JAVA-BASED SIMULATION OF CONSTRUCTION PROCESSES USING SILK

Anil Sawhney

Jayachandran Manickam
André Mund
Jennifer Marble

Del E. Webb School of Construction
Arizona State University
P.O. Box 870204
Tempe, AZ 85287-0204, U.S.A.

2007 Kohrman Hall, Construction Engineering
Western Michigan University
Kalamazoo, MI 49008-5064, U.S.A.

ABSTRACT

The modeling and analysis of construction processes is gaining recognition in the construction industry. Recent developments such as object-oriented modeling, Petri Net-based simulation, and construction project level simulation are being adopted by researchers in the field of construction simulation. This paper describes one such recent development. It highlights the work performed by the authors in using Java for the simulation of construction processes. The methodology used, simulation toolkit adopted, and the construction process simulated is described.

1 INTRODUCTION AND BACKGROUND

Simulation has recently gained importance in the area of construction engineering and management. Construction simulation has been extensively used for the design and analysis of construction processes (Halpin 1977, Halpin 1990, Halpin and Riggs 1992). The application of the construction process simulation ranges from productivity measurement and risk analysis to resource allocation and site planning (Sawhney and AbouRizk 1993). Recent developments such as object-oriented modeling (Liu and Ioannou 1992), Petri Net-based simulation (Sawhney et al. 1999), and construction project level simulation (Sawhney et al. 1998 and Odeh et al. 1992) are being reported by researchers in the field of construction simulation.

Ibbs (1986) summarized the findings of a workshop organized to develop new research directions in computerized applications to construction engineering and management. The investigators who participated in this workshop recommended that research in the area of simulation focus on developing modeling constructs for "process," "project," and "corporate" level simulation and making simulation modeling systems "friendlier" by using more graphics. Motivated by these findings, the authors

have undertaken an explorative research to develop Java based simulation strategies for the simulation of construction processes and operations over the Internet. This paper describes the tasks that have been performed to accomplish this. In the next section, a brief introduction to the concept of Java based simulation is provided. The third section of the paper describes benefits of the Java based simulation. The fourth section of the paper highlights some application areas of Java based simulation. In section five, Java based simulation toolkits currently available are described. Construction process simulation using Silk is described in section 6. Conclusions and future directions are provided in the last section of the paper.

2 WHAT IS JAVA - BASED SIMULATION?

Simply stated, a simulation programmed in the Java language can be defined as a Java based simulation. However, the term 'Java based simulation' is used more broadly to point to simulations that can be accessed over the Internet using a World Wide Web browser interface. This is also commonly referred to as the web-based simulation or simply "WebSim" (Powersim 1998). In this paper, the terms 'Java based simulation' and 'WebSim' are used interchangeably to mean simulation that can be performed over the Internet. It must be noted that Java based simulations can be compiled and executed locally on a computer. However, this study does not focus on such an approach.

Simulation, which is posted and can be run on the Internet, Intranet or Extranet, can thus be called a Web based Simulation or WebSim. Simulations are usually performed in a standalone environment for testing, experimenting, and studying a system or its prototype. As the ability to share knowledge across wide areas increased dramatically with the help of the Internet, simulation experts began to use the web to perform simulation. Web-

based simulation programs can be divided into the following two major categories (Page et al. 1997):

- Simulation programs that can be accessed remotely through web browsers comprise the first category. Typically, these simulations use forms-based Common Gateway Interface (CGI) scripts to allow the user to provide model execution parameters such as mean service times and arrival rates, number of replications, and so forth. A single copy of the simulation runs on a server and passes the results of model execution to the invoking client (Page et al. 1997).

- The second category of web-based simulation programs represents a variation of the first, but with the added feature of code mobility afforded by such network programming languages as Java. Here, the simulation executes on the client rather than the server (Page et al. 1997). In essence the simulation modeler develops a Java - based applet that is embedded in a web page. This category of web-based simulation shows a lot of promise and is being explored by academic and industry experts. The construction process simulations described in the paper belong to this category.

