[1]A New Approach to Simulation of Heavy Construction Operations

Kyong J. Kim, Senior Researcher, Korea Institute of Construction Technology
G. Edward Gibson, Jr., Associate Professor, University of Texas at Austin

Abstract

To support the utilization of discrete-event simulation in the analysis and planning of heavy construction operations, this research investigation has established a framework for a more simplified simulation environment. This framework is a resource operation cycle-based simulation modeling approach using an object-oriented information system to support project managers. This approach allows for reusable simulation modeling elements that build various resource (operation cycle) models. The prototype system, KMOS (Knowledge-embedded, MOdularized Simulation system), provides a modularized resource model-building environment, step-by-step guidance to the user, flexibility in modeling, and easy control of complex construction operations. It includes an information system which provides guidance in the overall simulation procedure based on embedded knowledge. This paper will outline the major components of KMOS, as well as results of simulations performed using stochastic data captured from concrete paving operations.

1. Introduction

Although discrete event simulation has a large potential as a management tool for construction projects, the application of simulation has been limited in the construction industry. The major reasons for difficulties stem from the complexity inherent in the model-building process required for computer simulation and the problems with attaining meaningful simulation-related data. In order to provide a more user-friendly environment for simulation model building and experimentation, this research investigated a resource operation, cycle-based modeling approach and an object-oriented information system.

2. Review of Previous Modeling Approaches for Discrete Event Simulation

Most discrete event simulation systems utilize one of three major modeling approaches which are the event-oriented, the process-oriented, and the object-oriented. In the event-oriented modeling approach, a system is modeled by events that change the status of the system and the logical relationship among those events (Shannon 1975, Law and Kelton 1991). SIGMA (Schruben 1992) embodies the event-oriented simulation modeling approach into a graphical user interface. This simulation modeling approach provides flexibility in simulation model building, but requires considerable time with which to gain expertise and proficiency in the simulation model building process.

SLAM-II (Pritsker 1986), SIMAN (Pegden, Shannon, and Sadowski 1990), and MICRO-CYCLONE (Halpin 1977, Halpin and Riggs 1992, Huang. and Halpin 1993) utilize the process-oriented modeling approach. The process-oriented modeling approach generalizes and pre-defines the logic associated with a sequence of events into the basic types of model building elements (Pritsker 1986). As a result, simplicity in simulation model building has been derived (Pritsker 1986, Oloufa 1991 and 1993). However, the event-oriented and process-oriented modeling approaches do not allow for a simulation model to be broken down into sub-models of system components (each independent construction resource model). As a result, the model building

[1] Point of contact: (011-82-3651)847-3447, kjkim@smtppc.kict.re.kr, or kyongju@mail.utexas.edu

for each construction resource can not be isolated from the model building of other resources. This results in complicating simulation model building process when modeling complex construction operations.

As object-oriented simulation languages, Simula, MODSIM, Sim++, and Smalltalk-80 provide simulation facilities and the ability to construct necessary objects to describe an operation (Bischak and Roberts 1991). In an object-oriented simulation modeling approach, a system is described by a set of objects which interact with each other. This has great intuitive appeal in that it conforms to the notion that the real world is composed of "objects" (Booch 1994). However, the description of objects and their interaction is the responsibility of the user. This requires the user to be a language designer. This is a formidable barrier to the utilization of this approach as a commercial simulation system. The combination of simulation and artificial intelligence in object functions is suggested for more user friendly environments (Bischak and Roberts, 1991).

3. Resource operation cycle-based simulation modeling approach

To provide more user-friendly simulation model building environments, this research investigated new modeling approach. In the new modeling approach, in order to reduce the complexity of an construction operation system and to allow modularized model building, construction operations are decomposed into all involved resources and then each resource model is represented independently as a self-controlling object. So far, this is same as the object-oriented simulation modeling. However, to support the description of each resource model, the behavior of construction resources in their operation cycle is generalized into basic types of states at which construction resources will stay during their operation. The basic types of states embed knowledge about the construction resource behavior that corresponds to the operation structure. According to a given operation structure, they duplicate the behavior of construction resources in the simulation.

4. Modeling environments and benefits

This resource operation cycle-based modeling approach provides a more simplified modeling environment. The basic types of states make up various and independent resource models. The independently developed resource models are integrated to build a simulation model according to a given operation structure.

