Applying Multi Agents to General Purpose Situational Simulations in Construction Management

Amlan Mukherjee[1] and Eddy M. Rojas[2]

Abstract

Situational simulation environments are temporally dynamic clinical exercises that expose participants to rapidly unfolding events and the pressures of decision-making, both characteristic of the implementation phase of construction projects. The need for contextually rich education environments in construction suggests the development of a general-purpose situational simulation framework, which can be used by independent developers to build effective training environments. However, the absence of a simulation paradigm to effectively develop such training environments indicates the need for research in the area to develop a new framework that can support general-purpose situational simulations in construction. This paper argues that a multi-agent framework can be used to create a general-purpose situational simulation environment for the construction domain.

Introduction

Traditional classroom curriculum does not allow fledgling construction managers to experiment and explore different ways of dealing with situations that unfold during the implementation phase of multi-faceted construction projects. McCabe et al. (2000) argue that current civil engineering coursework teaches only the theories of construction management and that students may encounter difficulties in applying theoretical principles when exposed to real world situations upon employment. Sawhney et al. (2001) state that civil and construction engineering curricula does not allow the inclusion of issues of importance to construction, or the significance of hands-on experience and interaction with practitioners.

In traditional construction education the learner and the learning context are detached. Concepts are presented as fixed, well-structured, independent entities and classroom activities are disconnected from authentic context resulting in fragmentation and specialization of courses and educational experiences.

Simulations in Construction Management

In a field such as construction management where problems present multi-faceted situations dependent on context, a traditional approach to learning is not fulfilling.

---

[1] Doctoral candidate, Department of Civil and Environmental Engineering, University of Washington, Seattle, WA 98195 amlan@hitl.washington.edu

[2] Assistant Professor, Department of Construction Management, University of Washington, Seattle, WA 98195 er@u.washington.edu

This understanding has led researchers to explore alternatives in construction education using gaming and simulation environments such as Superbid (AbouRizk 1993), STRATEGY (McCabe et al. 2000), ICMLS (Sawhney et al. 2001) and VIRCON (Jaafari et al. 2001). Some of these efforts have been inspired by earlier research undertakings in the area, such as CONSTRUCTO (Halpin 1970) and AROUSAL (Ndekugri and Lansley 1992). These efforts provided a stepping-stone towards creating participatory, contextually rich educational environments.

A survey of simulations in construction engineering and management suggests that we can classify them using three different approaches. The first approach classifies simulations based on whether they are simulating construction management processes or construction operations. While, Superbid (AbouRizk 1993), STRATEGY (McCabe et al. 2000), ICMLS (Sawhney et al. 2001), CONSTRUCTO (Halpin 1970) and VIRCON (Jaafari et al. 2001) are all examples of simulations that deal with construction management processes, Simphony (Hajjar et al. 1999) and STROBOSCOPE (Martinez et al. 1999) are examples of simulations that deal with construction operations like tunneling and earthmoving. The second approach to classifying simulations is based on whether they are of a special purpose or a general purpose in nature. The difference between special purpose and general-purpose simulations are:

- Special purpose simulations are restricted in scope (to a particular operation like tunneling or a particular management process like bidding).
- General-purpose simulations, unlike special purpose simulations, allow for greater flexibility of scope since they are programmable.
- General-purpose simulations can be used to promote new simulations and collaborations amongst developers.

A survey of current research efforts indicates that there exist general purpose and special purpose simulation tools and techniques for simulating construction operations (Simphony: Hajjar et al. 1999 and STROBOSCOPE: Martinez et al. 1999). All the simulations in the area of construction management processes are special purpose in nature. They deal with specific problems in planning or bidding (AbouRizk 1993) or negotiation

The third approach to classifying simulations can be based on how interactive they are. We define situational simulations as temporally dynamic, interactive simulations. In their simplest form simulations of construction processes use a set of initial conditions and parameters, and a well-defined model to project outcomes regarding the simulated operation. For example, given information regarding the availability of trucks and loaders, their unit costs and the amount of earth to be moved a process simulation would be able to project the total time and cost for an excavation operation. Situational simulations also have a well-defined model and a set of initial conditions, but as the simulation proceeds the system generates random events and expects the user to react to such events. How the simulation evolves is completely dependent on the model used, the way the events are generated and user interaction.

