First International Conference on Construction in the 21st Century (CITC2002)
"Challenges and Opportunities in Management and Technology"
25-26 April, 2002, Miami, Florida, USA

MODELING EARTH MOVING WITH COMPONENT-STATE-BASED CRITERIA METHOD

D. K. H. Chua
Associate Professor, Department of Civil Engineering,
National University of Singapore, Singapore

B. Tan
Research Scholar, Department of Civil Engineering,
National University of Singapore, Singapore

J. K. F. Lee
Final Year Undergraduate, Department of Civil Engineering,
National University of Singapore, Singapore

ABSTRACT

This paper presents a component-state-based criteria model for construction simulation that employs a modeling methodology from the object-oriented view. The method employs a criteria object to list and check component states and other special permits (e.g. weather permit) before initiating the interaction activity. Besides the interaction activity, the component object has its inherent state changing method. This approach also employs both a components-relation diagram and the process flow diagram to facilitate the model development. Finally, the methodology of the component-state-based criteria model for simulation and its potential are illustrated with examples of earth moving applications.

KEYWORDS

Construction, Simulation, Object-oriented, Resources, Discrete-event

1. INTRODUCTION

Discrete-event simulation is an accepted approach to study and analyze construction operations. The modeling methods may be categorized into process-oriented and object-oriented in terms of strategies. CYCLONE (Halpin 1977) and STROBOSCOPE (Martinez 1996) are among the process-oriented simulation systems that have been developed specially for construction. The modeling paradigm of these systems is based on the Three-Phase Activity Scanning (Martinez and Ioannou, 1999). Following this methodology, the modeler focuses on identifying the activities, the conditions under which the activities can happen, and the outcomes of the activities when they end (Martinez 1998). Queues are employed to represent preconditions for its assigned resources. It suggests availability of the resources prior to the occurrence of an event. However, the modification and management of these queues become a tedious work when the preconditions are complex. Furthermore, the preconditions of activity usually include many more factors other than the availability of the resource. Examples are physical relations, functional dependency, construction space occupation, resource allocation, productivity, safety, weather, contract stipulation and government regulation (Chua and Song, 2001). Although some of these factors are not the essential elements in a simulation system, others such as physical relations of the resource components, functional dependency and weather, have an important effect on simulation modeling.

In contrast, some other systems present the model of the construction operations from the object-oriented point of view. COOPS (Liu, 1992) is one of the earliest construction simulation systems, which models the interaction of components and its component parts with the advantage of object-oriented programming to improve its presentation. This object-oriented modeling has the advantage such that model modification on each object can be done without affecting other objects or the entire system. Animation environments are incorporated with COOPS-R (Liu, 1993) and ACPSS (Liu, 1996). Another early approach is a library-based simulation modeling developed by Oloufa and Ikeda (Oloufa and Ikeda, 1997), in which a library of pre-programmed construction resources is developed and targeted at a specific category of project. Manavachi (Manavachi, 2000) provides a simulation modeling approach from the inter-component and intra-component view to model the dependencies of components and structural relations among component parts. These works have employed the advantages of object-oriented programming to facilitate the presentation of both the interaction of components and that of component parts. However, little emphasis is put on how to express the relation of the component and the relation of the component parts. The main part of these relations is the various preconditions of these interactions.

This paper presents a component-state-based criteria model for construction simulation. It employs a criteria object to list and check component states before initiating the interaction activity. The criteria object enhances model representation at the resource level abstraction. It also groups the preconditions for each interaction into its respective criteria object location so that any modification to the preconditions can be easily implemented. Besides the interaction method, the component object has inherent state changing methods. A component-relation diagram and the process flow diagram are used to facilitate model development.

2. COMPONENT-STATE-BASED CRITERIA MODELING

The model comprises four basic modeling elements: the complex component object, the simple component object, the criteria object and the arrow object as in Fig 1.

