# Client-Server Interaction Knowledge Discovery for Operations-Level Construction Simulation Using Process Data

Reza AKHAVIAN[1] and Amir H. BEHZADAN[2]

[1] Ph.D. Student, Department of Civil, Environmental, and Construction Engineering, University of Central Florida, Orlando, FL 32816-2450, USA; PH (407) 823-2480; FAX (407) 823-3315; email: reza@knights.ucf.edu

[2] Wharton Smith Faculty Fellow and Assistant Professor, Department of Civil, Environmental, and Construction Engineering, University of Central Florida, Orlando, FL 32816-2450, USA; PH (407) 823-2480; FAX (407) 823-3315; email: amir.behzadan@ucf.edu

## ABSTRACT

For the past few decades, construction researchers have investigated the potential of simulation modeling in providing decision-makers with insights into different aspects of a project. Simulation can assist in studying the performance of an engineering system during early planning, pre-construction, execution, and maintenance. This is typically done by providing means and methods for preparing work plans, look-ahead schedules, and what-if analysis. However, most existing construction simulation paradigms address long-term planning and scheduling needs. Work is still needed to disseminate the use of simulation-based decision-making in short-term operations-level planning and control during project execution phase. Simulation models, and in particular, discrete event simulation (DES) models are often employed to get insight into the performance of systems that are repetitive yet stochastic in nature. A good example of such systems that can be found in almost all industries is queuing systems. In construction, queues are ubiquitous due to specific project plans, resource allocation patterns, or operational bottlenecks. Certain client-server interaction schemes determine the properties by which a queue is characterized and subsequently modeled in a computer simulation model. This paper presents a methodology that enables data collection, fusion, and mining from construction resources in a queuing system to discover necessary knowledge and generate and update corresponding simulation models.

## INTRODUCTION

The prospect of replacing human-centered decision-making by simulation-centered decision-making through the use of real time simulation systems has been investigated in domains such as operations research and artificial intelligence (Min and Yih 2003; Buro and Churchill 2012). If computer simulation is sought to generate realistic and timely results of a quality that can be readily used to make practical decisions, first and foremost, it must generate a true model of the real world system that it is representing. Construction simulation models are no exception as they are built to represent real engineering operations that generally evolve over time, and inherit several uncertainties and dynamic interactions between entities (AbouRizk et al. 2011). It is well known that a less accurate input to a simulation model will almost certainly result in a less reliable output (Kelton 1997; Banks 2005). Despite the extensive body of knowledge in construction simulation that covers several paradigms such as discrete event simulation (DES) and high-level architecture (HLA) (Taghaddos et al. 2008), continuously updating and modifying construction simulation models has never been a trivial task. The main reason behind this complexity is the multifaceted and unstructured nature of most construction projects that makes it tedious (if not impossible) to manually collect the most recent data and feed them in a timely manner into the simulation model. A quick industry survey highlights this challenge as a major impediment to the accreditation of simulation as a reliable decision support tool (AbouRizk 2010). This shortcoming has been also recently identified in a number of studies in the area of Architecture, Engineering, Construction and Facility Management (AEC/FM) (Akhavian and Behzadan 2012a; Lee et al. 2013).

A promising approach to bridge this important gap in developing adaptive simulation models that can evolve over time is to incorporate advanced knowledge discovery and data mining (KDD) into the core simulation framework. In fact, knowledge acquisition has been identified as the first major barrier for construction industry implementation of modeling and simulation (AbouRizk et al. 2011). The authors have previously investigated the potential of collecting, fusing, and mining process data by developing an integrated framework for construction equipment data driven simulation models (Akhavian and Behzadan 2012b; Akhavian and Behzadan 2013). There have also been other (mainly) sparse attempts by other researchers to address this problem in limited scopes (Hammad and Zhang 2011; Song and Eldin 2012). However, most such work mainly targeted field entities as individual data sources and used collected data streams to model site layout and activity durations. In doing so, very little (if any) attention was paid to using collected process data to gain in-depth understanding of how these entities would interact with one another over time, and whether there were predominant work patterns dictated by certain entities to the entire operations. Although the importance of enabling simulation models to accept input that describe not only individual entities but also the relationships and interdependencies between different entities and sequence of operations has been previously discussed (AbouRizk et al. 2011), this topic has been mostly overlooked in the literature.