3 BENEFITS OF JAVA - BASED SIMULATION

Java - based simulation presents a unique opportunity for revolutionary changes in the process of developing simulation models and in the mission of the simulation software firms that provide tools to support the model development phase (Kilgore et al. 1998). The use of Java as a programming language for simulation provides a number of direct benefits. They are as follows:

- Java is an object-oriented programming language that is capable of producing hierarchical, modular, and reusable simulation applications. The resulting object-oriented simulation has been shown to provide rich and lucid paradigm for building computerized models of real world phenomena (Roberts and Heim 1988, Rothenberg 1986).
- Java provides "write once, run anywhere" platform independence (Healy and Kilgore 1997). Simulations developed using Java can thus be distributed and shared freely.
- Java includes native support for networking and common Internet protocols, database connectivity, multi-threading, distributed

objects, and graphical user interfaces (Healy and Kilgore 1997).

Figure 1 shows the general framework under which a Java based simulation is developed. The developer of the Java based simulation toolkit first taps the benefits of Java programming language. The simulation modeler can not only draw benefits from the toolkit but also can also at the same time, draw benefits from the Java language by extending the toolkit in the simulation program. In general, it is felt that Java based simulation will expose the benefits of computer simulation to a larger audience of problem-solvers, decision-makers and trainers since models can be distributed and executed over the Internet using standard browser software on any operating system and hardware platform (Healy and Kilgore 1997).
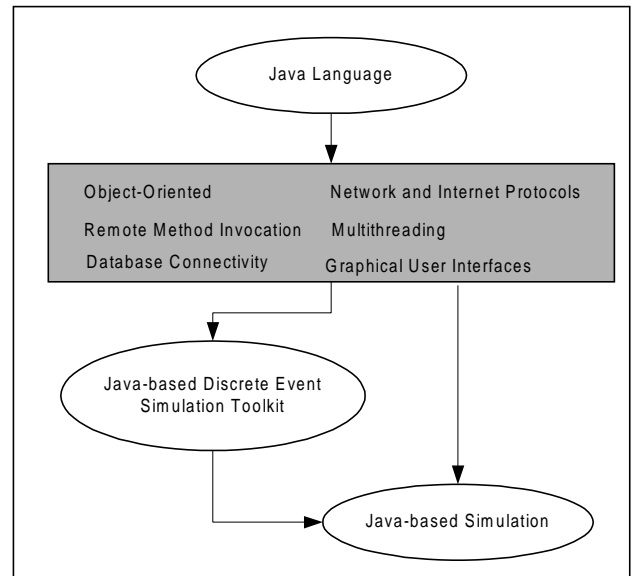

Figure 1: Framework of Java – Based Simulation

4 APPLICATIONS OF JAVA BASED SIMULATION

Java based simulation can be used beneficially in a number of ways. The following list provides some potential applications of Java based simulation:

- Simulation for Teaching and Training: The availability of simulation as a desktop, browser-based tool has the potential to significantly alter the current teaching and training methodologies (Page et al. 1997, Fishwick 1997). The primary objective of the authors in the development of Java based construction process simulations, is to enhance the undergraduate construction engineering teaching. They are developing

these simulations as part of a National Science Foundation research project titled "Internet - based Interactive Construction Management Learning System" that uses Internet based tools to enhance student education (Sawhney and Mund 1998).

- Web-Based Collaborative Decision-Making: Java - based simulation that can be accessed over the Internet can be effectively used as a collaborative decision making tool. The benefits of such an application increase when the team involved in decision making is present in different locations.

- Simulation-Based Marketing: Manufacturers of products and processes can use web-based simulation to demonstrate their technologies to prospective users over the Internet.

- Simulation Research Methodology: The ability to rapidly disseminate models, results, and publications on the web permits new approaches to the conduct of simulation research (Page et al. 1997).

