

TIME-STEPPED, SIMULATION-BASED SCHEDULING SYSTEM FOR LARGE-SCALE INDUSTRIAL CONSTRUCTION PROJECTS

Di Hu
Yasser Mohamed

Hole School of Construction Engineering and Management
Department of Civil and Environmental Engineering
University of Alberta
Markin/CNRL Natural Resources Engineering Facility
Edmonton, Alberta, T6G 2W2 CANADA

ABSTRACT

Industrial construction projects have recently moved more towards tighter schedules and fast-tracked engineering and construction. This leads to overlap between work packages, increased occurrences of resource over-allocation and site congestion, which pose challenges to project planners in scheduling construction works and dynamically allocating resources considering the construction site conditions. Most previous research related to scheduling and resource allocation assumes that allocated resources remain consistent throughout the execution of work packages. This paper presents a workforce planning system that allows for variable resource allocation and variable durations in execution of a work package, while holding logic relationships and space congestion constraints. A case study from a real industrial project is presented and results show that the proposed system reduces resource idle time as well as returns shorter project duration than traditional scheduling approaches.

1 INTRODUCTION

Industrial construction is a special type of construction project, which involves building facilities such as chemical processing plants, oil and gas production facilities and petroleum refineries. Mega-size industrial projects often spring up in western Canada due to the surging global demand for petroleum products. Mega industrial projects push beyond the limits of every aspect of a normal construction project. For example, the capital cost for a mega project will be more than \$1 billion US (ECC 2007). Fast-tracking is often the strategy used to execute this type of project in order to reduce the investment payback time and the period of risk exposure. Fast-tracking forces overlap between different project stages (e.g. design, fabrication, module assembly and site construction) and also compress each project stage. The former creates a great level of uncertainty that ripples through different stages and cause disruptions, while the latter increases interference between work packages within one stage.

Interference between work packages during on-site construction can be translated into resource over-allocation and space conflict. Since work packages often overlap rather than proceed consecutively, resource conflict or space conflict can occur even during the execution of work packages. Space conflict specifically refers to work area congestion rather than the conflict between physical facility components. Thus, frequent adjustments to the resource level of work packages are necessary. On the other hand, resource availability limit is often not fixed. Due to typical remote locations, resources that are used in these mega industrial projects are usually expensive. Resource availability usually starts at a low level, gradually increases as the project proceeds, and decreases near the end. Variation in resource availability over time is called *time dependent resource availability*.

Due to these factors, the resource level assigned to work packages is variable. As a result, durations of work packages are no longer fixed. Instead, durations can be shortened or extended as the resource level increases or decreases. This poses challenges for project planners: how to incorporate the dynamic characteristics of resource allocation into the project schedule, and how to keep the dependency relationship valid. This paper presents a congestion-constrained, dynamic resource allocation scheduling framework (CDRASS) using time-stepped simulation technology. It delves into a level of detail that allows variation in either resource availability or resource allocation in every simulation time unit (could be hour, day or week). The generated schedule also complies with various constraints (e.g. dependency, imposed dates, calendar and resource availability), as well as congestion limit.

The rest of this paper is organized as follows: the next section reviews the previous literature on construction scheduling and jobsite congestion, which is followed by discussion of their limitations. The time stepped simulation algorithm is then introduced and explained in detail. Application of this CDRASS framework on a real-life industrial construction case is presented and results are compared to those achieved by main stream commercial scheduling packages (e.g. Primavera P6 Professional R8.2 and Microsoft Project 2010). Finally, conclusions and future work are discussed.

2 LITERATURE REVIEW

2.1 Construction Scheduling

Construction scheduling has been one of the major construction research areas since the late 1950's. The resource-constrained scheduling problem, in particular, has received a lot of attention from academia. Generally speaking, the resource-constrained scheduling problem consists of two sub-topics: (1) resource allocation and (2) resource leveling. The former intends to find the shortest project duration under the resource constraint while the latter seeks to smooth out the use of resources under fixed project duration. This paper focuses on the former topic.