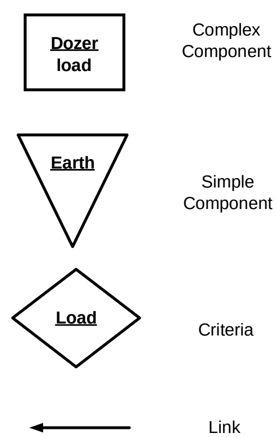

Figure 1: Basic Modeling Element

2.1 The Component Structure

Previous works mainly focused on resource component such as labor, equipment, material, time and space (Liu, 1992). Generally, the resources can be classified into two groups as Simple Resource or Complex Resource (Chua and Li, 2001) depending on whether they have state changing methods or not. The term component has been adopted instead to include the noting of all resources and building elements (such as trench or beam support etc.) those are relevant to the component operation. Similar to resources, the Component in the model is categorized into two groups: the Complex Component and the Simple Component. The information contained in each component is comprised of three important parts: (1) component attribute, (2) component type, and (3) component state information as in Fig 2.

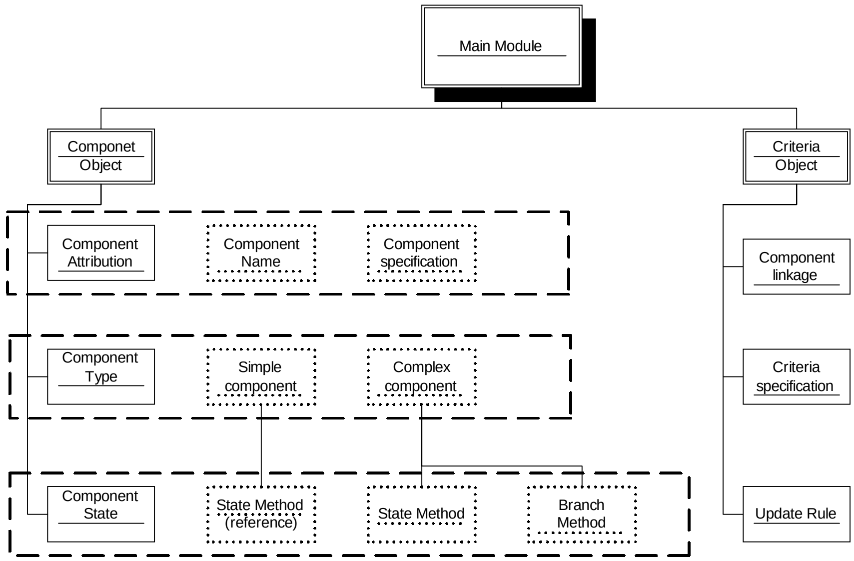

Figure 2: Information of Integrated Component

Component attribute

One of the attributes of a component is its name. This provides the link to the respective component in the component library. This would allow it to inherit the sets of component states in the library. The other attributes are the specifications of the components. Examples can be "amount of earth" for a truck, or "capacity" for a dozer. The value of these specifications may differ in the simulation program for each sub-component in the same component set.

Component type: Simple Component and Complex Component

Component type is the index for their different data structures and component functions. It distinguishes a complex component from a simple component according to the manner of state change. The complex component, such as truck or dozer, has its own state changing method (activity), while simple component such as "Earth" changes its states by referring to another complex component's states cycles. .

Component state

The states of a component represent all the possible status of objects in the construction operation. The states in both complex and simple components may form either a cycle-states chain or a line-states chain. The cycle-states chain depicts a component which begins and terminates at the same state. This is common in components such as truck, spotter and so on. Truck Component, for example, has a states chain: Load → Haul→ Offload→ Inspection →Load as depicted in Fig 3, whereas typical states chain for "Trench" component can be: Excavate → Prepare_trench → Lay_pipe → Backfill → Close_up. In this case, the states chain begins and terminates at different states, illustrated in Fig 3.

Associated with each state in a complex component is a state-changing method. Physically, the method corresponds to an activity in the operations. Thus, in the truck examples of Fig 3, the state "Haul" has a state changing method to transit the truck state to "offload" after it has finished the hauling activity. The pre-conditions belonging to a state-changing method distinguish it either as an inherent state-changing method or an interaction state-changing method. The inherent state-changing method will start its corresponding activity as soon as the component transits to the next state. The methodology of interaction state-changing method is different. Its corresponding activity will not start until its pre-conditions in the associated Criteria object are satisfied. In both methods, after the activity is triggered, the component will not change to the next adjoining state until the duration of the activity is passed.