According to the Handbook of Simulation (Banks 1998), examples of such underrated yet critical information in engineering systems include the arrival of orders in a material storage facility, times between arrivals to a service area, and times between machine breakdowns. Such examples are also very common in construction jobsites; dump trucks waiting to receive service from an excavator or loader, material orders arriving in an onsite storage facility, or concrete delivery and placing operations. All such examples consist of two types of resources one of which gives service while the other receives it. In queue terminology, the former is called a “server” and the latter is called a “client” or “customer” (Kingman 1992). Typically, a queue of clients waiting for the service is expected to form at the server location since (1) the resource that is transported through the client-server interaction is distributed in parts in each instance, (2) almost always the number of servers is less than the number of clients, and (3) giving service takes some time. The logics and detailed characteristics of waiting lines are studied under a branch of operations research called queuing theory for mathematical modeling and analysis of systems that provide service to random demands (Ralston et al. 2000).

### Queue properties

A queuing system is defined by certain characteristics describing the basic elements of a waiting line model based on client-server interactions, arrival process, service mechanism, and queue discipline (Banks 2005). A schematic representation of a simple queuing model is illustrated in Figure 1.

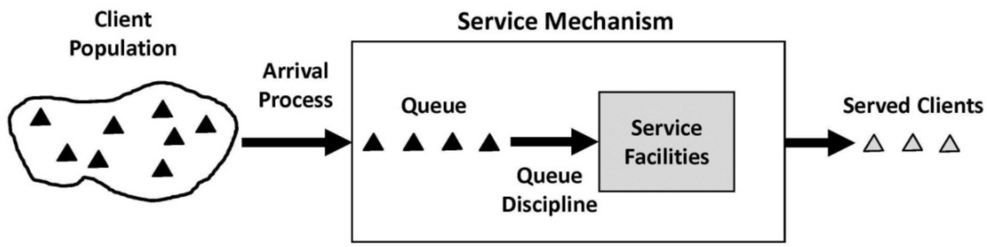


**Figure 1. Simple queuing model**

The arrival process is usually specified in terms of interarrival times of successive clients. In most cases, arrivals occur at random points in time and thus the interarrival times are usually characterized by a probability distribution. In queuing theory, the most widely used stochastic process for modeling random arrivals is the Poisson process (Kingman 1992). In a Poisson arrival process, if the interarrival time between customers $n$-1 and $n$ is represented by $A_n$ ($A_1$ being the arrival time of the first customer), then $A_n$ is exponentially distributed with a mean of $1/\lambda$ time units, which means that the arrival rate is $\lambda$ costumers per unit of time. The service mechanism describes (1) the number of servers used to give service to clients, and (2) the time it takes to serve one client, or the service time. Service time is usually dependent on the type of the costumer and is often characterized as a sequence of independent and identically distributed (IID) random variables. The queue discipline refers to the pattern or rule following which customers are selected to be served as soon as the server becomes available. First-in-first-out (FIFO), last-in-first-out (LIFO), priority-based service (PR), and service-in-random-order (SIRO) are common queue disciplines in real world queuing systems (Banks 2005).

**Queuing systems in construction**

Queue systems have been the subject of several studies within the domain of construction operations (Dunlop and Smith 2002; Darren Graham et al. 2005; Oberguggenberger 2005). Although the Poisson process and consequently the exponential distribution of interarrival times are commonly used to model queuing systems, it may not be necessarily applicable to all waiting systems. For example, some field observations indicate that actual haul unit arrival rates do not precisely follow the Poisson distribution (Nunnally 2000). In another study, it was shown that the concrete delivery and placement process is best represented by the lognormal probability distribution (Darren Graham et al. 2005). Similarly, the service time can vary under different project conditions and client-server interaction schemes. Furthermore, different services require diverse queue discipline; dump trucks forming a waiting line to be loaded by an excavator may be perceived to have a FIFO queue discipline by default, whereas in the presence of certain constraints the discipline may be altered mid operation. For instance, when various dump truck sizes are available, the priority may be given to bigger dump trucks, thus making them wait less in the queue and ultimately consuming less fuel. Furthermore, the LIFO queue discipline may seem to be rarely used with little implications in construction domain. However, a closer look at onsite material handling facilities reveals that releasing material from storage facilities often follows such pattern. In other words, the part that is stored last is often drawn out first from the storage (e.g. stacks of pipe spools, line of prefabricated elements).