5 JAVA - BASED SIMULATION TOOLKITS AVAILABLE

A number of discrete event simulation toolkits programmed in Java have been developed recently. Two important Java based discrete event simulation toolkits from the point of view of simulation of construction processes, are Simjava (Howell and McNab 1998) and Silk (Healy and Kilgore, 1997). The following paragraphs provide an overview of the important Java based simulation toolkits available.

Simjava is a Java based discrete event simulation package authored by Fred Howell and Ross McNab (1998). It is conceptually based on HASE++ simulation library (Coe et. al 1998) and Sim++ library for C++ (Jade Simulation International 1992). Simjava is based on a discrete event simulation kernel and includes facilities for representing simulation objects as animated icons on screen. Simjava simulations may be incorporated as "live diagrams" into web documents (Howell and McNab 1998). Currently, integration of Java based simulation written using Simjava and Virtual Reality Modeling Language (VRML) based animation is being conducted.

Silk is a general-purpose simulation language based on the Java programming language. Silk merges familiar process-oriented modeling structures with powerful object-oriented language features in an intelligent design that encourages model simplicity and reusability (Healy and Kilgore 1997). In the Silk simulation language, models are developed directly in the Java programming language using a package of classes consisting of a relatively few powerful process-oriented modeling features (Healy and

Kilgore 1997). The key feature that makes Silk very attractive is that it allows modelers to develop domain specific simulation objects using the JavaBeans methodology. The authors used Silk to perform the construction process simulation described in the paper.

A number of other Java based simulation methodologies are available. JSIM is a simulation toolkit in which models may be built using either the event package (Event-Scheduling Paradigm) or the process package (Process-Interaction Paradigm) (Nair 1997). It supports a good graphical environment for displaying queues, and uses a Java database for storing results. JavaSim is a set of Java packages for building discrete event process-based simulation, similar to that in Simula and C++SIM (Little 1997). SimKit is another Java based discrete event simulation toolkit that is currently being developed (Buss and Stork 1996).

6 CONSTRUCTION PROCESS SIMULATION USING SILK

Silk is a general-purpose, process-oriented modeling language that can be extended by users in a straightforward and unrestricted manner (Healy and Kilgore 1997). It has been developed and marketed by ThreadTec, Inc. In the Silk simulation language, simulation models are developed directly in the Java programming language using a package of classes that provide process-oriented, modeling features (Healy and Kilgore 1997). Figure 2 shows the classes available for developing simulation models in Silk.