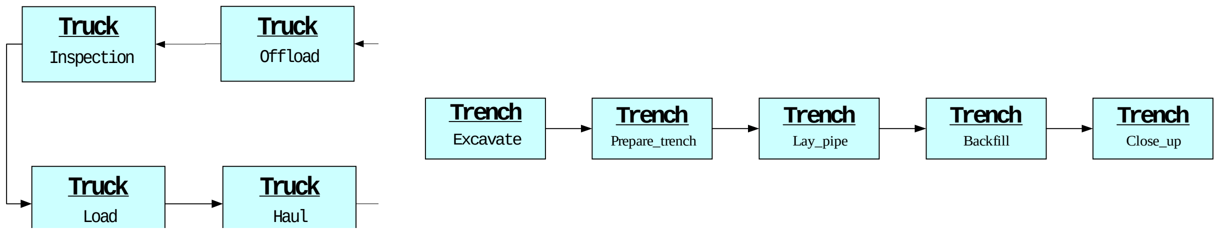

Figure 3: Cycle States and Line States Chain

The simple component does not have its own state-changing method. Instead, it makes reference to the associated complex component for state change as in the earth-moving problem. When the Truck Component changes from "Haul" to the "Offload" state, the state of the Earth Component is changed from "Haul" to "Offload" at the same time. When the states of the simple components are not important considerations in the simulation, simple components can operate solely with its attribute without the state changing.

There is another method associated with complex components, namely, the branch method. It provides the mechanism for the component to transit from its current state to one of several possible following states. This is facilitated by the probability assigned to each of the possible state transition paths.

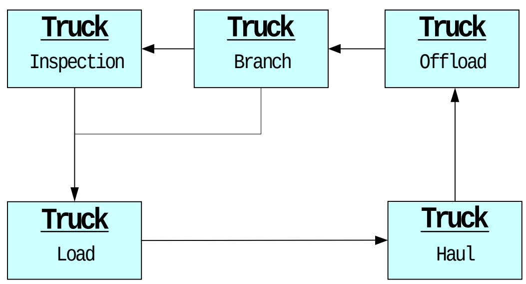

Figure 4: Truck States Chain with Branch Method

For example, in Fig 4, the "Inspection" state is only necessary for truck that requires attention. A "Branch" method is employed to provide the probabilistic branching. The probabilities are specified in the branch rule, or they can be specified as functions of the state attributes of the component.

2.2 Criteria Object

The distinctive feature of the proposed modeling approach is the Criteria object, which manages the interaction activities of the components in the operation modeling. For example, the Criteria object "Offload" in Fig 5, show the interaction of 2 components, Truck and Spotter, involved in the offloading process. The Criteria object contain the pre-condition that are necessary for the "Offload" method in the corresponding complex component (i.e. truck and spotter). The "Offload" activity can only be triggered when these conditions are satisfied. Otherwise, these components will remain in an inactive condition in those states.

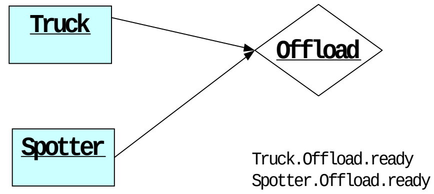

Figure 5: Criteria Object "Offload"

The criteria in this example can be simply specified as "Truck.Offload.ready AND Spotter.Offload.ready". These components must be in the inactive condition in the "Offload" states, which is specified as "ready to offload" in the Criteria Specification. Fig 6 shows the Criteria Specification dialog box for "Offload".

