Proceedings of the 2007 Winter Simulation Conference
S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, eds.

AGENT-BASED SIMULATION FOR COLLABORATIVE CRANES

Cheng Zhang

Building, Civil, & Environmental Engineering Dep.
1455 De Maisonneuve Blvd. W., Concordia University
Montreal, QC, CANADA

Amin Hammad

Concordia Institute of Information Systems Engineering
1455 De Maisonneuve Blvd. W., Concordia University
Montreal, QC, CANADA

ABSTRACT

Most of the materials and components for a construction project are lifted by cranes. Much research has been done for path planning, motion control, and online navigation to reduce conflicts and improve efficiency. This paper proposes a new simulation approach based on agents to coordinate crane operations where two cranes are working together. Agents can dynamically control the kinematic action of the two cranes respecting the functional constraints for safety and efficiency of operations. A simulation model is under development using a case study to investigate the feasibility of the proposed approach.

1 INTRODUCTION

Most of the materials and components in a construction project are lifted by cranes. It is estimated that one crane upset occurs during every 10,000 hours of crane use. Approximately 3% of upsets result in death, 8% in lost time, and 20% in damage to property other than the crane. Nearly 80% of these upsets can be attributed to predictable human error when the operator inadvertently exceeds the crane's lifting capacity (Davis and Sutton 2003).

Development in simulation software and visualization is making it possible to visualize simulation results (Kamat and Martinez 2001) and train equipment operators using virtual reality (Simlog 2007). *Cranimation* (2007) is a crane selection software, which calculates the outrigger forces for mobile cranes, the distribution of ground pressures for crawling cranes, and the minimum and maximum radius ranges. *LiftPlanner* (2007) is a 3D crane and rigging planning software, which produces drawings to plan and document critical lifts. The advantage of visualizing the work is that the user can simulate and check the functional constraints and interferences that may happen in reality between the 3D physical elements and virtual workspaces. However, these simulation tools focus on equipment working individually rather than coordinating the work of several cranes, such as the case of two cranes working together to lift a heavy or large object. Furthermore, they provide more off-line planning rather than real-time control of the movement. The complexity of coordinating equipment requires more detailed planning and better real-time control of the work.

Ali et al. (2005) have proposed a path planning approach using a Genetic Algorithm (GA) for automating the path planning of two cooperative construction manipulators. Artificial intelligence (AI) research aiming at the creation of unmanned construction systems, capable of performing complex tasks as well as human operators, has been carried out to control construction work on hazardous sites or for space and underwater constructions. These systems have been applied to perform emergency countermeasure and restoration work at disaster sites (Ban 2002). It is mentioned that the efficiency of unmanned construction is roughly 60% to 70% of that of manned construction, but sharply decreases in cases where the machinery moves or high precision work is necessary (Ban 2002). For example, collaborative equipment work is a common case in construction where communication and negotiation are essential to properly accomplish the work.

Some research involving AI has been done to enhance communication between team workers and resolve problems in the construction industry. The concept of agents in AI refers to relatively independent and autonomous entities, which operate within communities in accordance with complex modes of cooperation, conflict resolution, and competition in order to survive and perpetuate themselves (Russell and Norvig 2003). Using agents to plan and coordinate construction activities can simulate the manoeuvring of the equipment and enhance communication to reduce conflicts and improve efficiency. Collaborative agents emphasize autonomy and cooperation with other agents in order to perform tasks in open and time-constrained multi-agent environments (Nwana and Ndumu 1998). Agent systems have been used for construction claims negotiation (Ren 2002) and dynamic rescheduling negotiation between subcontractors (Kim et al. 2003). However, little research has focused on real-time control for construction equipment operation us-

1-4244-1306-0/07/$25.00 ©2007 IEEE

ing agents. Furthermore, activities may need to be carried out in a multi-equipment environment to achieve a specific goal, such as two cranes working together to lift heavy or big objects. Multiple agents can be used to simulate such type of collaborative work. The distributed organization is able to adapt more easily to unforeseen modifications in the environment and, in particular, to possible malfunctions of certain agents (Ferber 1999). In some cases, re-planning is needed to avoid obstacles that were not considered in the original plan. Once this happens, communication between operators is essential to exchange information and generate new actions based on the individual agent's knowledge. Multi-agents are able to interactively simulate such kind of re-planning based on negotiation.