The simulation entities, resources, and queues around which a simulation is generally centered, are provided as classes in Silk. The modeler develops problem-specific entities, queues, and resources by directly extending and inheriting these classes from Silk. Silk also provides animation features by providing a set of JavaBeans that have animation features (JavaBeans or simply Beans are software components written in Java. These components are self-contained, reusable software units that can be visually composed into applets or applications using visual application builder tools (Sun Microsystems 1997). Entities, queues, and resources defined by the modeler can be linked to these animation JavaBeans to provide process animation. In addition, other Silk classes can be used to perform other routine simulation tasks such as random distribution generation, statistics collection, and output generation. Silk has been designed in such a manner that it can be extended and customized for a particular domain such as the construction industry. This can be accomplished by developing JavaBeans and integrating them into a simulation model developed using Silk. Due to the basic features of the Java language, these domain specific simulation JavaBeans can also be freely distributed and reused. In order to demonstrate the features of Silk and show its application in construction engineering and management education, the authors have developed
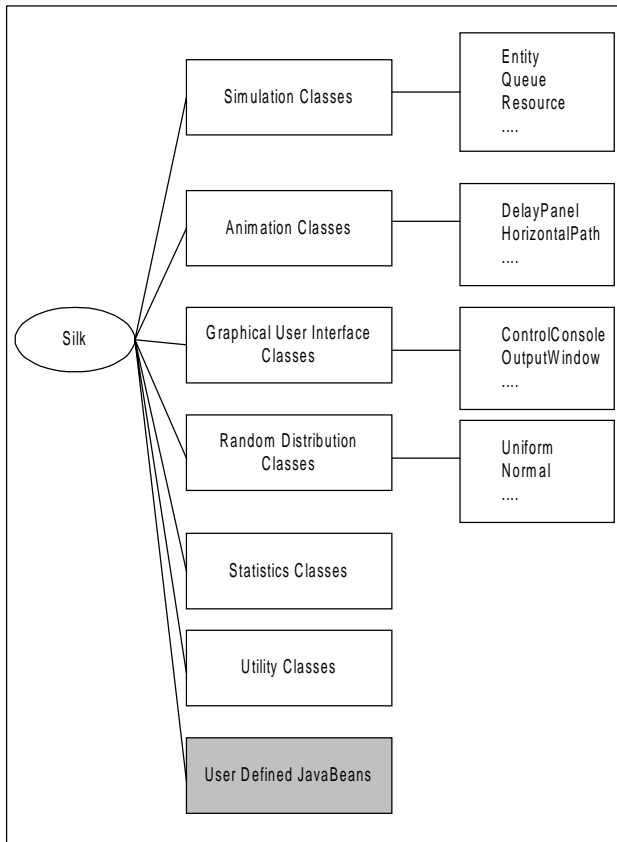
Figure 2: Silk Classes

Figure 3: Schematic Representation of the Earthmoving Process

Figure 4: Defined Entities and Queues

simulation models of common construction processes using Silk. One of the models developed is described in the following sub-section.

6.1   Earthmoving Process

In this example, a Silk based simulation model of the earthmoving process was developed. The process entails moving of dirt stockpiled at location A to location B. Dump trucks are used to haul the dirt from one location to the other and front-end loaders are used to load the dump trucks. Figure 3 shows a schematic illustration of the process under consideration.

Entities, queues, and resources were defined for the above-described process to develop a Silk based simulation model. Figure 4 shows the listing of entities and queues defined for the model.