Such an interactive simulation is very useful in developing learning environments in which the participant is capable of exploring 'what-if' scenarios involving construction management processes. For instance, in a situational simulation of an earth moving operation participants might be in a situation where they have to deal with finishing an operation within time and budget constraints, under the influence of bad weather and a labor strike. The participants' ability to interact with the environment and effect changes in it exposes them to clinical exercises that test their decision-making skills in critical scenarios. Most of the surveyed construction simulations have very limited user interactivity. ICMLS and STRATEGY allow user interaction and are both designed for educational purposes. However, they lack programmability and are special purpose in nature. When implemented through virtual environments, situational simulations can provide participatory and contextually rich educational environments. A general purpose framework for situational simulations dealing with construction management processes could be useful for developing a very wide variety of training environments for the construction engineering and management domain.

Simulation languages like STROBOSCOPE and CYCLONE have for a long time provided a general and special purpose framework for simulating construction operations and construction management processes, with absent or limited interactivity. They are based on the Activity Scanning simulation paradigm, which treats activities and events as time points, and do not include any temporal reasoning. The temporally dynamic nature and the interactivity of situational simulations demand a representation, which can express multiple parallel events in time effectively. Existing simulation paradigms and languages in construction may not be appropriate for developing a general-purpose framework for situational simulations.

The absence of an existing simulation paradigm to effectively develop general-purpose situational simulation environments for the construction domain indicates that there is scope for research in the area of developing such a framework. In this paper, we introduce the use of a multi-agent framework to develop a general-purpose situational simulation environment for the construction management domain.

A Multi-Agent Framework for General Purpose Situational Simulations

Background. Investigations in the use of agents in interactive environments like traffic simulators and battlefield simulation environments suggest the use of multi-agent frameworks. A multi-agent framework uses more than one autonomous agent in a collaborative environment to perform multiple goals (and sub-goals) in a distributed fashion. Tambe et al. (1995), have also explored the use of intelligent automated agents for simulation scenarios in air combat. Their environments are based on Distributed Interactive Simulation (DIS) technology; in which large scale interactive simulations are built from a set of independent simulators linked together by a network. They developed independent, intelligent automated pilots in the environment based on the underlying Soar integrated architecture for general intelligence (Laird et al. 1987).

Tambe et al. (1995) argue that finite state machine (FSM) languages are too restrictive to represent autonomous intelligence. Situational calculus, an FSM language, is inadequate for representing information about the parallel nature of events in the construction domain. Even though the situational calculus approach was used in the air-combat domain, parallelism and simulation of multiple fighter planes could be achieved through DIS technology. By running multiple copies of the same simulation, they were running multiple FSMs. The framework proposed in this paper runs multiple finite state machines (for each activity context) within a single simulation model, as explained in detail by Mukherjee and Rojas (2003).

In the absence of DIS technology, the agent framework introduced in this paper uses temporal reasoning based on an interval representation of time (Allen and Ferguson 1994) to represent parallel activities within the construction domain. This architecture ascribes operations to agents. However, the operations are not defined to create transitions between states. Instead, the agent operators change attribute values of entities, which are logical aggregates of variables. Each variable defines some aspect of the environment. The time interval reasoning allows the description of an aspect of the environment as an assertion about a variable attribute over a time interval. Different entities are affected at different times by different agent operations and at any time it is possible to have persistent states of variables or multiple operators acting on multiple entities each specific to a particular context or activity.

The M-RAM (Soibelman et al. 2000), a multi-reasoning model uses an agent-like approach to develop modules, each of which is specialized to perform particular tasks. The M-RAM model was used to support the conceptual phase of structural design and to study the applicability of agents to support the sub-processes of a divided structural design process. This paper looks at using agent modules, each of which is specialized to perform a particular thread of reasoning pertinent to the implementation phase of the construction project. The autonomous reasoning and problem solving capabilities of the agents allow us to efficiently design situational simulation environments for the construction domain.

Representing and Reasoning about Situational Simulations. The construction management domain can be abstracted to a planning problem during the implementation phase and a constraint satisfaction problem during the pre-construction phase. It involves satisfaction of resource and precedence constraints, and a causal reasoning, which, governs actions and events in the construction environment.

