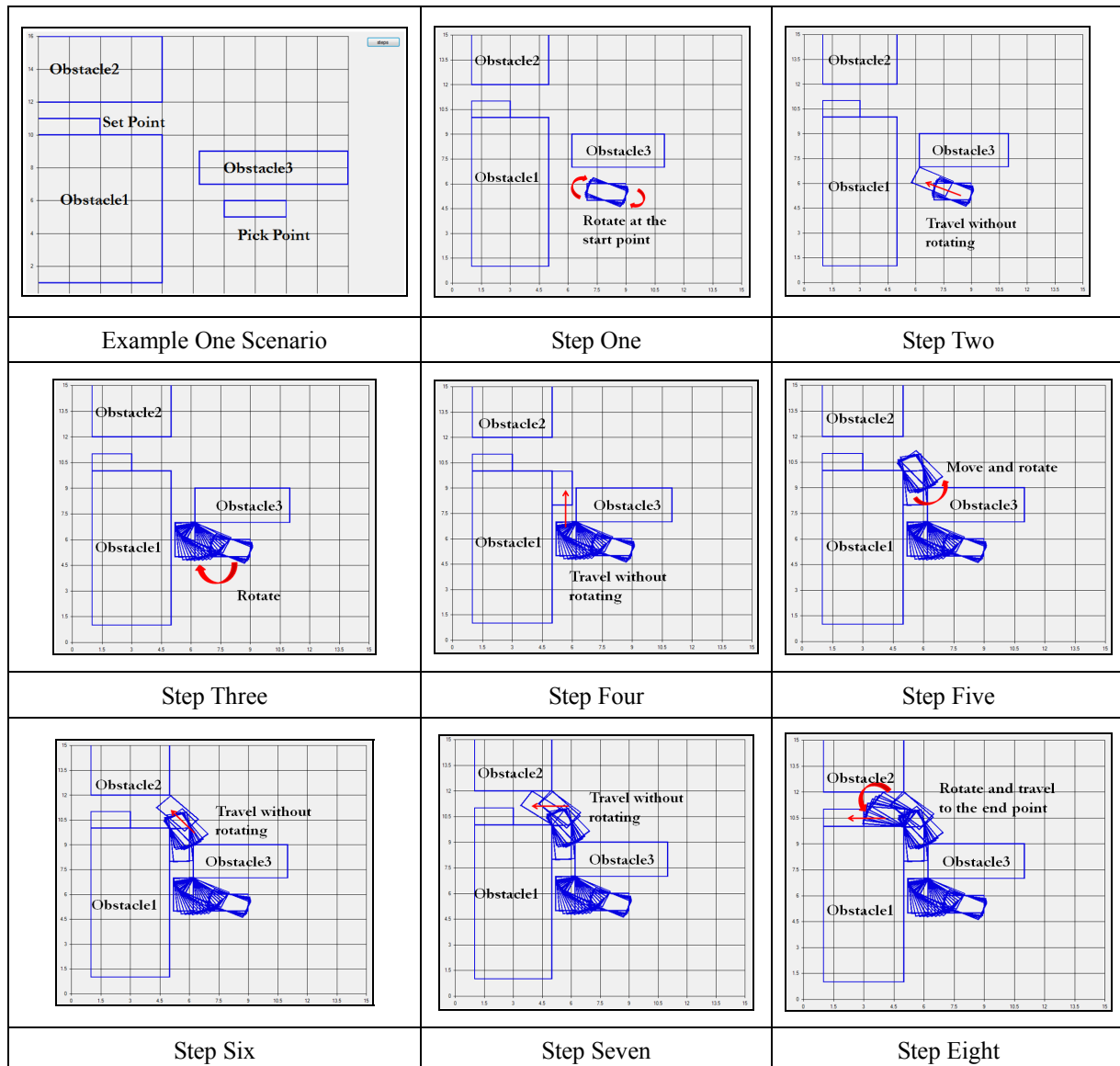

Fig. 4 C-Obstacle Creation with Fixed Rotation

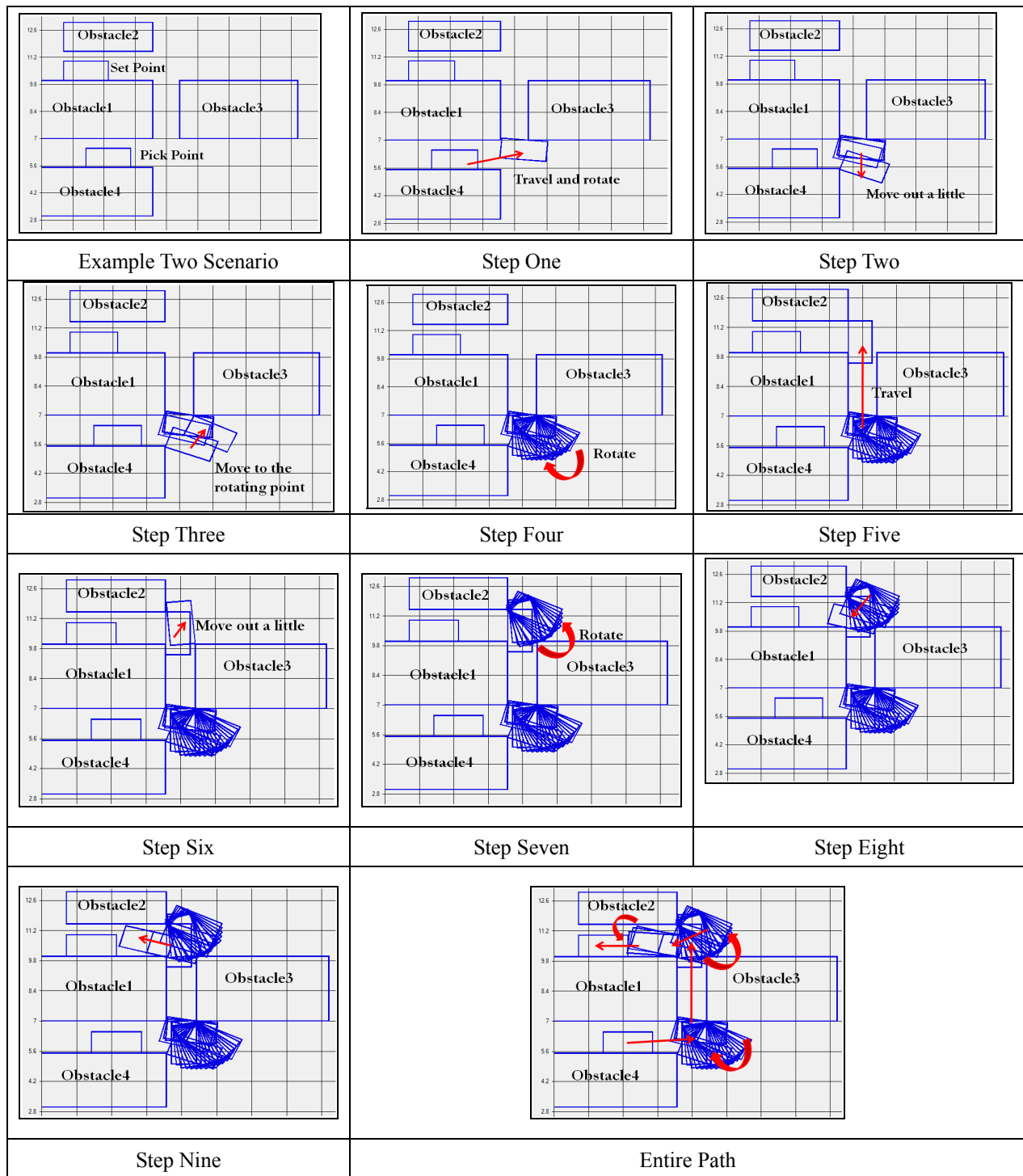| Example One Scenario | Step One | Step Two |
| Step Three | Step Four | Step Five |
| Step Six | Step Seven | Step Eight |