These classes in Java are derived from the entity and queue classes in Silk. The process logic is captured in these classes. Interaction between these classes mimics the earthmoving process. The "entTruck" entity is defined as the leading entity in this simulation. This entity waits in its queue, "queTruck", and continuously checks the availability of entity "entLoader" and "entDirt" in queues "queLoader" and "queDirt", respectively. Once the"entTruck" finds a unit of "entLoader" and "entDirt", it starts processing. It removes
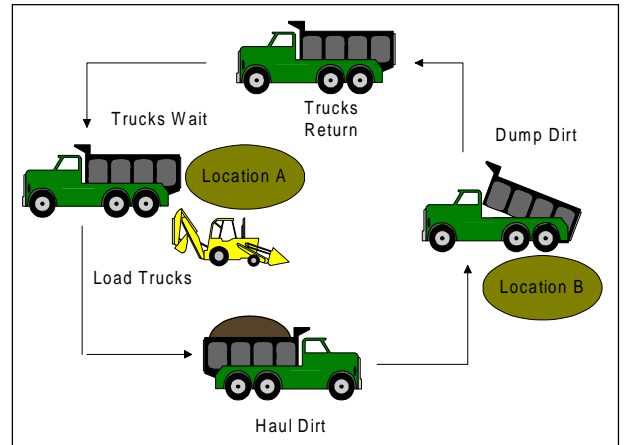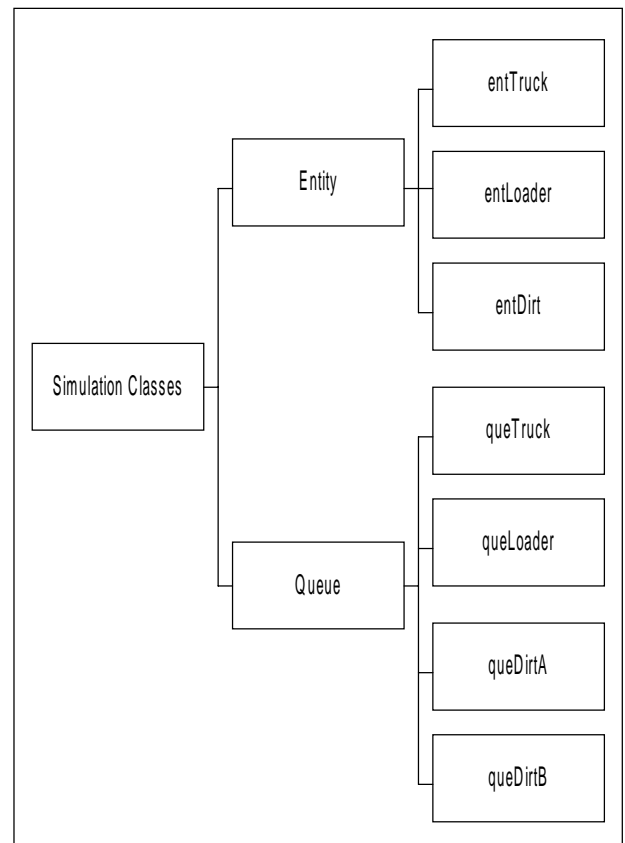
itself and "entLoader" and "entDirt" from the respective queues. The "entLoader" and "entDirt" are activated and begin processing. Positioning, loading, hauling, dumping, and returning tasks are then undertaken. Upon completion of the loading task the entity "entLoader" is released back. After returning, the entity "entTruck" joins the queue "queTruck" again. A portion of the Java code written for the entity "entTruck" is shown in Figure 5.

```
//*********************************************************************************
//  Truck Class
//*********************************************************************************
import com.threadtec.silk.*;
import com.threadtec.silk.util.*;
import com.threadtec.silk.random.*;
import com.threadtec.silk.statistics.*;
import com.sun.java.swing.ImageIcon;

public class Truck extends Entity {

public static double attArrivalTime, attLoadTime, attHaulTime, attDumpTime,
attReturnTime;
static Uniform attHaulTime = new Uniform ( 18.0, 22.0 );
static Uniform attLoadTime = new Uniform ( 1.5, 2.5 );
static Resource resTruck = new Resource ( "Truck" );  // used to provide resource
statistics
static Queue queTruck = new Queue ( "Truck Queue" );
static Observational obsTimeInQueue = new Observational( "Truck idle time" );
static TimeDependent tdsTruckUtil = new TimeDependent( resTruck.numBusy, "Truck
utilization" );

.........

public void process ( ) {
    while( true ) {

        attArrivalTime = time;
        symbol = entSymbol[2];
        queue ( Simulation.queTruck );
        //check the availability of loader and dirt
        while ( condition ( (queLoader.getLength( ) == 0) | (queDirt.getLength( ) == 0)
));

        obsTimeInQueue.record( time - attArrivalTime );
        attArrivalTime = time;
        seize(resTruck);
        dequeue ( queTruck );
        //the loader and dirt in halt conditions are activated by truck
        Loader entLoader = (Loader)queLoader.remove(1);
        entLoader.activate( );
        Dirt entDirt = (Dirt)queDirt.remove(1);
        entDirt.activate( );
        symbol = entSymbol[0];
        delay( attPositionTime, AnimationJFrame.vpl);
        delay( attLoadTime, AnimationJFrame.dpl);
        symbol = entSymbol[1];
        delay( attHaulTime, AnimationJFrame.hp1);
        delay( attDumpTime);
        symbol = entSymbol[2];
        delay( attReturnTime, AnimationJFrame.hp2, true);
        release ( resTruck );
    }}
```

Figure 5: Java Code for Truck Entity

Similar Java code is also written for the two remaining entities. Figure 6 shows the remaining classes defined for the earthmoving simulation.