The situational simulation environment generates actions and events to emulate real life scenarios. Actions and events are dependent on the context of the simulated project and represent constraint violations in the environment. The participant's skills are tested by how well they can take corrective measures to satisfy such violated constraints.

The main threads of reasoning underlying the situational simulation system can be listed as: reasoning about actions and events in the environment and reasoning about the dynamics of the simulated system. Relationships between different aspects of the construction industry have been mathematically modeled in Rojas and Mukherjee (2003). Reasoning about actions and events is based on axiomatic semantics described in Mukherjee and Rojas (2003).

Actions are triggers, which create events and situations. Events reflect the effects of real life episodes on resource and precedence constraints within the construction domain. Each event is associated with three sets of variables: the *Pre-Condition* set, the *Event Condition* set and the *Consequence* set and is triggered by a unique action. Actions trigger events by changing the attribute values of variables in the *Pre-Condition* set to yield the *Event Condition* set. Consequences of an event are assertions about the future, and are stored in the *Consequence* set. Information about actions and events is stored in a knowledge base and is based on the event and action definitions.

Agents. An agent is anything that can perceive its environment through sensors and can act upon that environment through effectors (Russell and Norvig 2002). By allowing agents to reason logically and act autonomously (free of human control) towards a goal, they can be attributed a notion of intelligence. They are aware of the repercussions of their actions on the environment and dynamically integrate their experiences into existing reasoning mechanisms. In the suggested multi-agent environment, each agent handles a specific reasoning aspect of the environment.

The agent is responsible for simulating the environment by generating current events that are an outcome of past participant interactions or, by randomly generating seed events. Secondly, the agent can predict future consequences of present circumstances; as warning flags for the participant and also as a guideline for effectively planning the future of the environment. Finally, the agent should be able to depict the sensitivities of the environment to changes in specific aspects and its inter-relationships, in other words, the system dynamics. This allows it to portray differences in the 'As-Built' and the 'As-Planned' trends.

In order to accomplish the first two duties, the agent needs to be perceptive to changes in the environment affected by the participant as well as be able to effect changes in it. It must also have awareness regarding the context specific causal reasoning about actions and events, which governs the environment.

There are two kinds of agents, The Mathematical Agent (MA) and the Logical Agent (LA). Agents interact with the environment by changing values of the variables.

Mathematical Agent. Systemic reasoning is based on the mathematical model defined in Rojas and Mukherjee (2003). It deals with reasoning about how events affect the net equilibrium of the system. If the project is executed ‘As-Planned,’ then the system equilibrium is not affected. However, every time there is an event, which results in a crisis, the equilibrium is disturbed. This allows the simulation to constantly give the participant graphical feedback regarding progress made in the project execution as compared to the ‘As-Planned’ implementation.

Logical Agent. The logical agent can create events and also infer events, which follow as a result of user interactions with the simulated environment. It can create an event when it takes an action, which change attributes of variables that coincide with some pre-condition set of variables that associated with an event defined in the knowledge base of events. The participant encodes his/her decisions by changing attributes of variables accessible to them. Hence by perceiving changes in variable and comparing with previous values of variables the agent can logically infer the effect of the participant’s changes on the environment. Reasoning is based on the following closures:

- Event Closure: An occurrence of an event implies that an action occurred.
- Attribute Closure: Reflects a closure on the attributes and variables and expresses that any change in attributes of variables implies that an event has occurred.