In the present paper, a new approach based on collaborative agents is proposed to coordinate construction equipment operation. A multi-agent framework is developed to simulate the working environment of two cranes on site. Agents can dynamically control the actions of the two cranes respecting the functional constraints for safety and efficiency of operations. A simulation model is under development using a case study to investigate the feasibility of the proposed approach. Compared with the previous research for cranes path planning, this method enhances the real-time control based on the perception of the environment.

2 FRAMEWORK

2.1 Framework Structure

Figure 1 shows the framework of an agent-based system for collaborative cranes. Many agents are involved to plan the path and guide the two cranes to finish the task on the worksite. Agents have separate but interdependent tasks to meet their final objective and to carry their work. Every agent has basic functionalities of sending messages, receiving messages, and decision making. Three major agents are involved: *Crane Agent-A*, *Crane Agent-B*, and *Site State Agent*.

Coordination-by-planning technique is the most traditional approach in AI, which is based on breaking the work down into two phases. In the first phase (planning), a set of plans are produced including a set of actions to be carried out by agents to achieve a goal. The selection of a correct representation of actions becomes even more crucial than in the planning for a single agent (Ferber, 1999). During the planning phase, only the two crane agents are involved. Based on the information from the KB and the goal description, the two crane agents negotiate through messages. One crane agent is given the priority to start proposing actions, while the other agent can simply *accept* or *reject* the proposed action from the other agent to achieve a shared plan through these messages. Genetic algorithm is used to find the near optimal plan for individual crane agents.

In the second phase (executing), the shared plan is executed. Due to the dynamic environment, re-planning may be needed, which requires real-time information updating and searching new path based on this information. Therefore, a Site State Agent is added to monitor on site situation.

Detailed description of agents are shown in the following sub-sections.