Once the Java code for the earthmoving process is completely written, the simulation is published to a web server. A client connected to the Internet can then view the published simulation with the help of a standard web browser. When a client launches the simulation, the Control Console is displayed. This is shown in Figure 7. Using the console, the user can configure the simulation and also set variables, such as, number of trucks, number of loaders, etc. Simulation and animation of the earthmoving process can be started, paused, and stopped from the Control Console.

Figure 8 shows the animation screen of the earthmoving process. Once the simulation and animation is completed the Output Window displays the standard output. This is shown in Figure 9.
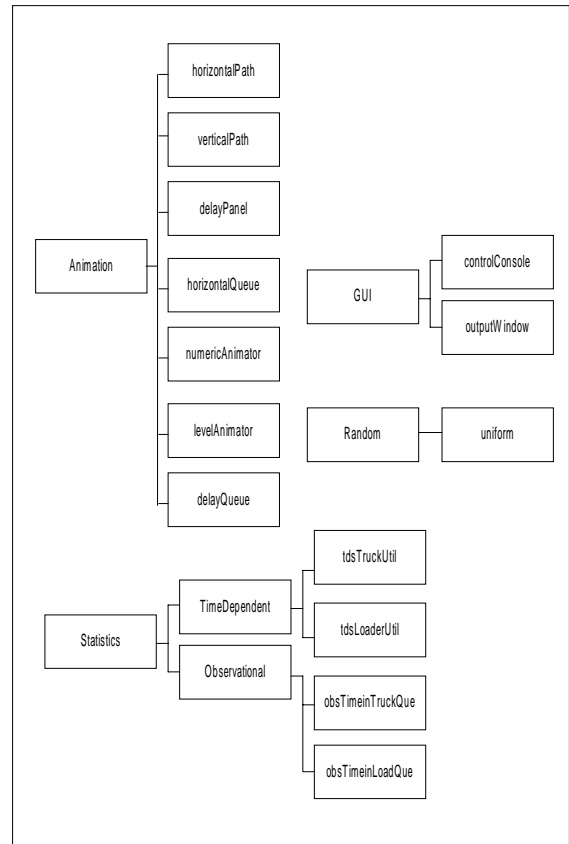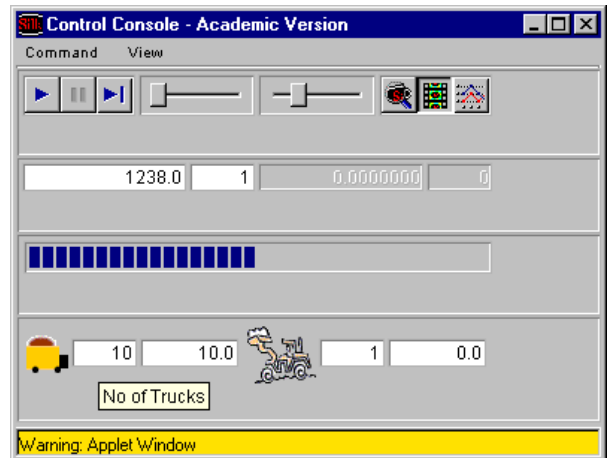