This modeling approach provides a more intuitive approach in building simulation models and also makes modularized resource model building possible. The modularized resource model-building, in turn, makes step-by-step guidance to the project manager in the overall simulation procedure possible. In addition, this modeling approach provides a base for the design of a resource model library from which the project manager can collect the appropriate resource models to build a simulation model, and reuse the models at a later date.

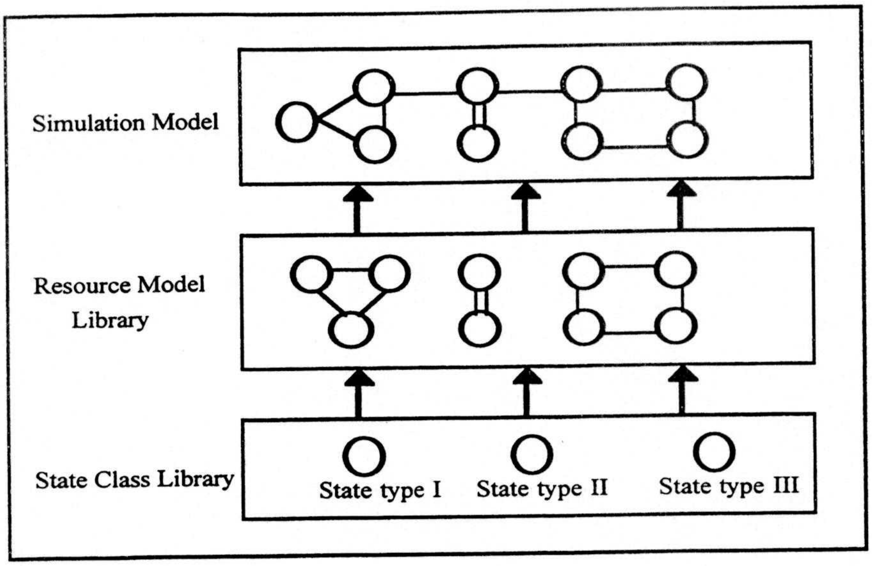

Figure 1. Hierarchic structure of multiple levels of construction operation

5. The characteristics of construction operations

To identify the basic types of states, the characteristics of construction operations were analyzed. If we define a construction process as a system handling several construction material, construction process can be represented as a hierarchical structure composed of operation level, resource level, and state class level (Figure 1). Here, the construction operations are defined as sub-systems controlling one material. In the process level, its structure can be represented by the sequence of handling several material.