Such wide range of applications of various queuing models in construction operations that are typically good candidates for DES modeling necessitates that the queuing aspect of such systems is adequately studied and modeled in order to provide reliable simulation outputs. Therefore, the main aim of the work presented in this paper is to investigate methods that facilitate the extraction of computer-interpretable knowledge pertaining to queue properties and client-server interactions.

## METHODOLOGY

In this research, three modes of data (i.e. positional data, body orientation, and payload) are collected from construction resources (i.e. clients and servers) at any point in time. Positional data can be captured using sensor-based data acquisition technologies such as the global positioning system (GPS), ultra-wide band (UWB), or radio frequency identification (RFID). As constantly streaming data is received and processed, knowledge about the time at which an entity arrives at the queuing system (a.k.a. arrival or birth time), starts receiving service, and finishes receiving service (a.k.a. leaving or death time) is discovered. Moreover, since the data acquisition system receives raw data from individual sensors each attached to a server or a client, the order by which clients are served can be also detected. This information is used by the reasoning process to discover knowledge about the queue discipline, since both arrival times and service start times are known given the knowledge extracted from the time-stamped data. In this manner, all three properties of the queuing system, namely interarrival times, service times, and queue discipline can be discovered.

Since the knowledge discovery procedure relies on the spatio-temporal trend of data, a data preparation procedure is also included in the overall methodology. Such procedure includes finding the outliers that may result in detecting specific events (e.g. a client approaching a server to start receiving service) and consequently, suggesting a change in the state of the target entity. The data preparation procedure also accounts for the missing data points that accidentally failed to be collected due to system errors or ambient noise. Because the main goal of extracting such knowledge about client-server interactions in the real system is to use this knowledge inside a simulation model and update the model input parameters accordingly, it should be next translated into meaningful statements for the simulation model. While queue discipline is directly fed into the simulation model, the other two components of queuing systems (i.e. interarrival and service duration times) should be presented as probability distributions inside the simulation model. Therefore, a distribution that is well-fitted to the histogram of extracted interarrival and service times should be selected. In order to measure how well a distribution fits a pool of collected data, goodness-of-fit tests can be performed. Figure 2 depicts the general approach based on which the presented study is designed.

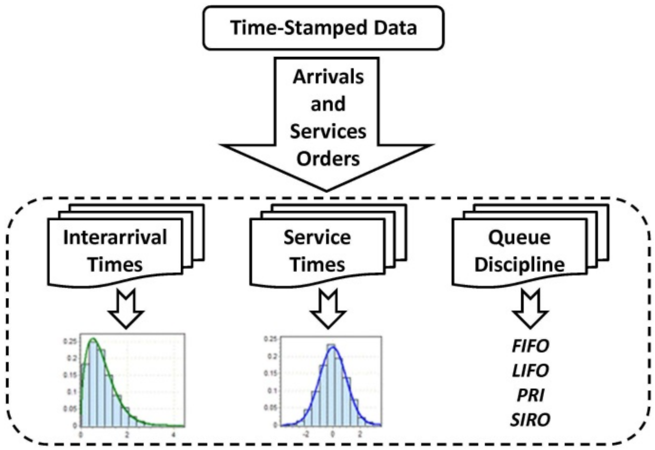


**Figure 2. Queue properties discovery approach**

The goodness-of-fit test steers the process of input modeling by evaluating different potential probability distributions. However, it should be noted that there is no single correct distribution in real world applications, so the final selection based on all resulting candidates is not the absolute result and can be replaced by other candidates per user's discretion. In order to mathematically assess if a probability distribution fits the histogram of extracted information, three tests are commonly used: (1) Kolmogorov–Smirnov (K-S) test, (2) Anderson–Darling (A-D) test, and (3) Chi–Square test. All such goodness-of-fit tests are generally structured so that they set up two hypotheses and test them versus each other. The first hypothesis, $H_0$ (i.e., the null hypothesis) indicates that the data actually conforms to the assumed distribution, while the other, $H_1$, indicates that data does not conform to the distribution of choice (Banks 2005).