The resource allocation problem is essentially an optimization problem and has been tackled from various approaches, e.g. mathematical programming (Karshenas and Haber 1990; Demeulemeester and Herroelen 2002), heuristic approaches (Moselhi and Lorterapong 1992; Lu and Li 2003), and metaheuristic approaches (Hegazy 1999; Christodoulou 2010; Lu et al. 2008). Each approach has advantages and disadvantages. This means that not a single approach is guaranteed to have superior performance than the others. Mathematical programming can return a global optimal solution, but is found impractical for real life scheduling problems. In contrast, heuristic approaches are easy to apply and involve less computation, but can only offer near-optimal solutions. Meta-heuristic approaches attempt to strike balance between computations and optimality. However, they also have limitations. For example, Genetic Algorithm (GA) is found to have a tendency to be trapped in local optima and to require long processing time (Elbeltagi et al. 2005).

Prior research is mostly focused on schedule optimization; however, it all implicitly contains two assumptions: (1) each work package has a predefined resource requirement (type of resources and required amounts) and it can only start when the requirement is satisfied; (2) the resource level (i.e. the amount of resource assigned) of the work package stays constant throughout its duration. For example, assume that a work package, W1, requires 10 skilled workers of trade T1. This work packages is allowed to start only when 10 or more workers from T1 are available. Otherwise, it would be delayed. However, in reality, the work package might start even with 8 workers. The resource level assigned to a work package could be any value within a range that is composed from a minimal, a normal and a maximal amount. Plus, the resource level might also change during the execution of the work package. For example, when the work area gets too congested, the resource level of the work package might be less than the normal amount. Or, the resource originally assigned to the work package in progress might be diverted to a higher priority work package whose completion is more critical to the overall project. The dynamic nature of resource allocation is largely neglected in previous scheduling research.

2.2 Jobsite Congestion

The job congestion issue has been studied in two different ways: (1) space scheduling and (2) productivity loss. The former uses space as one of the resources to perform a work package. The main objective is to detect the space conflict between work packages, e.g. clashes between physical components or one work package blocking the material handling path of another. The second, on the other hand, aims at studying the impact of work area over-crowding on crew productivity. The implicit assumption is that work packages can share the same work area and not necessarily cause the aforementioned space conflict. This paper is more related to the latter topic; only literature about work area congestion will be reviewed.

Many researchers (Ahuja and Nandakumar 1985; Dozzi and AbouRizk 1993; Thabet and Beliveau 1994) have attempted to quantify the impact of site congestion on crew productivity, but few have tried to incorporate its impact into schedules of work packages. Zouein and Tommelein (2001) argued that congestion can be alleviated by reducing the resource level of work packages that take place in that work area. However, they still assume that the resource level is pre-determined and is invariable throughout the duration of work packages. Thabet and Beliveau (1994) also recognized that work area congestion affects the execution of work packages and attempted to capture the impact in the schedules of work packages. They suggested that the productivity of affected work packages should be reduced, and thus, their durations should be extended. They also imply that work packages can keep maintain a normal level of resources, regardless of how crowded the work area is, and only be penalized by decreased production rates. However, this is not the case in practice, since congestion not only decreases the productivity, but also increases safety hazards. Therefore, the congestion should be constrained by a maximal limit.

Review of prior research on construction scheduling and jobsite congestion finds that dynamic resource allocation due to work package overlapping, site congestion and time dependent resource availability has not been fully addressed. Instead, a deterministic point of view dominates previous research. A new scheduling system is needed to capture the dynamic nature of resource allocation and accurately reflect it in the schedule.

3 TIME-STEPPED SIMULATION ALGORITHM

3.1 Time Stepped Simulation

Discrete Event Simulation (DES) has long been employed to model and study construction processes, with the main objective of improving the long-term performance of these processes. However, it is increasingly being used to plan and schedule short-term construction operations (Song and AbouRizk 2006; Mohamed et al. 2007; Hu and Mohamed 2010). DES can be categorized, based on the time advance mechanism, into two groups: (1) event-driven and (2) time stepped (Figure 1). Event driven simulation only models the start and the finish of work packages. It implies that the execution of work packages is determined and will change during their performance. Time stepped simulation advances the simulation in equal steps (time units). For each time step, it checks for scheduled events. If one is scheduled, the system state is updated and time advances to the next time step; otherwise, the system state remains the same but the simulation time is still advanced. This procedure repeats until all events have been executed or the predefined time limit has been reached.

Most previous simulation-based scheduling systems adopt an event-driven type of DES. However, time stepped simulation is preferred in this research because time stepped simulation slices the execution of work packages into segments by each time unit (i.e. users can define time unit as hour, day or week, etc.). Time stepped simulation is able to capture any changes occurring in these time units as events, and to timely reflect the impact of the change/event on the execution of work packages.