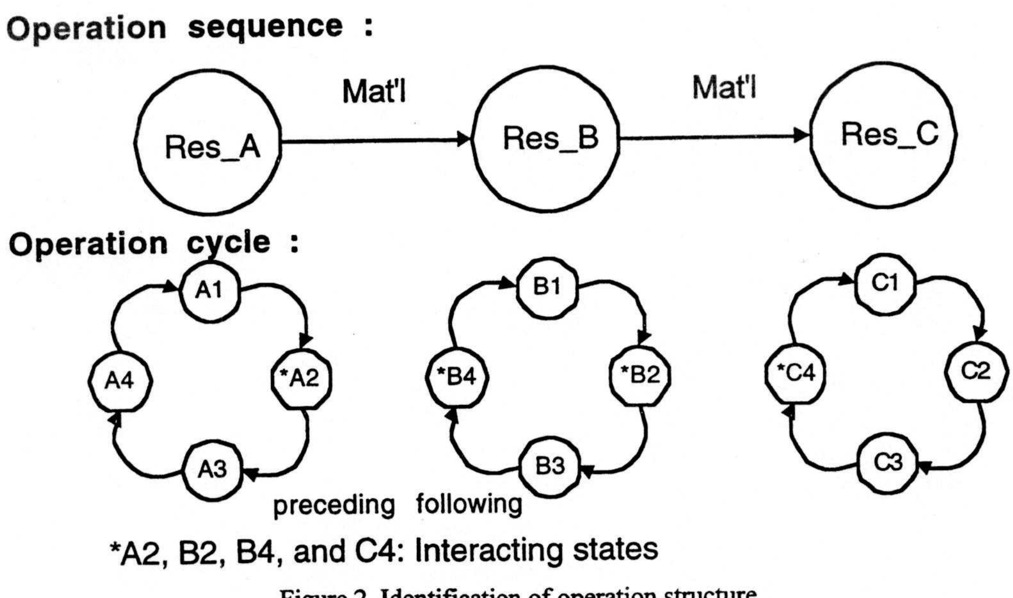

Figure 2. Identification of operation structure

In the operation level, construction material passes sequentially and repetitively through all involved resources. As a result, construction operation structure can be represented sequentially along the path of the material. Identification of all sequentially involved resource formulates an operation structure. For the design of resource model as a self-controlling object, the resource model is composed of states behaving by itself and interacting with other resource. If the interacting states are classified and designed to enable the communication between interacting resource models by referencing its operation sequence, simulation model can be built by the repetitive identification of all involved resource models and higher level operation structure (Figure 2). New simulation modeling approach is based on this concept.

To support this concept, interacting states are identified as follows. The basic types of states at which each construction resource can stay are the idling state and the working (busy) state. The working state can be classified into the working alone state and the interacting state. At the interacting state, material passes from one resource to another sequentially. Therefore, one resource provides service (active) and the other resource receives service (passive). One resource sequentially precedes the other resource (and the other resource sequentially follows). Therefore, the interacting state can be categorized into preceding active, preceding passive, following active, and following passive states. According to the inherent characteristics of the states, the preceding active (passive) state can interact only with the following passive (active) state. By utilizing the characteristics of the interacting states, each interacting state of a resource model can identify its corresponding state of its interacting resource model.

As a result, the whole life cycle of each resource can be represented by queuing (idling), active alone, preceding active, preceding passive, following active, and following passive state. In integrating several operation cycle models (resource models) to build a simulation model, the interacting state can be used as a connection point between resource models. A simulation model for an earthmoving operation performed by shovel and truck can be represented by Figure A1.

6. A framework for an information system

In order to utilize and support resource model-based simulation modeling, and to provide step-by-step guidance in the overall simulation procedure, this research developed a framework for an information system and established a prototype system. The identified framework includes several system modules:

1) Knowledge Processing Module. The knowledge processing module contains common knowledge about the overall simulation procedure by the new modeling approach; it guides the project manager in inputting information required for operation structure description and simulation experimentation. Keyed by the user's input, the processing module formulates a simulation model, executes the simulation, and then provides output by utilizing the operation board, state class library, resource database, and the supporting class library.
2) Operation Board. To support the behavior of each state class, The operation board stores information about the static operation structure and the dynamic status of the operation. This information describes the behavior of the operation and composes a simulation model for the operation. The static operation structure refers to the sequentially involved resources to perform the operation, and the operation characteristics and cycle model of each resource involved. This information is obtained through user input. The status of the resources involved changes as the operation proceeds. The information about the dynamic operation status includes the current operation status of each resource, the list of each resource waiting on each queue state, and the list of the scheduled state which will occur in the future. Such information about the dynamic operation status changes allows easy control of complex construction operations including resources composed of different capacities.
3) State class library. The *state class library* contains the basic types of state classes, which are model building elements. The basic types of state classes access the information in the operation board (static and dynamic operation structure). By utilizing the information on the operation board, each state

identifies its interacting resources, interacting states, preceding state, and following state. According to the operation structure, it duplicate the behavior of the construction resource that corresponds to the operation structure. The state classes are designed to help each resource model behave as a self-controlling object.

4) Resource Database. The resource database is an object-oriented database containing data and functions to manipulate the data. The resource database contains pertinent information on equipment performance, working capacity, proper operation conditions, proper capacity balance, maintenance records, etc. The database is queried by the knowledge processing module to support proper equipment selection and data preparation.
5) User Interface. A user interface was designed to support user input and to allow project managers to respond interactively to the guidance of the knowledge-processing module.
6) Supporting Class Library. Simulation supporting classes were developed to provide many functions that support simulation experimentation.

All of these functions were established in a prototype system.

This prototype system was developed to provide a framework for more simplified discrete event simulation in heavy construction operations. It is called Knowledge-embedded, MOdularized Simulation system (KMOS). The input and output of the prototype system are as follows.