Figure 6: Other Classes

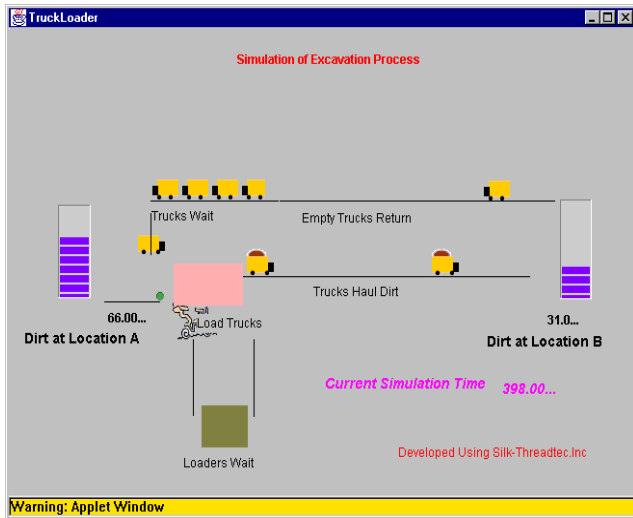Figure 7: Control Console Window

Figure 8: Animation Screen

Figure 9: Standard Output Window

7   CONCLUSIONS

This paper provides information about the development of Java based simulation of the earthmoving process using Silk. The development of the Java based simulation that can be used over the Internet is part of the development of an Internet based interactive learning tool for construction engineering and management education. Currently, the authors are experimenting with the development of JavaBeans for special purpose construction simulation using Silk.

ACKNOWLEDGEMENT

9   REFERENCES

Barrie, D.S. and Paulson, B. C. 1992 *Professional Construction Management* 3rd Ed. New York: McGraw Hill.

Buss, A. H. and Stork, K. A. (1996) Discrete event simulation on the world wide web using JAVA. In *1996 Winter Simulation Conference Proceedings*, ed. J.M. Charnes, D.M. Morrice, D.T Brunner, and J.J. Swain. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Coe, P. S., Howell, F. W. Ibbett, R. N. and Williams L. M. (1998) A Hierarchical Computer Architecture Design and Simulation Environment, *ACM Transactions on Modeling and Computer Simulation*, vol. 8, no. 4.

Fishwick, P. A. (1997) Web-Based Simulation, *Proceedings of the 1997 Winter Simulation Conference*, ed. A. Andradottir, K. J. Healy, D. H. Withers, and B. L. Nelson. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey

Halpin, D. W. (1977). CYCLONE: Method for Modeling of Job Site Processes, *Journal of the Construction Division*, ASCE, 103(3):489-499

Halpin, D. W. (1990). *MicroCYCLONE User's Manual*, Division of Construction Engineering and Management, Purdue University. West Lafayette, Indiana.

Halpin, D. W. and Riggs L.S. (1992) *Planning and Analysis of Construction Operations*. Wiley Interscience. New York, N.Y.

Healy, K. J. and Kilgore, R. A. (1997). Silk[TM]: A Java-Based Process Simulation language. *Proceedings of the 1997 Winter Simulation Conference*, ed. A. Andradottir, K. J. Healy, D. H. Withers, and B. L. Nelson. Institute of Electrical and Electronic sEngineers, Piscataway, New Jersey

Howell, F. and McNab, R. (1998) Simjava: a discrete event simulation package for Java with applications in computer systems modeling, *First International Conference on Web-based Modeling and Simulation*, San Diego CA, Society for Computer Simulation.

Ibbs, C. W. (1986) Future directions for computerized construction research, *Journal of Construction Engineering and Management*, ASCE, Vol. 112 No 3, p.p. 326-345.

Jade Simulations International (1992) *Sim++ user manual*, Jade Simulations International Corporation.

Kilgore, R. A., Healy, K. J. and Kleindorfer, G. B. (1998) The Future of Java-based Simulation. *1998 Winter Simulation Conference Proceedings*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson, M. S. Manivannan. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Little, M. C. (1997) "JavaSim World Wide Web Home Page" http://javasim.ncl.ac.uk/.

Liu L.Y., and Ioannou P.G., (1992). Graphical Object-Oriented Simulation System for Construction Process Modeling, *Proceedings of the Eighth Conference on Computing in Civil Engineering*, ASCE, Dallas, Texas. 1139-1146.