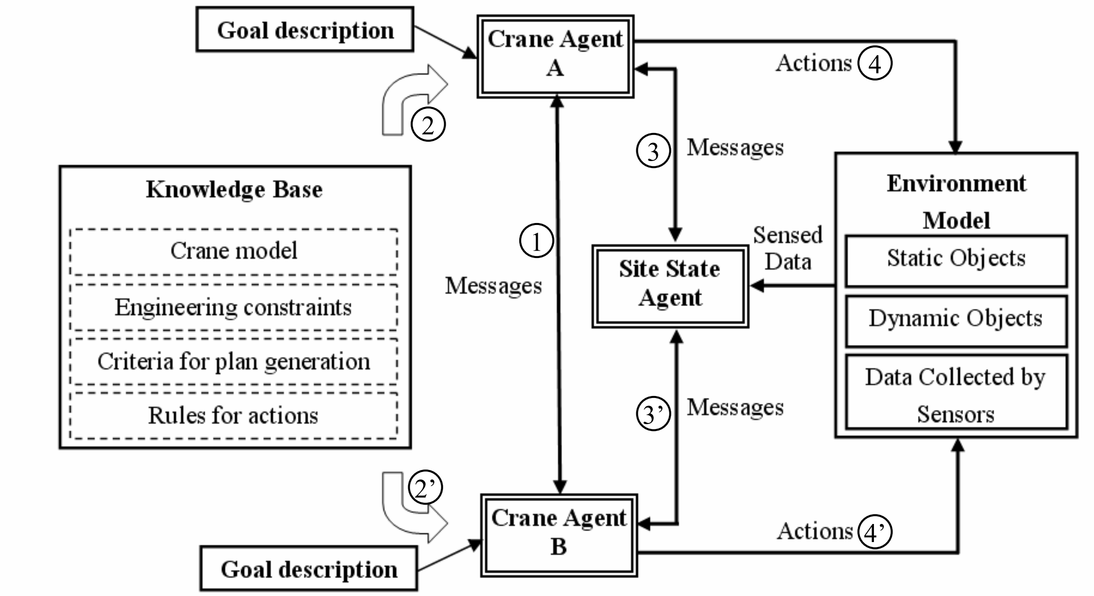

Figure 1: Framework of an Agent-based System

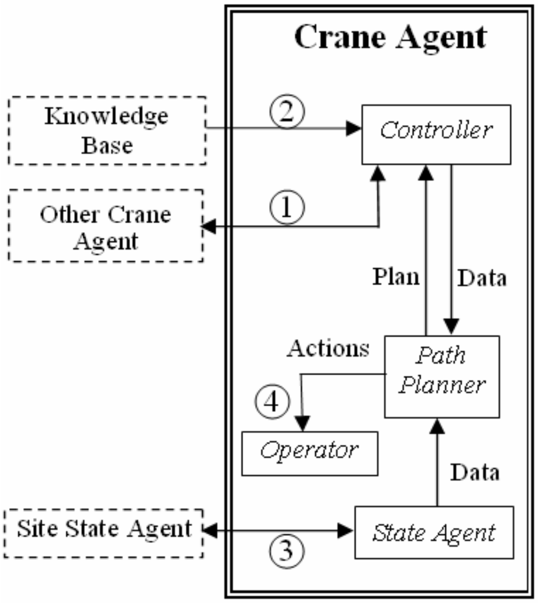

Figure 2: Crane Agent Hierarchy

2.1.1 Crane Agents

A *Crane Agent* has several low-level agents working together to communicate, plan the path, and monitor the position of the crane.

As shown in Figure 2, the *Controller* is responsible for getting information from the knowledge base and the other crane agent and extracting and transferring data to the *Path Planner*. The information obtained from the other crane agent shows the states of the other crane. The *Path Planner* will read the goal description and search the possible path according to predefined rules to avoid the obstacles and meet the constraints imposed either from the engineering or safety aspects. The path decided by the *Path Planner* is sent to the *Controller* to check with the other *Crane Agent* and revise the plan if necessary. After a final path is decided, the *Path Planner* sends the actions to the *Operator*, which transfers the actions to lower level actions that can be understood by the crane operator on the construction site or to automatically operate the crane.

The *State Agent* is responsible for getting information from the *Site State Agent* about the crane components' position, other parameters necessary to describe the states of the crane, and the site situation. It transfers these data to the *Path Planner*, to check whether a re-planning is needed due to the dynamic environment.

The *path planner* is encoded in a genetic algorithm to search the path. Rules and other information are transferred from the *Controller*.

2.1.2 Site State Agent

The *Site State Agent* is responsible for collecting information from the crane agents and the work site environment model, including both static and dynamic information. Static information includes geometric, kinematic, and static information of all system components. Dynamic information includes current component configuration and any dynamic state information (Fraile et al. 1999).

Most of the previous research has been focusing on path planning with assumptions of the site containing static obstructions (Sivakumar et al. 2003). The present work tries to enhance the communication by collecting information from the dynamic world. Therefore, knowing the position of each part of the boom and detecting any obstacle on the moving path is essential to ensure that the work is done properly while meeting the kinematics and engineering requirements.

Different sensors can be used to either report the position of the crane (DiaLog Visu 2007) or to detect the collisions in real time (Bosche et al. 2006). As shown in Figure 1, the sensed data are collected in real time and sent to the Site State Agent. These data will provide the current position of the crane components, e.g., the boom length, the boom angle, and the hook location, which will be compared with the calculated states to mitigate conflicts. Another important information can be collected by sensing the environment of the crane for collision detection. This information will be sent to the Site State Agent to check the feasibility of the next movement of the crane.

2.1.3 Knowledge Base

The *Knowledge Base* includes four parts: crane model, engineering constraints, criteria for plan generation, and the rules for actions.

The crane model has the kinematic constraints for the selected cranes. A loaded crane has a maximum of eight degrees of freedom (DoFs) (Reddy and Varghese 2002), and path planning for manipulators having more than four DoFs is considered to be complex (Hwang and Ahuja 1992). As mentioned by Reddy and Varghese (2002), there can be multiple solutions to configure the DoFs of the manipulator for a particular location of the end-effector (i.e., the hook); therefore, simplifying the representation and avoiding the inverse kinematics should be considered. The scope of the present work is limited to four DoFs, as shown in Figure 3.

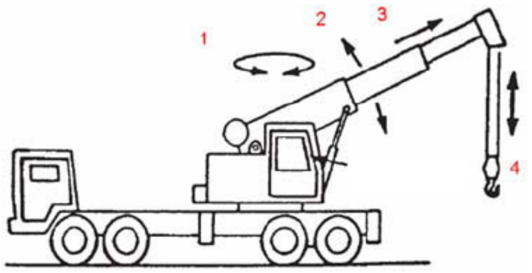

Figure 3: The DoFs of a Crane

The engineering constraints of cranes are mainly from the working range and the load charts. The working range shows the minimum and maximum boom angle according to the length of the boom and the counterweight. Load charts give the lifting capacity based on the boom length, boom angel to the ground and the counterweight.