6.1 Input to the Prototype System

In order to build a simulation model, KMOS needs extensive input from the project manager in regard to a description about the construction operations. The description includes the sequential list of resources involved in the operation, and their characteristics and behavior in the operation. According to the resource type, necessary data about the characteristics of the resource are different. When the resource type is equipment, the characteristics include the number of pieces of equipment, combination of capacities, and the working capacity of each resource. The number of pieces of equipment will be the initial number of equipment waiting on a queue state at an operation's starting point. In the case of space resources, the maximum number of pieces of equipment which can work in a space at the same time is a characteristic representing its buffer size. The maximum capacity of the space which can store material in the space can be another constraint of the resource.

The behavior of each resource in the operation is represented by the operation cycle. The project manager needs to select or input the basic types of states which make up the operation cycle of the resource. These input data compose a simulation model for a given operation. In order to execute the simulation model, the system requires additional information, such as a random number seed and the number of simulation run to be repeated.

6.2 Output of the Prototype System

In order to provide more comprehensible simulation results, the simulation output gives the total simulation time to finish an operation, idle time of each resource at each queue state, and the efficiency of each resource. The idle time at each queue state of a type of resource helps to identify the bottlenecks of the operation. From this information, the project manager can perform time and cost analysis under many different conditions.

6.3 Development environments

The overall prototype system was developed with a Borland® C++ version 4.01 which provides an integrated development environment for object oriented system development. In particular, user interfaces, knowledge processing modules, operation board, state class library, and supporting class library are implemented in this environment. For the design of resource database, POET™ was utilized which is an Object-Oriented Data Base Management System (OODBMS). An object oriented database combines the semantics of an object oriented programming language with the data management and query facilities of a conventional database system.

7. Application

KMOS is applied and evaluated using stochastic data obtained from paving operations at Austin-Bergstrom International Airport. In order to compare the simulation output with the actual performance of the operation in the real project, videotape was taken for total 4 hours on the concrete pavement operations, and then the videotapes were analyzed using time and motion study techniques to determine actual site productivity of the operation. Table 1 shows the comparison between the real efficiency of each resource and the simulation output on the pavement operations from the prototype system. To get reliable simulation data, multiple numbers of simulation runs were executed with various random number seeds in the prototype system. From this comparison in Table1, simulation results are able to approximately reflect the real world operations.

From this comparison in Table 1, simulation results are able to approximately reflect the real world operations. However, there is some discrepancy in the efficiency of the trucks. This difference comes from the fact that, in real world operations, trucks experience random states in their operation cycles such as a state of washing concrete from the truck before arriving at the concrete batch plant. This state is not a constant state in its operation cycle. Therefore, the prototype system needs to include this type of random state in its state class library.

Table 1. Comparison of real data and simulation output

	Efficiency of concrete batch plant	Efficiency of concrete distributor	Efficiency of trucks	Total operation time for the section
Real operation data	72	64	95.2	4 hr 15 min
Simulation results	78.2	64.49	93.73	3 hr 58 min

8. Conclusions

The major objective of this research has been to establish a more simplified environment for the utilization of discrete event simulation in the design and planning of heavy construction operations. In order to meet this objective, a resource operation cycle-based simulation modeling approach was proposed and an object-oriented information system was established to support the project managers. This new modeling approach provides more chances for the integration of simulation functions and information system which will provide more user friendly simulation environments. As a result, this new approach can be a viable tool to support the utilization of discrete event simulation in heavy construction operations. However, the developed system is a prototype. In order to reflect the construction operations more accurately and to provide more flexibility in modeling, this approach needs some improvement for resource sharing problems, including more parameters impacting on the construction operation performance (job efficiency).

REFERENCE

1. Bischak, D. P. and Roberts, S. D. (1991). "Object-oriented simulation." Proc. 1991 WinterSimulation Conf., 194-203.
2. Bernold, L. E. and Halpin, D. W. (1986). "Advanced microcomputer simulation for construction managers." Proc., 4th Conf. on Computing in Civ. Engrg., ASCE, New York, N. Y., 762-772.
3. Booch, G. (1994). *Object-oriented Analysis and Design with Applications*. Rational, Santa Clara, CA.
4. Halpin, D. W. (1977). "CYCLONE - method for modeling job site processes." *J. Constr. Div.*, ASCE, 103(3), 489-499.
5. Halpin, D.W., and Riggs, L.S. (1992). *Planning and analysis of construction operations*. John Wiley & Sons, Inc., New York, N.Y.
6. Huang, R. and Halpin, D. W. (1993), "Dynamic Interface Simulation for Construction Operations (DISCO)." proceedings of ISARC, Houston.
7. Law, A.M. and Kelton, W.D. (1991), *Simulation Modeling and Analysis*, Second Edition. Mcgraw-Hill, Inc., New York.
8. Oloufa, A. A. (1991), "Intuitive simulation modeling using object oriented constructs." preceedings of ISARC, Stuttgart, Germany, 727-736.