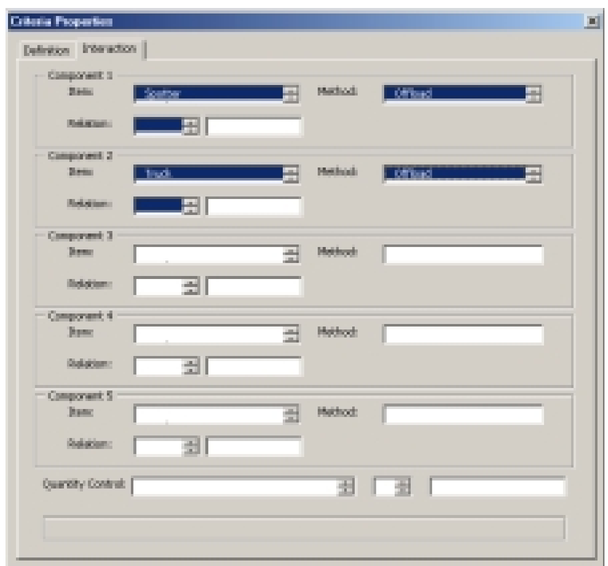

Figure 6: Criteria Specification Dialog Box for the "Offload"

3. MODELING OF AN EXAMPLE

The Component-State-Based criteria modeling approach allows the modeler to begin construction of the simulation model from the process diagram of the component: their respective state chains. These process flow diagrams provide a detail abstraction of the model. In this level, the flow of the component states is clearly presented. On the other hand, the component-relation level diagram is the higher lever abstraction above the process flow level diagram. It provides the overview of the simulation model exhibiting interaction of the components using Criteria objects. The criteria object provides the means to manage the interactions of the components at the component relation level. The two-level presentation facilitates model development and modification. The detailed process level of the component may be modified without affecting the inter-component simulation logic at the component relation abstraction level. Fig 6-a shows a typical model earth-moving example using a process flow simulation type approach. The component relation level abstraction provides a simplified model representation of the same interaction logic in Fig 6-b.

In the Fig 6-b, Truck Component object represents the group of all trucks with the inherited "capacity" attribute and same states-chain cycle. Sub-components are the next level in the object hierarchy to specify trucks with the same attributes type but different attribute values. For example, five 20T capacity and seven 15T capacity trucks involved in the same earthmoving operation can be grouped into sub-components "20T Truck" Component and "15T Truck"

Component under the same "Truck" Component. Each of these 12 trucks is treated as an entity in the model, which can be tracked separately.