Three major criteria should be taken into account: Lift path clearances, capacity during lift, and ground support during lift. Rules should be developed to represent these constraints which are stored in a database. One important rule is that the distance between two hooks should be equal to the length of the object, and crane load lines must be kept plumb at all times for multiple crane lift (Shapiro et al. 2000). Other rules include avoiding combinations of hoisting and swinging or hoisting and luffing at the same time; and avoiding motion when a crane is traveling. Additional rules have to be developed to decide the priority of the movement of each crane and reduce the steps taken to achieve the goal.

2.2 Collaboration of Agents

As introduced in Section 2.1, coordination of agents is based on breaking the work down into planning and executing phases. When several agents are working together, it is necessary to define the relationships between their actions to improve the coordination of these actions and reduce conflicts. Communication between agents is essential for coordinating the behaviors of the agents in time and space, which basically requires exchanging messages between agents. KQML (Knowledge Query and Manipulation Language) (Finin et al. 1994) provides a framework for agents to exchange information and knowledge. It defines the operations that agents may attempt on each other's knowledge bases and provides a basic architecture for agents to share knowledge and information. The messages transferred between crane agents and the *Site State Agent* follow the KQML format (Figure 1).

3 COMPUTING ASPECTS

Properly represented information is important to avoid complex computation and improve the knowledge acquisition. Parameters are defined to describe the goals, actions and the states. These facts can be used to fire specific rules defined in the Knowledge Base to generate new facts and guide further actions.

3.1 Representation of Goals

The goal of the crane operation can be simply represented by two points related to the load object: origin and destination. Origin (ob, P_o, Φ_o) represents the original position P_o and orientation Φ_o of the object ob. P_o (x_o, y_o, z_o) is given by the coordinates of the reference point of ob. Destination (ob, P_d, Φ_d) represents the destination position P_d and the orientation Φ_d of ob. Duration (t_1, t_2) represents the start time t_1 and the end time t_2 of the work.

3.2 Representation of Actions

Different movements of a crane can be decomposed into a series of actions. Taking a hydraulic crane as an example, the movement of the crane includes the following actions:
Base movement: BaseMove, BaseStop;
Boom movement: BoomRaise, BoomLower, BoomExtend, BoomRetract, BoomSwing, BoomStop;
Hook movement: HookHoist, HookLower, HookStop, HookGrip, HookRelease.

3.3 States Description

States representation is based on the actions taken before. For example, at State j:
ObjectLocation (ob_k, P_{kj}, Φ_{kj}): object ob_k is at position P_{kj} with orientation Φ_{kj};
CraneLocation ($crane_i$, P_{ij}, Φ_{ij}, θ_{ij}, α_{ij}, l_{ij}, P^{h}_{ij}): crane i is at location P_{ij}, with base orientation Φ_{ij}, boom swing angle θ_{ij}, boom angle to the ground α_{ij}, boom length l_{ij}, and hook position P^{h}_{ij};
HookGrip ($crane_i$, ob_k): the hook of $crane_i$ is gripping ob_k;
Distance ($hook_i$, $hook_{i+1}$, d_j): the distance between two hooks is d_j;

3.4 Plan Generation

Generating a plan may be seen as a state space search. Most implementations of search algorithms should be assisted by appropriate domain heuristics to find a good/optimal path within a reasonable time (Reddy and Varghese 2002). As discussed in the previous section, the kinematic motion requirements and engineering constraints are integrated to generate reasonable plans for each crane.

The whole plan can be divided into tasks which consist of sub-tasks or a set of crane actions. Three major tasks category can be defined as: pre-lift task, lifting task, and post-lift task. The pre-lift task includes the actions for positioning the cranes on site, and attaching the load to the hook; the lifting task, which is the main body of the work; and the post-lift task, which includes detaching the load and moving to another place. The lifting task combines several sub-tasks, which indicates the milestones on the moving path. These milestones can be used as the target when re-planning is needed to reduce the search time.

Each task is fulfilled by taking actions to change the states of the crane. Figure 4 shows a simple example for the movement of one crane. S_j represents different states after the actions taken.