## VALIDATION EXPERIMENTS

In order to validate the possibility of collecting pertinent client-server interaction data for the purpose of discovering knowledge that describes three important properties of queues, a set of experiments were performed in the Decision Support, Information Modeling, and Automation Laboratory (DESIMAL) of the University of Central Florida (UCF). All experiments were conducted using an UWB localization system that consisted of UWB receivers at the periphery of the target experiment zone, UWB tags attached to the target entities (herein server(s) and clients), a hub as a central processing unit, and a computer for system configuration and data storage. Experiments were designed using 10 UWB tags representing clients waiting in a queue to be served by another UWB tag that represented the server. In each experiment, a specific service pattern representing queue discipline was implemented to evaluate the functionality of the developed framework in detecting the correct queue discipline in addition to the interarrival and service time probability distributions. Also, in each experiment, more than 50 instances of clients' arrival-service-leave cycles in a time span of 15 to 20 minutes were created, and in total, around 78,000 time-stamped positional data points were collected from the server(s) and clients. Table 1 lists a sample of collected data points in one of the experiments.

**Table 1. Sample Data Points from One Experiment**

| Time | Tag ID | X | Y | Z | Tag Status |
|---|---|---|---|---|---|
| 18:28:06 | 020-000-147-212 | 3.08748 | 3.02152 | 0.178556 | In System |
| 18:28:07 | 020-000-147-252 | 3.62637 | 3.0435 | 0.265829 | In System |
| 18:28:07 | 020-000-147-020 | 3.89411 | 3.06027 | 0.132566 | In System |
| 18:28:07 | 020-000-147-207 | 3.87354 | 3.06197 | 0.148414 | In System |
| 18:28:07 | 020-000-147-175 | 3.79864 | 3.08372 | 0.13733 | In System |
| 18:28:07 | 020-000-147-211 | 3.706 | 3.06462 | 0.118912 | In System |
| 18:28:07 | 020-000-154-195 | 3.90496 | 3.04561 | 0.1512 | In System |
| 18:28:07 | 020-000-148-011 | 4.65458 | 3.95913 | 0.316904 | Just Born |
| 18:28:07 | 020-000-147-220 | 3.8502 | 3.04176 | 0.155713 | In System |
| 18:28:07 | 020-000-147-212 | 3.08696 | 3.02207 | 0.179014 | In System |
| 18:28:07 | 020-000-147-252 | 3.62177 | 3.04439 | 0.265192 | In System |
| 18:28:07 | 020-000-147-020 | 3.89573 | 3.06664 | 0.131962 | In System |

Collected data represented the coordinates of each tag (in a 3D space) identified with the tag ID at any time during the experiment. Based upon the arrival, service start, and service finish times of tags, the interarrival and service durations were computed. Furthermore, tags IDs helped in listing and detecting the trends of arrival and service orders that ultimately resulted in understanding about the queue discipline. Table 2 shows sample of information extracted from the raw data of the same experiment represented in Table 1. In three of the conducted experiments, queue disciplines were selected to be FIFO, LIFO, and PR (i.e. even tag IDs had priority over odd tag IDs). This property along with the interarrival times and service durations were precisely discovered using the designed data mining methods. Results are presented in the following Subsection.