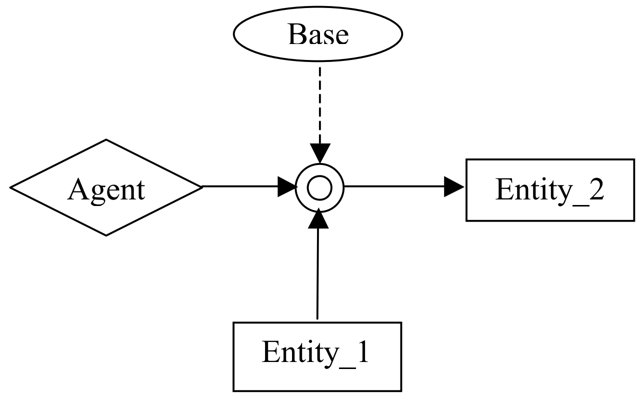

Figure 1. An Agent Operation

Each agent has a finite set of operators associated with it. MA operators are Unite and Compute, while LA operators are Inference and Event Generation. Entities are defined as the different classes of information in the simulation environment. Every agent operation takes an information entity as an input and transforms it to another information entity (Fig. 1). Atomic entities can be combined to create super entities when the super entity is a logical parent of the atomic entities. The set of entities in the simulation environment consist of: As-Planned Data, As-Built Data, Static Derived Data, Dynamic Derived Data, Activity Dependent Events, Global Events, Activity Specific Variables, Global Variables and Independent Variables.

Pro and Information Attitudes (Woodridge and Jennings 1995) are inherited by agents from knowledge bases (KB), databases (DB) and feedback from user interaction (PR). This allows the agent to reason autonomously.

The Multi-Agent framework is based on a combination of Agent operations (Fig. 2). Operators are reasoning mechanisms attributed to each agent. This allows the agent to reason autonomously. Entities are defined as the different classes of information in the simulation environment. Every agent operation takes an information entity as an input and transforms it to another information entity. The framework can be expressed as a series of operations, in series and parallel. Developers of general-purpose simulations can come up with different simulations by changing the properties of the entities and the rules used by the agent to reason about the simulated world within the prescribed framework.

Conclusions

This paper briefly summarizes the research efforts at developing a multi-agent framework for a general-purpose situational simulation environment. The logical and mathematical agents have already been implemented. The interface development is part of ongoing research. The framework described will provide the first step towards developing a general-purpose language for developing situational simulations. Given the need for simulation environments in training construction managers. This could encourage the development of educational simulations within a general-purpose framework.