In the CDRASS framework, simulation is designed to advance time in a 24/7 manner (i.e. 24 hours per day and 7 days per week). Each simulation time step represents an hour in reality. This is to offer a common foundation to incorporate different calendars. For example, one work package uses the 10/5

calendar (i.e. 10 hours per day and 5 days per week) while another work package follows the 8/6 calendar (i.e. 8 hours per day and 6 days per week).

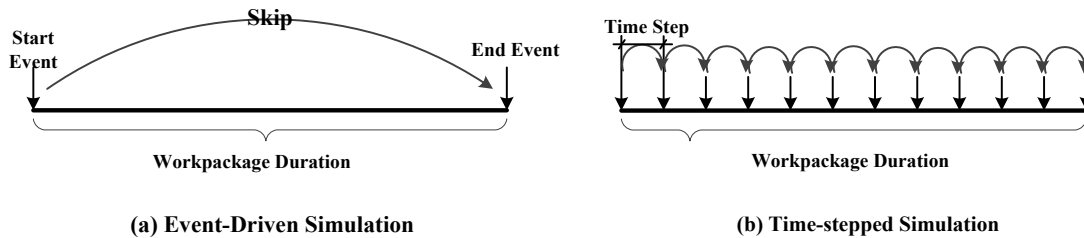


Figure 1: Event driven simulation and time stepped simulation

3.2 Simulation Algorithm

Time stepped simulation is designed to go through a routine procedure (Figure 2) in each time unit. Four major steps include: (1) identify eligible work packages, (2) allocate resources to work packages, (3) advance the simulation time, and (4) update the progress of work packages. Each step is explained as follows.

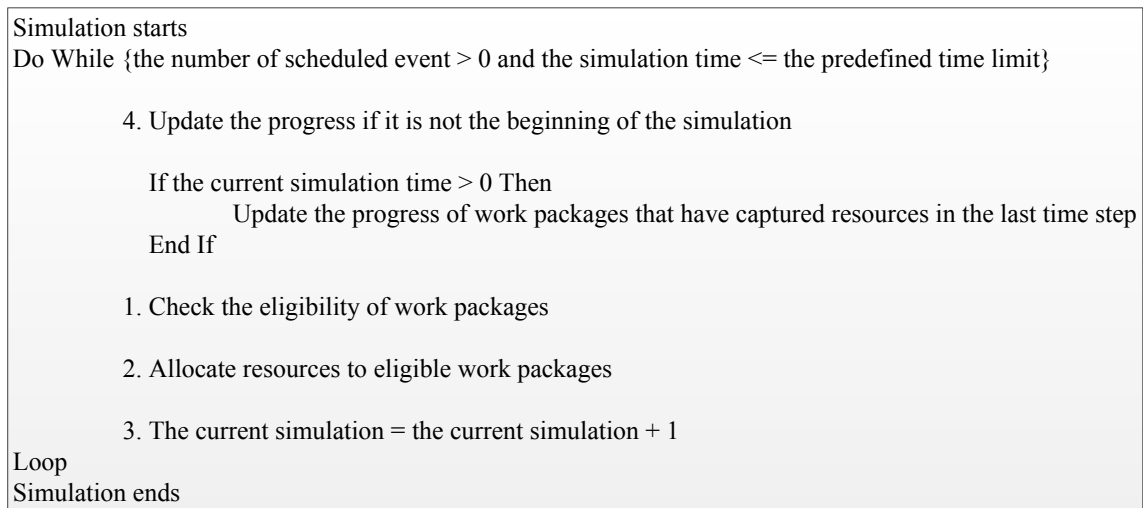


Figure 2: Time stepped simulation algorithm

3.2.1 Step 1— Check the Eligibility of Work Packages

Work packages need to be checked against various constraints before they can be scheduled. First, the simulation checks precedence relationship(s) of every work package in the work package pool. Normally, four types of precedence relationship could exist between work packages: (1) start-to-start (SS); (2) start-to-finish (SF); (3) finish-to-start (FS); and (4) finish-to-finish (FF). Both positive and negative time lags could be used to complement the four relationship types. The progress of predecessor work packages and the current simulation time determine if a precedence relationship has been met or not. For a work package to start, all its precedence relationships should be satisfied. The simulation continues to check time constraints. A work package could have certain time constraints that control its start and/or finish time. The simulation converts the current simulation time into calendar date to determine the eligibility of work packages. Finally, the simulation also checks the availability of prerequisites of a work package.