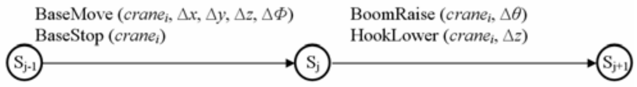
BaseMove ($crane_i$, Δx, Δy, Δz, $\Delta \Phi$)
BaseStop ($crane_i$)
BoomRaise ($crane_i$, $\Delta \theta$)
HookLower ($crane_i$, Δz)
S_{j-1}
S_j
S_{j+1}

Figure 4: Actions and States Changes

The study of Varghese et al. (1997) has shown that no industry-wide standard for heavy lift planning practices exist at present. The experts rely primarily on experience to develop the plans or to perform optimization. Furthermore, collaborative requirements also limit the possible movement of each crane, which reduces the actions that can be taken by agents. In one scenario, one crane agent is given the priority to generate the actions and the partial plans to move the object lifting half of its weight. The other crane agent can follow by taking reacting actions or reject the actions due to its own constraints. Rules are used to check the feasibility of each action. The priority of an agent may change according to specific rules. Through negotiation, an effective plan can be generated based on possible combinations of movements of cranes from one step to another.

4 CASE STUDY

The re-decking project of Jacques Cartier Bridge in Montreal is used to demonstrate the proposed collaborative agent-based system. The deck of this bridge was replaced during 2001-2002. The existing deck was removed by saw-cutting the deck into sections. Each section was removed by two telescopic cranes and a new panel was installed using the same cranes. Figure 5 shows two telescopic cranes positioned on both sides of the section to be replaced.

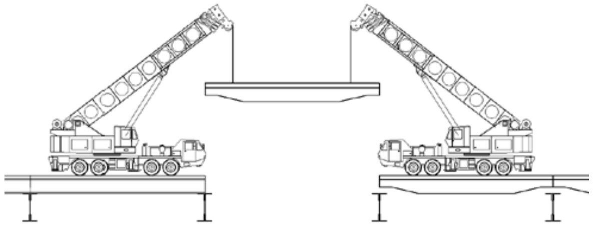

Figure 5: Two Cranes Lifting a Panel (Zaki and Mailhot 2003)

The case study focuses on the activities of removing existing deck sections and installing new panels on the main span of the bridge. Due to the low clearance of the truss structure on the main span of the bridge, the cranes had to work with a near-flat boom (α = 15°), which results in decreasing its lifting capacity by more than 50%. The contractor built a physical model of a part of the bridge and different types of cranes to check the feasibility and to plan the movement of each crane. In the real lifting situation, the operators were guided by a coordinator to ensure the safety and to guide the operations. By repeating the same tasks, the productivity was improved from 1 panel to 6 panels per night. Figure 6 shows the schematic representation of two cranes working together to lift an old section and load it to the trailer. The task shown here can be described by changing only the swing angles of the cranes (θ_1 and θ_2). One possible plan to synchronize the movement of the booms of the two cranes is to define the unit movement of the cranes as θ_1/T and θ_2/T, respectively (T is the duration of the task).

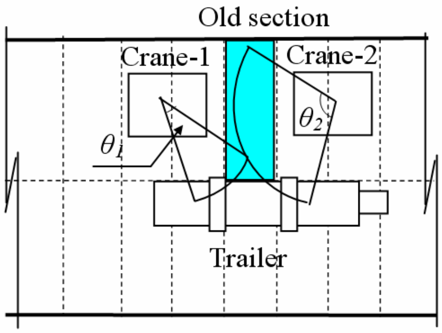

Figure 6: Swing Range for Two Cranes

A simulation model is under development using Java language to create a virtual environment showing the work site, the bridge structure and two virtual cranes. Transformation matrices are used to specify the relative location of each component of a crane with respect to its parent. Load charts and working ranges are retrieved from a database to control the cranes' movement. The rule engine used in the application is following the algorithms described in Bigus and Bigus (2001). A preliminary test is under development to facilitate the communication between two agents representing the two cranes.

5 CONCLUSIONS AND FUTURE WORK

This paper described a framework of multi-agent system for collaborative cranes on construction site. Agents are used to plan and execute the work on site by communicating with each other and making decisions for actions based on negotiation. Some computing aspects are discussed for representing the goal, actions, and states and path planning. A simulation model is under development using a case study, which is about a bridge rehabilitation project, where two cranes are working together to lift a panel with height limits from the bridge structure. The preliminary test shows the feasibility of the proposed approach. Future work will focus on further developing and testing the system.