Fig. 5 Example One Solution Path

Fig. 6 Example Two Solution Path

5. CONCLUSIONS

This research is motivated by the need to develop a decision support system for the user in order to automate heavy lift path planning since the current path planning process is time-consuming and prone to errors. The developed system in this research can potentially benefit the current crane heavy lift planning in many aspects. The developed system automates the process of heavy lift path planning. However, the developed system considers only the 2D planar layout of the construction site. This research project will be further extended into a 3D space environment in which more possibilities of path planning

can be tested. With the current system, the user can easily obtain a yes-or-no answer for one specific lift operation path planning. The future research will put efforts to expand the 2D system to a 3D path planning system. In addition, since the designed system laid the foundation of generating a collision-free path for a moving object, the system can be used for other crane-involved applications.

5. ACKNOWLEDGEMENTS

REFERENCES

[1]Al-Hussein, M. (1999). "An integrated system for crane selection and utilization." Ph.D. thesis, Department of Building Civil & Environmental Engineering, Concordia University, Montreal, Canada.

[2]Al-Hussein, M., Alkass, S., and Moselhi, O. (2005). "Optimization algorithm for selection and on-site location of mobile cranes." *Journal of Construction Engineering and Management*, ASCE, 131(5), 579-590.

[3]Hanna, A. S. and Lotfallah, W. B. (1999). "A fuzzy logic approach to the selection of cranes." *Automation in Construction*, 8(5), 597-608.

[4]Sawhney, A. and Mund, A. (2002). "Adaptive Probabilistic Neural Network-based Crane Type Selection System." *Journal of Construction Engineering and Management*, ASCE, 128(3), 265-273.

[5]Tam, C.M., Tong, K.L., and Chan, K.W. (2001) "Genetic algorithm for optimizing supply location around tower crane." *Journal of Construction Engineering and Management*, 127 (4), 315– 321.

[6]Tam, C.M. and Tong, T.K.L. (2003). "GA-ANN Model for Optimizing the Location of Tower Crane and Supply Points for High-Rise Public Housing Construction." *Construction Management and Economics*, 21, 257-266.

[7]Manrique, J.D., Al-Hussein, M., Telyas, A., and Funston, G. (2007). "Constructing a complex precast tilt-up-panel structure utilizing an optimization model, 3D CAD and animation." *Journal of Construction Engineering and Management*, ASCE, 133(3), 199-207.

[8]Al-Hussein, M., Alkass, S., and Moselhi, O. (2000). "D-CRANE: Database system for utilization of cranes." *Canadian Journal of Civil Engineering*, 27, 1130-1138.

[9]Hasan, S., Al-Hussein, M., Hermann, U. H., and Safouhi, H. (2010) "Interactive and dynamic integrated module for mobile cranes supporting system design." *Journal of Construction Engineering and Management*, ASCE, 136(2), 179-186.

[10]Ali, M. S., Babu, N. R., and Varghese, K., (2005). "Collision free path planning of cooperative crane manipulators using genetic algorithm." *Journal of Computing in Civil Engineering*, 19 (2), 182–193.

[11]Reddy, H. R., Varghese, K., (2002). "Automated path planning for mobile crane lifts." *Computer-Aided Civil and Infrastructure Engineering*, 17 (6), 439–448.

[12]Kang, S. C., and Miranda, E., (2006). "Planning and visualization for auto-mated robotic crane erection processes in construction". *Automation in Construction*, 15 (4), 398–414.

[13]Lozano-Pérez, T. (1983). "Spatial planning: A configuration space approach." *IEEE Transactions on Computers*, 32(2), 108-119.

[14]Latombe, J.C. (1991). "Robot motion planning." Kluwer Academic Publishers, Norwell, MA.

[15] Asano, T., Guibas, L., Hershberger, J., and Imai, H. (1985). "Visibility-polygon search and Euclidean shortest path." *The 26th Symposium on Foundations of Computer Science*, Portland, Oreg., October 21-23, 155-164.