Nair, R. S., Miller, J. A. and Zhang, Zhang, Z. (1997) Java-Based Query Driven Simulation Environment, *1996 Winter Simulation Conference Proceedings*, 786-793. J.M. Charnes, D.M. Morrice, D.T Brunner, and J.J. Swain. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Odeh A.M., Tommelein I.D., and Carr R. I. (1992). Knowledge-Based Simulation of Construction Plans, *Proceedings of the Eighth Conference on Computing in Civil Engineering*, ASCE, Dallas, Texas. 1042-1049.

Page E. H., Moose, R. L., and Griffin, S. P. (1997) Web-based Simulation in Simjava using Remote Method Invocation, *Proceedings of the 1997 Winter Simulation Conference*, ed. A. Andradottir, K. J. Healy, D. H. Withers, and B. L. Nelson. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Powersim (1998) *Metro JX Sever Documentation*, Powersim Corporation, Herndon, VA.

Roberts, S.D. and Heim J. (1988) A Perspective on Object-oriented Simulation. *Proceedings of the 1988 Winter Simulation Conference,* ed. M.A. Abrams, P.L. Haigh, and J.C. Comfort. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey. 277-281.

Rothenberg J. (1986) Object-oriented Simulation: Where do we go from here? *Proceedings of the 1986 Winter Simulation Conference*, ed. J.R. Wilson, J.O. Henricksen, and S.D. Roberts. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Sawhney, A and AbouRizk S. M. (1993) An overview of construction simulation and its applications, *Journal of Construction Management, National Institute of Construction Management and Research*, India, Volume VIII, Number II, July 1993, 93-100.

Sawhney, A. and Mund, A (1998) Simulation based Construction Management Learning System, *1998 Winter Simulation Conference Proceedings*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson, M. S. Manivannan. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Sawhney, A., AbouRizk, S. M. and Halpin D. W. (1998) Construction Project Simulation using CYCLONE, *Canadian Journal of Civil Engineering, Canadian Society of Civil Engineering*, Volume 25 Number 1, 16-25.

Sawhney, A., Abudayyeh, O., and Chaitavatputtiporn, T. (1999) Modeling and Analysis of a Concrete Production Plant using Colored Petri Nets. *Journal of Computing in Civil Engineering*, American Society of Civil Engineers (ASCE), To Appear in July 1999 Issue.

Sun Microsystems, Inc. (1997) "JavaBeans Concepts" http://java.sun.com/beans/docs/javaBeansTutorial-Nov97/javabeans/index.html.

AUTHOR BIOGRAPHIES

ANIL SAWHNEY received his bachelor of Engineering degree from India in 1987 and a Master of Building Engineering and Management degree from school of Planning and Architecture, New Delhi in 1990. He completed his Ph.D. studies at the University of Alberta in June 1994. He is currently working as an Associate Professor in the Del E Webb School of Construction at Arizona State University. His research interests are mainly focused on construction simulation techniques and use of computers in construction education.

JAYACHANDRAN MANICKAM received his Bachelor of Engineering degree from the University of Madras in India in 1997. Currently, he is pursuing his Master of Science studies in Computer Science at Western Michigan University. He has been working as a research assistant in the Department of Construction Engineering and Management since his arrival at Western Michigan University in 1998. His research interests centers on Java simulation modeling and software development.

ANDRE MUND received his Bachelor of Engineering degree from UAL in Portugal in 1994. Prior to beginning his Master of Science studies in Construction Management at Western Michigan University in 1997 he worked for a contractor in Berlin, Germany, form 1994-1997. Since his arrival at Western Michigan University he has been working first as a teaching and now as a research assistant in the Construction Management Program. He is interested in the area of heavy construction equipment selection and computing and information technology applications in construction.

JENNIFER MARBLE is an undergraduate student in the Construction Engineering and Management program at Western Michigan University (WMU). She completed her first two years of study at the Kalamazoo Valley Community College (KVCC) and received a WMU Transfer Student Scholarship based on her performance at KVCC. Currently, she is also working as an Undergraduate Research Assistant on a project funded by the National Science Foundation as a part of the Research Experiences for Undergraduate's Program.