**Table 2. Sample Knowledge Discovered from One Experiment**

| Arrival Order | Arrival Times | Interarrival Times | Service Order | Service Start | Service Finish |
|---|---|---|---|---|---|
| 020-000-147-020 | 18:25:38 | 0:00:17 | 020-000-147-020 | 18:27:27 | 18:27:39 |
| 020-000-154-195 | 18:25:55 | 0:00:21 | 020-000-154-195 | 18:27:42 | 18:27:47 |
| 020-000-148-011 | 18:26:16 | 0:00:19 | 020-000-148-011 | 18:27:50 | 18:27:54 |
| 020-000-147-212 | 18:26:35 | 0:00:07 | 020-000-147-212 | 18:27:57 | 18:28:17 |
| 020-000-147-252 | 18:26:42 | 0:00:07 | 020-000-147-252 | 18:28:20 | 18:28:26 |

**Experiment results**

In order to rank probability distributions that fit the extracted interarrival times and service durations, @RISK statistical analysis software was used (Palisade Corporation 2013). For each experiment, best fitted distributions were ranked based on their test statistics that were previously introduced for each goodness-of-fit test. Table 3 ranks the candidate probability distributions (i.e. Beta, Erlang, Exponential, Gamma, Normal, and Triangular) for interarrival times and service durations based on the Chi-Square, K-S, and A-D goodness-of-fit tests for all three experiments.

**Table 3. Candidate Distribution Rankings Based on Three Goodness-of-fit Tests**

| Experiment | Test | Beta | Erl. | Exp. | Gam. | Norm. | Tri. |
|---|---|---|---|---|---|---|---|
| 1 | **Interarrival** | | | | | | |
| | *Chi- Square* | 5 | 1 | 2 | 4 | 6 | 3 |
| | *K-S* | 1 | 3 | 4 | 2 | 5 | 6 |
| | *A-D* | 2 | 3 | 4 | 1 | 5 | 6 |
| | **Service Time** | | | | | | |
| | *Chi- Square* | 3 | N/A | 4 | N/A | 1 | 2 |
| | *K-S* | 3 | N/A | 4 | N/A | 1 | 2 |
| | *A-D* | 2 | N/A | 4 | N/A | 1 | 3 |
| 2 | **Interarrival** | | | | | | |
| | *Chi- Square* | N/A | 3 | 4 | 1 | 5 | 2 |
| | *K-S* | N/A | 2 | 3 | 1 | 4 | 5 |
| | *A-D* | N/A | 3 | 4 | 1 | 3 | 5 |
| | **Service Time** | | | | | | |
| | *Chi- Square* | 1 | N/A | 4 | N/A | 2 | 3 |
| | *K-S* | 1 | N/A | 4 | N/A | 2 | 3 |
| | *A-D* | 1 | N/A | 4 | N/A | 3 | 2 |
| 3 | **Interarrival** | | | | | | |
| | *Chi- Square* | N/A | 1 | 5 | 2 | 4 | 3 |
| | *K-S* | N/A | 1 | 5 | 2 | 3 | 4 |
| | *A-D* | N/A | 2 | 4 | 1 | 3 | 5 |
| | **Service Time** | | | | | | |
| | *Chi- Square* | 2 | N/A | 4 | N/A | 1 | 3 |
| | *K-S* | 2 | N/A | 4 | N/A | 1 | 3 |
| | *A-D* | 2 | N/A | 4 | N/A | 1 | 3 |

As shown in Table 3, different goodness-of-fit tests resulted in different rankings of the candidate probability distributions. However, it is observed that for each experiment, one (two – in case of the second experiment interaarival times) particular distribution(s) stands out as the best fit based on the least average of the three rankings. In Table 3, these probability distributions are highlighted for each experiment. For example, in experiment 1, Erlang and Gamma distributions can be selected as the best fit due to their high overall ranks resulting from all three tests. Cells containing N/A indicate that the corresponding probability distributions were not valid to be fit to the resulting interarrival or service times.

## IMPLICATION OF RESULTS IN A SIMULATION MODEL

A DES model was created to demonstrate the implication of the discovered knowledge. The activity cycle diagram (ACD) of the model is illustrated in Figure 3. As shown in this Figure, clients arrive (i.e. are born) one by one in the `Arrival` Activity that draws one client at a time from the `Dummy` queue, wait in line (i.e. `ClientsWait`

Queue) for being served by the server (i.e. `Service` Activity) as soon as the server is available (that is when the `ServerWait` Queue is not empty), and finally depart the system to the `ClientsLeft` Queue.