9. Oloufa, A. A.(1993), "Modeling of building construction activities using forms." proceedings of ISARC, Houston, 237-244.
10. Paulson, B. C. Jr. (1978), "Interactive graphics for simulating construction operations." *J. Constr. Div.*, ASCE, 104(1), 69-75.
11. Pegden, C. D., R. E. Shannon, and R.P. Sadowski. (1990), *Introduction to simulation using SIMAN*. McGraw-Hill, New York, NY.
12. POET - User's Guide. Version 2.1. (1994). POET Software Corporation, Santa Clara, CA.
13. Pritsker, A.A.B. (1986), *Introduction to simulation and SLAM II*, Systems Publishing Corp., West Lafayette, Indiana.
14. Schruben, L.W.(1992). *SIGMA: A graphical simulation modeling program,* The Scientific Press., South San Fransco, CA.
15. Shannon, R. E. (1975). *Systems Simulation: The Art and Science*, Prentice-Hall, Englewood Cliffs, N. J.

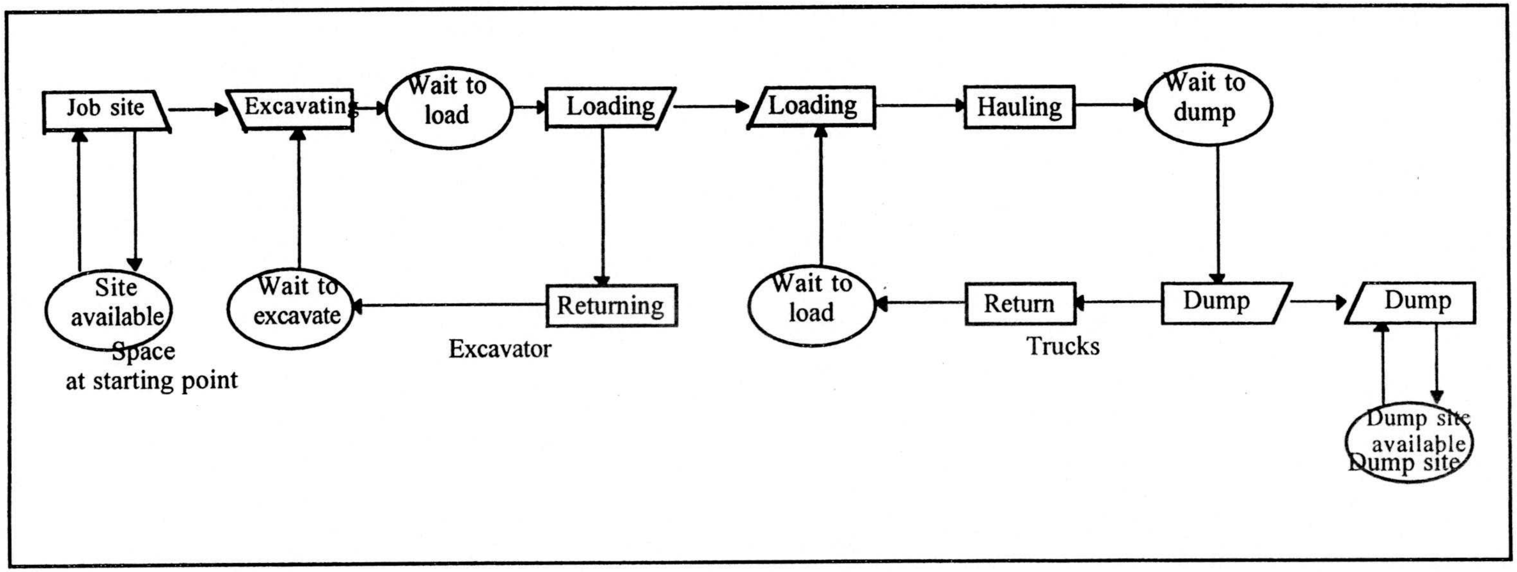

Figure A1 Simulation model for earthmoving by space resources, excavator and trucks