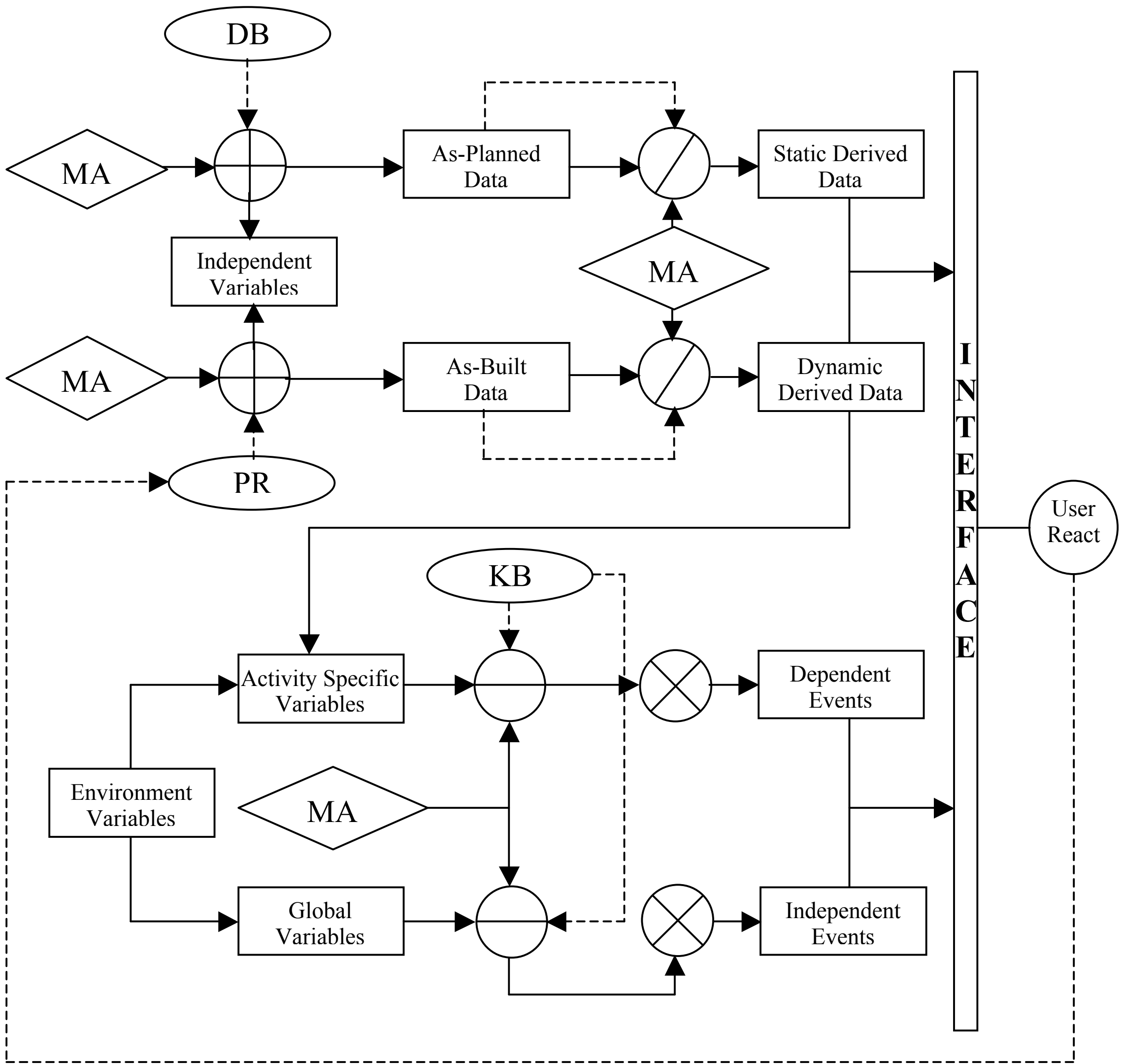

Figure 2. The Agent Entity Framework

References

AbouRizk, S. (1993). "Stochastic simulations of construction bidding and project management." Microcomputers in Civil Engineering, 8, 343-353.

Allen, J.F. and Ferguson, G. (1994). "Actions and events in interval temporal logic." Journal of Logic and Computation, Special Issue on Actions and Events,

Hajjar, D. and AbouRizk, S. (1999) "Simphony: an Environment for building Special Purpose Construction Simulation Tools." Proceedings of the Winter Simulation Conference, 1999, P.A.Farrington, H.B.Newbhard, D.T.Sturrock and G.W.Evans, eds.

Halpin, D. and Woodhead, R. (1970). "A Computerized Construction Management

Game." Department of Civil Engineering, University of Illinois, Urbana Champagne.

Jaafari, A., Manivong, K., Chaaya, M. (2001). "VIRCON: interactive system for teaching construction management." Journal of Construction Engineering and Management, 127 (1), 66-75.

Laird, J.E., Newell, A. and Rosenbloom, P.S. (1987). "Soar: an architecture for general intelligence." Artificial Intelligence 33(1), 1-64.

Martinez, J. and Ioannou, P. (1999). "General-purpose systems for effective construction simulation." Journal of Construction Engineering and Management, 125 (4), 265-276.

McCabe, B., Ching, K.S., and Savio, R. (2000). "STRATEGY: a construction simulation environment." Proceedings of Construction Congress VI, ed. Kenneth D. Walsh, pp 115-120.

Mukherjee, A. and Rojas, E. (2003) "Reasoning about actions and events in situational simulations." Proceedings of Winter Simulation Conference, 2003

Ndekugri, I. and Lansley, P. (1992). "Role of simulation in construction management." Building Research and Information. 20 (2), 109-115.

Rojas, E., and Mukherjee, A. (2003) "Modeling the construction management process to support situational simulations." Journal of Computing in Civil Engineering, ASCE, vol 17(4) pp 273-280.

Russell, S. J. and Norvig, P. (2002) "Artificial intelligence: a modern approach." 2nd. Ed. Prentice Hall.

Soibelman, L. and Peña-Mora, F. (2000) "Distributed multi-reasoning mechanism to support conceptual structural design." Journal of Structural Engineering, June 2000, pp: 733-742.

Tambe, M., Johnson, W. Lewis, Jones, Randolph M., Koss, F., Laird, John E., Rosenbloom, Paul S., and Schwamb, K. (1995) "Intelligent agent for interactive simulation environments." Spring 1995 Issue of AI Magazine

Woodridge, M. and Jennings, N.R. (1995) "Intelligent Agents: Theory and Practice." Springer Verlag lecture notes in AI, Volume 890, January 1995.