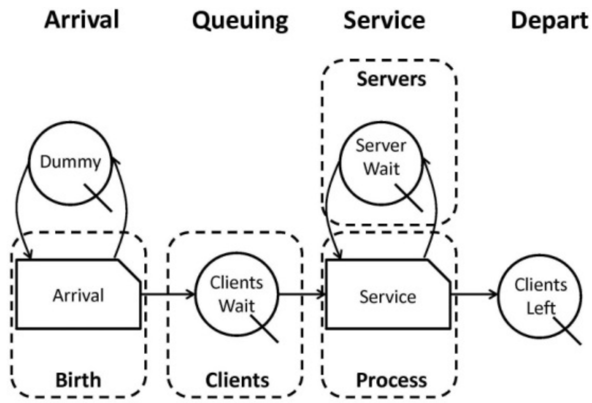


**Figure 3. Activity cycle diagram of a queuing system**

Using the discovered knowledge from the first experiment, the process was modeled inside Stroboscope, a simulation programming language customized for construction operations (Martinez 1996). As soon as the knowledge required for updating the simulation model (i.e. interarrival times, service times, queue discipline) is obtained, the corresponding expression in the simulation script is accordingly updated. In particular, the Interarrival time probability distribution serves as the duration of the `Arrival` Activity, the service times probability distribution is used as the duration of the `Service` Activity, and the queue discipline is inputted and updated through `DESCIPLINE` command that serves as a property of the `ClientsWait` Queue. According to Table 3, for the first experiment, the Normal distribution is significantly superior to other distributions for service time and therefore, it was used inside the simulation model. For interarrivals, the best distributions, namely Gamma and Erlang were selected for use inside the model along with Exponential, a distribution widely used for modeling interarrivals in queuing problems (Nunnally 2000). Next, as a measure of effectiveness, the waiting time of the clients was extracted from the collected data and compared to the waiting time listed by Stroboscope output for `ClientsWait` Queue. The extracted waiting time was 96 seconds and the mean waiting times resulted from ten replications of running the simulation model are shown in Table 4. As Table 4 suggests, significant discrepancies between the mean waiting times occurred when different probability distributions were used. However, it was interesting to observe that the Erlang distribution showed closer result to the actual mean waiting time of 96 seconds. Even Exponential distribution had a relatively better performance compared to the Gamma distribution which was one of the best selection according to Table 3. This indicates that discovering knowledge through using factual data may affect the selection of input probability distribution and thus assist in providing the model with higher quality input model than what is typically suggested by goodness-of-fit tests.

**Table 4. Means and Standard Deviations of Waiting Times Based on Three Probability Distributions**

| | Gamma | Erlang | Exponential |
|---|---|---|---|
| **Mean Waiting Time (Sec)** | 10.84 | 114.15 | 14.42 |
| **Std Waiting Time (Sec)** | 3.16 | 20.21 | 10.65 |

## CONCLUSIONS AND FUTURE WORK

Client-server interaction has many applications in construction operations. Most often, this interaction yields to formation of waiting lines the performance of which can be best modeled and analyzed using simulation tools. Having a realistic input data for a simulation model pertaining to client-server relationships can result in a more reliable output from that model. In this study, a methodology was presented that targeted three main properties of queues, namely interarrival time, service time, and queue discipline. Through a set of experiments, required data for discovering knowledge about these properties were conducted. Probability distributions were fit to the interarrival times and service durations and used inside a simulation model. Comparison of results obtained from the simulation model generated by selected probability distributions based on goodness-of-fit tests to the actual measures observed in the real system indicated that probability distributions selected based on the goodness-of-fit tests may not be always the best options; rather a knowledge-based simulation model can assist in finding a better representative distribution. Further research is required to evaluate the efficiency of knowledge-based simulation in providing more realistic and robust representations of real construction systems.