The start of the work package should match with the delivery dates of required drawings and material. Work packages that pass all three checks are schedulable, while others are sent back to the work package pool and will be checked again in the next time step.

3.2.2 Step 2 — Dynamic Resource Allocation Algorithm

A two-round resource allocation algorithm is designed to allow for dynamic resource allocation while maintaining the continuity of work packages. It should be noted that resources here include both skilled workers and congestion limit. The congestion limit can be treated as a type of resource that is expressed as how many workers can be present in the same work area. In this way, the resources allocated to a work package are constrained by both resource limit and congestion limit.

First, all eligible WAWPs (work area work packages) are divided into two categories: (1) in-progress work packages, and (2) newly-included work packages. The first round of resource allocation only focuses on in-progress work packages. It allocates resources to each work package, up to its minimal resource requirements. The second round of resource allocation combines all work packages together and sorts them according to their priority (i.e. calculated based on particular heuristics such as the least float or the least current float). It attempts to fulfill the resource requirements of higher priority work packages. The resource allocation is successful when the total amount of resources obtained from both rounds is more than the minimal resource level. Otherwise, the work package will be delayed. The maximal amount of resources that can be allocated to a work package is the minimum of the resource availability and the number of people that can be present in the work area. The actual resource allocation could be any value between the minimal resource level and the maximal resource level.

The first round of resource allocation guarantees that all in-progress work packages can be performed continuously. However, it does not mean that their resource levels will stay the same. For example, if a higher priority work package that requires the same resource is introduced to the eligible list and successfully captures resources in the second round of resource allocation, the resource level of the in-progress work package might be reduced. Likewise, a higher priority work package that occurs in the same work area might also cause the in-progress work package to capture fewer resources. The resource level can also be increased if these situations are inversed (i.e. the completion of a higher priority work package).

If the work quantity of a work package is fixed, its duration is bound to change as the resource level changes. As such, the original duration of a work package is merely an estimate, so the actual duration may differ. Variable durations pose a challenge to maintenance of the precedence relationships between work packages, especially those related to the finish of the predecessor work package (e.g. FS and FF). For example, assume that the resource level of the predecessor has been reduced and its estimated finish time has been extended. In this case, the resource levels of all its successor work packages should be decreased accordingly. This adjustment mechanism is built in the dynamic resource allocation algorithm.

3.2.3 Step 3 — Advance the Simulation Time

After the resource allocation has been determined, the simulation advances the simulation time to the next time step.

3.2.4 Step 4 — Update the Progress of Work Packages

Only those work packages that have successfully captured resources in the step 2 are to be updated. The progress that a work package gains in the previous time step is equal to the amount of resource multiplied by the time unit. For example, if a work package manages to capture 8 workers and each time unit is an hour, then 8 man-hours are obtained. If the time unit is one day and the calendar is 5/10, then 80 man-hours are collected. Once a work package fulfills the total man-hour requirement, it is deleted from the eligible work package list and transferred to the completed work package list.

The current moment is both the end of previous time step and the beginning of the current time step. A number of preparations should be done before step 1 can be re-started. First, all the resources that have been captured by work packages should be released to replenish the resource pool. Second, the availability of resources can be updated according to the pre-defined time-dependent values.

3.3 Simulation Run Efficiency

It is rare for any work package to be performed 24 hours per day and 7 days per week, although the simulation is run in this manner. There must be certain time steps that are considered non-working time for all work packages. A check is then built which functions at the beginning of every time step to see if it is non-working time. If it is, then step 1, 2 and 4 will be skipped. Only the simulation time will be advanced to the next time step. This reduces the computation time required by the simulation and hence increases its efficiency.

4 IMPLEMENTATION

The proposed time stepped simulation algorithm is implemented using Visual Basic.Net and Symphony.Net 4.0. Symphony.Net is a discrete-event simulation environment for construction projects that facilitates graphical, hierarchical and modular modeling (AbouRizk and Mohamed 2000). It has evolved through several versions concurrent with upgrades of the Microsoft.NET framework. The most recent version is Symphony.Net 4.0, which is based on the .NET 4.0 framework. Visual Studio 2010 is the main programming environment for Symphony.Net 4.0 and Microsoft Access 2007 is used as the main data base management system (DBMS).