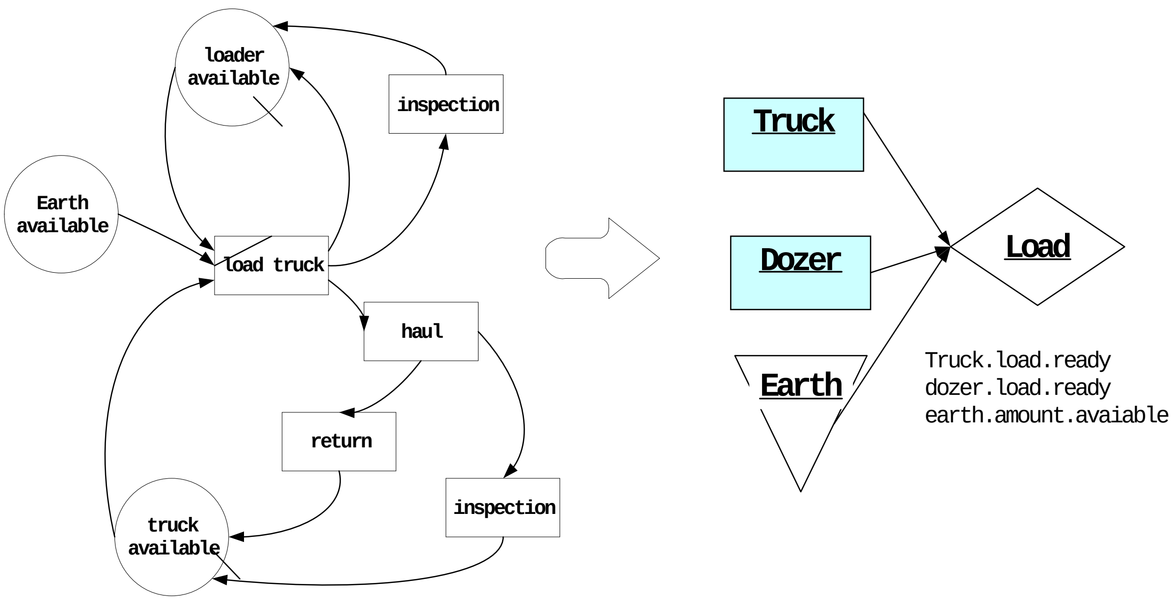

Figure 6-a and 6-b: Transformation of Earth Moving Example Model

4. SAMPLE APPLICATION

In this Earth-moving operation, three components are involved. "Truck" and "Dozer" are complex component, and "Earth" is a simple component. At the process level, the inherent cycle for "Truck" is : Load → Haul → Return → Branch → Inspection and Load, while that for "Dozer" is: Load→ Branch → Inspection and Load. The process level diagrams for each component are depicted in Fig 7.

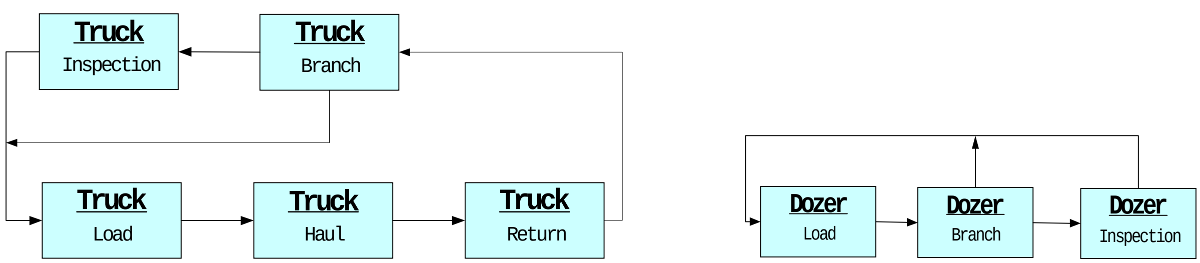

Figure 7: Process Flow Diagram of the Truck and Dozer

The interaction of there components only occurs in activity, "Load". This is controlled by the "Load" Criteria object. This load process requires a "Dozer" to participate in a "Load" activity with an empty "Truck" at the site. The specification for the interaction Criteria is, "Truck.Load.ready AND Dozer.Load.ready AND Earth.amount.available". After loading, the "Truck" now carrying the earth will haul its cargo to the location and return to the site either for inspection or loading according to the probability assigned to the Branch as depicted in the dialog box of Fig 8. The "Dozer" follows a similar states cycle till the operation terminates when all the earth had been transported to the site.

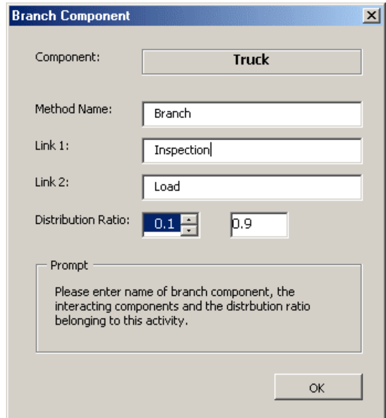

Figure 8: Branch Dialog Box for "Truck"

"Amount" is the single attribute belonging to the Earth Component. In this application, it is not essential to track its state, therefore, no reference state method is necessary. The application example, durations of each state method are specified in Table 1.

Table 1: State Method of Complex Components

Component	No. of Entities	State Method	Duration(mins)
Truck	7	Load	Exponential(3.0)
		Haul	Normal (45.0, 5.0)
		Return	Normal (40.0, 5.0)
		Inspection	UniformReal (10.0, 90.0)
Dozer	2	Load	Exponential(3.0)
		Inspection	UniformReal(20.0,100.0)

The model can be tracked at the entity level as demonstrated in Table 2. Accordingly, performance statistics can be provided for each entity. In this example, the percentages of the time that the trucks and dozers are engaged had been calculated. The number of instances that each truck was served or dozer had served had also been tracked.