## REFERENCES

AbouRizk, S. (2010). "Role of Simulation in Construction Engineering and Management." *J. Constr. Eng. Manage.*, 136(10), 1140-1153.

AbouRizk, S., Halpin, D., Mohamed, Y., and Hermann, U. (2011). "Research in modeling and simulation for improving construction engineering operations." *J. Constr. Eng. Manage.*, 137(10), 843-852.

Akhavian, R., and Behzadan, A. H. (2012a). "An Integrated Data Collection and Analysis Framework for Remote Monitoring and Planning of Construction Operations." *Adv. Eng. Inf.*, 26(4), 749–761.

Akhavian, R., and Behzadan, A. H. (2012b). "Remote Monitoring of Dynamic Construction Processes Using Automated Equipment Tracking." *Proc., Construction Research Congress 2012*, ASCE, 1360-1369.

Akhavian, R., and Behzadan, A. H. (2013). "Knowledge-Based Simulation Modeling of Construction Fleet Operations Using Multimodal-Process Data Mining." *J. Constr. Eng. Manage.*, 139(11), 04013021.

Banks, J. (1998). *Handbook of simulation : principles, methodology, advances, applications, and practice*, Wiley, New York.

Banks, J. (2005). *Discrete Event System Simulation*, Pearson Education, New Jersey.

Buro, M., and Churchill, D. (2012). "Real-Time Strategy Game Competitions." *AI Magazine*, 33(3), 106-108.

Darren Graham, L., Smith, S. D., and Dunlop, P. (2005). "Lognormal distribution provides an optimum representation of the concrete delivery and placement process." *J. Constr. Eng. Manage.*, 131(2), 230-238.

Dunlop, P. G., and Smith, S. D. (2002). "Simulation analysis of the UK concrete delivery and placement process–a tool for planners." *Proc., 18th Annual ARCOM Conference*, 2-4.

Hammad, A., and Zhang, C. (2011). "Towards Real-Time Simulation of Construction Activities Considering Spatio-Temporal Resolution Requirements for Improving Safety and Productivity." *Proc., Proceedings of the 2011 Winter Simulation Conference*, Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 3533-3544.

Kelton, W. D. (1997). "Statistical analysis of simulation output." *Proc., Proceedings of the 29th conference on Winter simulation*, IEEE Computer Society, 23-30.

Kingman, J. F. C. (1992). *Poisson processes*, Oxford university press, USA.

Lee, S., Behzadan, A., Kandil, A., and Mohamed, Y. (2013). "Grand Challenges in Simulation for the Architecture, Engineering, Construction and Facility Management Industry." *Proc., Computing in Civil Engineering*, 773-785.

Martinez, J. C. (1996). "Stroboscope: State and Resource Based Simulation of Construction Processes." Ph.D. thesis, University of Michigan, Ann Arbour, MI.

Min, H.-S., and Yih, Y. (2003). "Development of a real-time multi-objective scheduler for a semiconductor fabrication system." *Int. J. Prod. Res.*, 41(10), 2345-2364.

Nunnally, S. (2000). *Managing Construction Equipment, Prentice Hall, New Jersey*, Prentice-Hall, Upper Saddle River, N.J.

Oberguggenberger, M. (2005). "Queueing models with fuzzy data in construction management." *Analyzing Uncertainty in Civil Engineering*, Springer, 197-210.

Palisade Corporation (2013). "@RISK." <http://www.palisade.com/risk/>. (September 10, 2013).

Ralston, A., Reilly, E. D., and Hemmendinger, D. (2000). *Encyclopedia of computer science*, Nature Publishing Group.

Song, L., and Eldin, N. N. (2012). "Adaptive Real-Time Tracking and Simulation of Heavy Construction Operations for Look-Ahead Scheduling." *Autom. Constr.*, 27, 32-39.

Taghaddos, H., AbouRizk, S. M., Mohamed, Y., and Ourdev, I. (2008). "Distributed agent-based simulation of construction projects with HLA." *Proc., Proceedings of the 40th Conference on Winter Simulation*, Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 2413-2420.