5 CASE STUDY

The case study is the construction of a utility plant in an oil sands development (called the Kearl Initial Development) by PCL Industrial Management Inc. This is a typical mega size oil sands project, being carried out in the province of Alberta, Canada. The project size is beyond what can be illustrated in this paper; hence, this paper only focuses on pipe-rack area of the utility plant (the pink color area in Figure 3).

The major work in this case study consists of setting pre-assembled pipe-rack modules. Other work includes piling, support structure (foundation), piping (interconnecting modules), silencer installation, hydro-testing, insulation as well as electrical cable tray. A total of 51 work packages are identified with quantity, work areas, precedence relationships, and resource requirements. The details are omitted due to the page limit. Assume that the starting date for this sub-project is September 10th, 2012. After running the simulation, it returns a completion date at January 7th, 2013.

The same case study is also implemented manually in Microsoft Project 2010 and Primavera P6 Professional R8.2. Microsoft Project 2010, after resource leveling, reaches a project completion date at Jan. 24th, 2013. Since MS Project is using its own proprietary algorithm, there is no other resource allocation algorithm to choose. P6 results in the same project completion date when using the least total float as the main resource allocation. Since P6 allows users to experiment with many other heuristics (early start/finish, late start/finish, total float, free float, etc.), a few more experiments are carried out. When adopting free float as the main heuristic, P6 manages to return a project completion date at January 22th, 2013. However, this result is still not as desirable as the one also returned by the simulation. This is because the simulation allows for variable resource allocation and variable durations in execution of a work package while holding logic relationships and space congestion constraints; therefore, the proposed system reduces the resource idle time and returns a shorter project duration than traditional scheduling approaches.

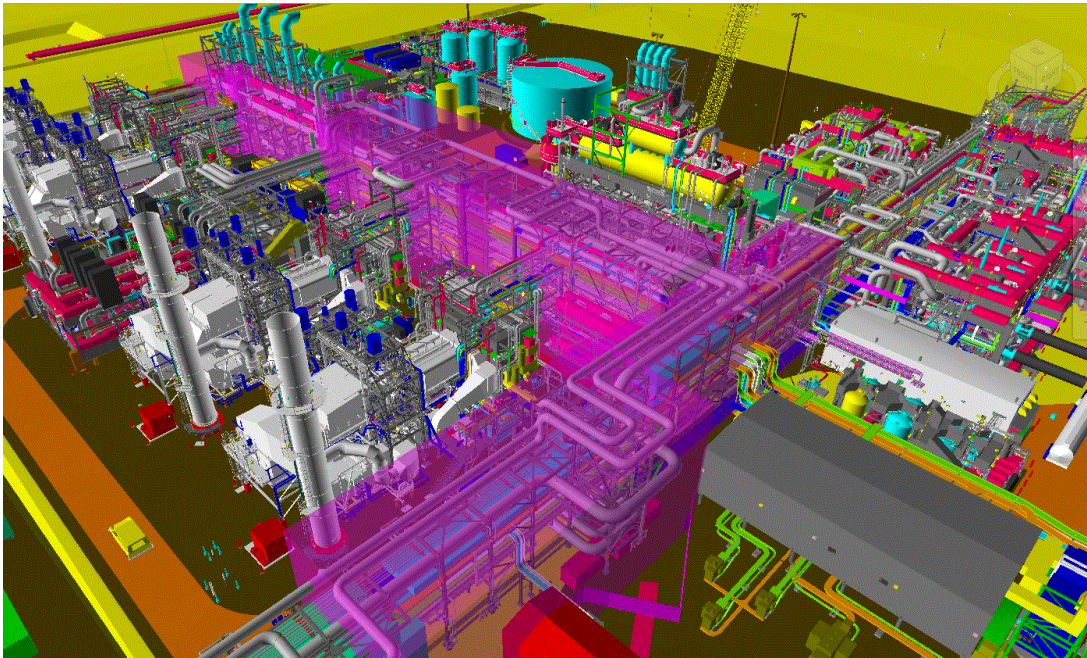


Figure 3: Pipe-rack area in Kearsley Initial Development (KID) project

6 CONCLUSIONS

This paper presents a time-stepped simulation-based scheduling framework that (1) complies with various constraints of work packages (precedence dependency, time dependent resource limit, calendar, as well as time constraint), (2) dynamically allocates resources to work packages, and (3) accounts for jobsite congestion limits of work areas. It is an automated simulation-based scheduling process where users do not have to get involved in model development. A real industrial construction case was used to test the practicality of the framework. It is observed that the framework returns a shorter project duration (13 working days shorter) than those derived from MS Project and Primavera P6.