REFERENCES

Ali, M.S.A.D., N.R. Babu, and K. Varghese. 2005. Collision free path planning of cooperative crane manipulators using genetic algorithm. In *Journal of Computing in Civil Engineering*, ASCE, 19(2):182-193.

Ban, Y. 2002. Unmanned construction system: present status and challenges. In *Proceedings of the 19th International Symposium on Automation and Robotics in Construction*, Washington, U.S. 241-246.

Bigus, J.P., and J. Bigus. 2001. *Constructing Intelligent Agents Using Java*. 2nd ed. John Wiley & Sons, Inc.

Bosche, F., J. Teizer, C.T. Haas, and C.H. Caldas. 2006. Integrating data from 3D CAD and 3D cameras for real-time modeling. In *Proceedings of Joint International Conference on Computing and Decision Making in Civil and Building Engineering*, Montreal, Canada, 37-46.

Cranimation 2007. Available via <http://www.cranimax.com> [accessed March 31, 2007].

Davis, B.R., and S.C. Sutton. 2003. A guide to crane safety. N.C. Department of Labor, Division of Occupational Safety and Health.

Dialog Visu. 2007. Available via <http://www.manitowoccranegroup.com/MCG_POT_AM/Home/EN/BrandHome.asp> [accessed March 31, 2007]

Ferber, J. 1999. *Multi-Agent System, An Introduction to Distributed Artificial Intelligence*, Addison-Wesley, Pearson Education Limited.

Finin, T., R. Fritzson, D. McKay, and R. McEntire. 1994. KQML as an agent communication language. In *Proceeding of the third International Conference on Information and Knowledge Management*. Gaithersburg, Maryland, U.S. 456-463.

Fraile, J.C., C.J.J. Paredis, C.H. Wang, and P.K. Khosla. 1999. Agent-based planning and control of a multi-manipulator assembly system, In *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, Detroit, Michigan, 1219- 1225.

Hwang, Y.K., and C.T. Hass. 1992. Gross motion planning: a survey. In *ACM Computing Surveys*, 24(3): 219-291.

Kamat, V.R., and J.C. Martinez. 2001. Visual simulated construction operations in 3D. In *Journal of Computing in Civil Engineering*, ASCE, 15(4): 329-337.

Kim, K., and Jr.B.C. Paulson. 2003. Agent-based compensatory negotiation methodology to facilitate distributed coordination of project schedule changes. In *Journal of Computing in Civil Engineering*, ASCE, 17(1): 10-18.

LiftPlanner 2007. Available via <http://www.liftplanner.net> [accessed March 31, 2007].

Nwana, H.S., and Ndumu, D.T. 1998. A brief introduction to software agent technology. *Agent Technology, Foundations, Applications, and Markets*, Springer.

Reddy, H.R., and K. Varghese. 2002. Automated path planning for mobile crane lifts. In *Computer-aided Civil and Infrastructure Engineering*, 17: 439-448.

Ren, Z., and C. Anumba. 2002. Learning in multi-agent systems: a case study of construction claims negotiation. In *Advanced Engineering Information*, 16(4): 265-275.

Russell, S., and P. Norvig. 2003. *Artificial Intelligence, A Modern Approach*, second edition, Prentice Hall.

Shapiro, H.I., J.P. Shapiro, and L.K. Shapiro. 2000. *Cranes and Derricks*, 3rd Ed., McGraw-Hill, New York.

Simlog. 2007. Available via <http://www.simlog.ca> [accessed March 31, 2007].

Sivakumar, PL., K. Varghese, and N.R. Babu. 2003. Automated path planning of cooperative crane lifts using heuristic search. In *Journal of Computing in Civil Engineering*, ASCE, 17(3): 197-207.

Zaki, A.R., and G. Mailhot. 2003. Deck reconstruction of Jacques Cartier Bridge using precast prestressed high performance concrete panels, *PCI Journal*.

AUTHOR BIOGRAPHIES

CHENG ZHANG is currently a Ph.D. student at the Department of Building, Civil and Environmental Engineering at Concordia University. Her research interests are focused on spatio-temporal issues in infrastructure management systems. Her email address is <zha_che@encs.concordia.ca>.

AMIN HAMMAD is an Associate Professor at Concordia Institute for Information Systems Engineering. His research interests are mainly focused on telegeoinformatics, infrastructure and urban management systems. His email address is <hammad@ciise.concordia.ca> and his web address is <http://www.ciise.concordia.ca/~hammad>.