Table 2: Output Report of "Truck" and "Dozer" Complex Component

Input parameters for Multi truck / Multi dozer:											
Number of Trucks: 7					Number of Dozers: 2				Earth amount is 1000.00		
Simulation results :											
Stop time in hours: 9.62											
	Truck 1	Truck 2	Truck 3	Truck 4	Truck 5	Truck 6	Truck 7	Mean	Dozer 1	Dozer 2	Mean
Busy Percent	81.97%	84.43%	85.45%	85.16%	91.28%	91.16%	88.88%	86.91%	25.45%	49.84%	37.64%
Number Served	8	10	10	9	10	10	10	/	42	25	/

5. CONCLUSION

In this paper we described the Component-state-based criteria model for construction simulation, which is developed and presented at two abstraction levels. Component objects possess an inherent cycle that is built at the process level abstraction. These component attributes and associated states chain can be built from the component library or new ones may be added. The constructed components are employed at the component relation level to model the interaction of the component in the operation. The Criteria object makes the interaction explicit in the model. It also specifies the preconditions that ought to be satisfied before interaction can be triggered.

Following this methodology, presentations of inter-component and intra component relation are improved with the two abstraction level diagrams. Criteria object also facilitates presenting and further modification of the component relations. This paper had shown some examples utilizing this methodology.

A prototype and the template have been built based on this component-state-based criteria model. In this system, the users utilize the templates and the component library for model development. The templates are then translated into a program code compatible with MODSIM III®, which acts as the simulation engine.

REFERENCES

Chua, D.K.H., and Li, G.M. (2001). "Modeling construction operations with RISim." *Journal of Computing in Civil Engineering*, Vol. 15, No.4, pp. 320-329

Chua, D.K.H., and Song, Y.B. (2001). "Component state model and its application in constructability analysis of construction schedule" *Proceedings of the 8th International Conference on Civil Structural Engineering Computing,* pp. 23-24

Halpin, D.W. (1976). "CYCLONE - methods for modeling job site processes." *Journal of Construction Division*, Vol. 103, No. 3, pp. 489-499.

Halpin, D.W., and Woodhead, R.W. (1976). *Design of construction and process operations*. John Wiley & Sons, New York, USA.

Liu, L.Y. (1992). "Graphical object-oriented discrete-event simulation system ".*Proceeding of the 1992 Winter Simulation Conference*, Arlington, VA, USA, pp. 1285 - 1291.

Liu, L.Y. and Ioannou, P.G. (1993). "Graphical resource based object-oriented simulation for construction process planning ". *Computing in Civil and Building* Engineering, ASCE, Published by ASCE, New York, NY, USA, pp. 1390-1397.

Liu, L.Y. (1995). "Simulating construction operations of precast-concrete parking structures". *Proceeding of the 1995 Winter Simulation Conference*, Arlington, VA, USA, pp. 1004-1008.

Liu, L.Y. (1996). "ACPSS - animated construction process simulation system". *Computing in Civil Engineering*, New York, USA, pp. 397-403.

Manavazhi, M.R. (2000). "Simulating the construction of structures – an automated approach", *Simulation Practice and Theory*, Vol. 8, No. 3-4, pp. 161-176.

Martinez, J.C. (1996). "STROBOSCOPE: State and resource based simulation of construction processes." PhD dissertation, University of Michigan, Ann Arbor, Mich, USA.

Martinez, J.C. (1998). "EarthMover - Simulation tool for earthwork planning." *Proceeding of the 1998 Winter Simulation Conference*, Washington DC, USA, pp. 341-348.

Martinez, J.C. Ioannou P.G. (1999). "General-purpose systems for effective construction simulation." *Journal of Construction Engineering and Management*, Vol. 125, No. 4, pp. 590-606.

MODSIM III reference manual. (1997). CACI Products Company, La Jolla, CA.

Oloufa, A.A., and Ikeda, M. (1997). "Library-based simulation modeling in construction." *Computing in Civil Engineering*, pp.198-205.