REFERENCES

- AbouRizk, S., and Y. Mohamed. 2000. "Symphony – An Integrated Environment for Construction Simulation." In *Proceedings of the 2000 Winter Simulation Conference*, Edited by J. A. Joines, R. R. Barton, K. Kang, P. A. Fishwick, 2: 1907-1914. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Ahuja, H. N., and V. Nandakumar. 1985. "Simulation Model to Forecast Project Completion Time." *Journal of Construction Engineering and Management* 111(4): 325–342.
- Christodoulou, S. 2010. "Scheduling Resource-constrained Projects with Ant Colony Optimization Artificial Agents." *Journal of Computing in Civil Engineering* 24(1): 45–55.
- Demeulemeester, E. L., and W. S. Herroelen. 2002. *Project Scheduling: A Research Handbook*. Kluwer Academic Publishers, Boston, Mass, USA.
- Dozzi, S. P., and S. M. AbouRizk. 1993. *Productivity in Construction*. Institute for Research in Construction, National Research Council, Ottawa, ON, Canada.
- ECC. 2007. "Tackling the Unique Challenges of Mega Projects." Panel discussion. In *Proceedings of the 39th ECC Conference, Redefining Our Industry: Solutions through Collaboration, Innovation and Organization*, Sept. 2007.
- Elbeltagi, E., T. Hegazy, and D. Grierson. 2005. "Comparison Among Five Evolutionary-based Optimization Algorithms." *Journal of Advanced Engineering Informatics* 19(1): 43 – 53.

- Hegazy, T. 1999. "Optimization of Resource Allocation and Leveling Using Genetic Algorithm." *Journal of Construction Engineering and Management* 125(3): 167–175.
- Hu, D., and Y. Mohamed. 2010. "State-Based Simulation Mechanism for Facilitating Project Schedule Updating." In *Proceedings of the Construction Research Congress 2010*, Edited by J. Ruwanpura, Y. Mohamed, S. Lee, 369-378. Reston, Virginia: ASCE.
- Karshenas, S., and D. Haber. 1990. "Economic Optimization of Construction Project Scheduling." *Journal of Construction Management and Economics* 8(2): 135-146.
- Lu, M., H. C. Lam, and F. Dai. 2008. "Resource-constrained Critical Path Analysis Based on Discrete Event Simulation and Particle Swarm Optimization." *Automation in Construction*, 17(6): 670–681.
- Lu, M., and H. Li. 2003. "Resource-activity Critical-path Method for Construction Planning." *Journal of Construction Engineering and Management* 129(4): 412–420.
- Mohamed, Y., D. Borrego, L. Francisco, M. Al-Hussein, S. Abourizk., and U. Hermann. 2007. "Simulation-based Scheduling of Module Assembly Yards: Case Study." *Engineering, Construction and Architectural Management* 14(3): 293-311.
- Moselhi, O., and P. Lorterapong. 1993. "Near Optimal Solution for Resource-constrained Scheduling Problems." *Construction Management and Economics* 11: 293-303.
- Song, L., and S. M. AbouRizk. 2006. "Virtual Shop Model for Experimental Planning of Steel Fabrication Projects." *Journal of Computing in Civil Engineering* 20(5): 308–316.
- Thabet, W. Y., and Y. J. Beliveau. 1994. "Modeling Work Space to Schedule Repetitive Floors in Multistory Buildings." *Journal of Construction Engineering and Management* 120(1): 96–116.
- Zouein, P. P., and I. D. Tommelein. 2001. "Improvement Algorithm for Limited Space Scheduling." *Journal of Construction Engineering and Management* 127(2): 116–124.

AUTHOR BIOGRAPHIES

DI HU is currently project control engineer with PCL. He received his PhD from the University of Alberta, with a research focus on simulation based scheduling for construction projects and automated planning and sequencing of industrial construction operations. His email address is dhu@ualberta.ca.

YASSER MOHAMED is an Associate Professor in Construction Engineering and Management in the Department of Civil and Environmental Engineering, at the University of Alberta. His research focuses on simulation modelling of construction processes to support project planning and control. His email address is yaly@ualberta.ca.