Proceedings of the 2008 Winter Simulation Conference
S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler eds.

DISTRIBUTED AGENT-BASED SIMULATION OF CONSTRUCTION PROJECTS WITH HLA

Hosein Taghaddos
Simaan M. AbouRizk
Yasser Mohamed
Ivan Ourdev

Dept. of Civil and Environmental Engineering
University of Alberta
Edmonton, AB, CANADA

ABSTRACT

Simulation techniques can provide a resource-driven schedule and answer many hypothetical scenarios before project execution to improve on conventional project management software applications for large-scale construction projects. However, the current process of simulation and optimization of resource utilization is a time consuming process especially for large-scale projects. This study employs High Level Architecture (HLA) to develop distributed agent based simulation models. These models are composed of several individual modeling components (federates) that can cooperate with each other for the simulation model (interoperability). These federates are developed in a generic way for reuse on future construction projects. A number of agent-based federates are considered for managing various aspects of the project and to enhance the performance of the simulation model. This framework is illustrated using two case studies, module assembly yard and tower crane, that investigate the feasibility of the proposed approach.

1 INTRODUCTION

Simulation modeling has become a reputable tool for capturing uncertainty in construction projects and for managing resources in the planning and execution phases. An elegant simulation model can answer many hypothetical scenarios. However, the current process of simulation modeling cannot easily find the best scenario. Improving and optimizing the simulation model is labor-intensive for the user (modeler or scheduler). He\she must build the model, observe the model, find the improvement areas and implement the required changes to be able to make decisions at different phases of simulation experiment. Studies show that inserting intelligent agents into the model empowers the simulation process significantly (Mohamed and AbouRizk 2005; Mukherjee 2005; Sawhney 2003; Van Tol 2005).

Previous studies on building intelligent simulation models have shown that the intelligent simulation models in the current framework quickly become overwhelmed in terms of the size, detail, and the nature of their interactions. Van Tol (2005) indicates that “the manner in which the simulation program currently handles artificial intelligence programs outside of the simulation program heavily taxes the performance of the computer.” Mohamed also highlights that embedding an agent inside the simulation model as an event scheduling simulation approach results in “an inefficient simulation processing time that renders the simulation of realistic large model impractical” (Mohamed and AbouRizk 2005).

This research addresses the challenge to model efficiency using a High Level Architecture (HLA) framework. The HLA architecture supports building complex systems using distributed simulation technologies. In addition, it provides standards for splitting the model into a number of manageable components (federates) while maintaining interoperability between them. The main benefit of this approach is to facilitate reusability of the developed federates in other simulation models. In this architecture, some federates can act as intelligent agents to manage various aspects of the simulation model. For instance, an agent can act as a resource-leveling agent that manages the resource allocation and levels the resource utilization. This system has proved to work efficiently in other industries (Bruzzone et al. 2005; Cicirelli 2007; Lees et al. 2007). Every agent tries to balance and satisfy its own objective function in the best way. Different agents communicate with each other and collaborate to reach their own goals.

The proposed approach leads to an optimized system whose successful implementation will have a significant impact on improving the schedule and productivity of a project. This approach is illustrated using two actual case studies.

2 BACKGROUND

To provide a better perspective on the research topic, a brief review of the fundamental concepts is presented in this section.

2.1 Intelligent Multi-Agent System

The term “agent” is widely used in a number of technologies, such as in databases, artificial intelligence, operating systems, and computer networks literature (Bellifemine et al. 2007). Although there is no universally accepted definition of the term agent, there is a general consensus that autonomy is essential to the notion of agency (Weiss 1999). Wooldridge and Jennings (1995) describe an agent as “a self-contained program capable of controlling its own decision-making and acting based on its perception of its environment, in pursuit of one or more objectives”. Nwana (1996) emphasizes the possession of autonomy, co-operation, and learning for the agents (Figure 1). These behavioral attributes assist the agent in operating without human guidance, in cooperating with other agents, and in learning (Wooldridge and Jennings 1995). The possession of all three attributes is what indicates an intelligent agent. Weiss (1999) defines intelligent agents as being capable of flexible autonomous action.

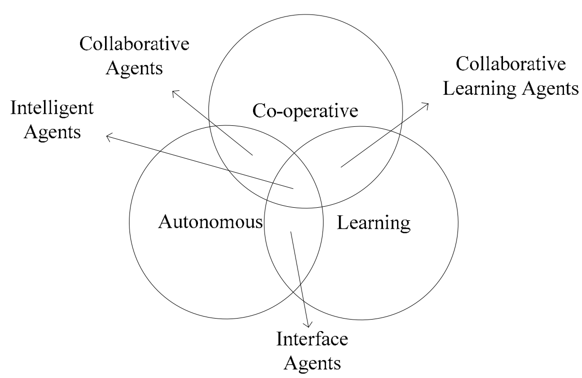


Figure 1: Nwana’s (1996) requirements for agenthood

A number of agents can act collectively as a society and generate a multi-agent system (MAS). The main advantage of MAS is in expanding the functions of individual agents beyond their interconnected capabilities (Ren and Anumba 2004). However, the development of MAS under current tools and methodologies is often very difficult. MAS simulation has helped to create more flexible systems to simplify the modeling approach, allowing the system designer to focus on the main aspects of each system.

In spite of the ease in their modeling, MAS simulators still suffer from lack of interoperability and performance. An agent that is designed for one specific model may not be easily applied in another project. The problem is particularly complicated if an agent is required to act in a different simulation environment. Moreover, MAS simulation may overwhelm computational resources in a single computer, greatly decreasing its performance speed. Distributed simulation techniques (i.e. HLA) provide a promising solution to solve these challenges (Lees et al. 2007).

2.2 High Level Architecture (HLA)

High Level Architecture (HLA) is an approach for integrating separate components of a simulation models, referred to as federates, into a single distributed simulation model, called a federation (Fujimoto 2003; Kuhl, Weatherly, Dahmann 1999). The main intent of HLA is to promote interoperability between simulations and to aid the reuse of models in different contexts, ultimately reducing the time and cost required to create a new environment (Fujimoto 2003; Kuhl et al. 1999; Shahin 2007). These features significantly improve the development of construction simulations since simulation models of different construction applications share a number of common components. This strategy also helps to develop each federate separately with a different simulator and then to combine them (Kuhl et al. 1999).

IEEE standards (The Institute of Electrical and Electronics Engineers, Inc. 2000) characterize HLA by three main components: HLA rules, HLA interface specification, and Object Model Template (OMT). The OMT provides a common framework for data exchange between different federates. The run-time infrastructure (RTI) is software that conforms to the HLA specifications and provides software services such as synchronization, communication, and data exchange between federates to support an HLA-compliant simulation.

3 STATE OF THE ART

Some researchers in construction management have attempted to embed agents and produce automated evaluation tools. McCabe (1997) integrated belief networks into simulation to provide diagnostics for evaluating their performance. She considered some performance measurement indices, such as queue length, queue wait, server quantity, server utilization, and customer delay indexes. At the end of each simulation run, these indexes were calculated to evaluate remedial actions for the next simulation run by means of belief network (McCabe 1997). Van Tol tried to make this approach dynamic to allow the simulation to react to the changes during simulation run (Van Tol 2005). However, he didn't manage to create an autonomous agent that was readily applicable in different construction projects. One of his main problems was that the process interaction construction simulations such as Simphony (Hajjar and AbouRizk 2002) do not provide an efficient environment for handling autonomous generic agents. Van Tol (2005) indicates that "the manner in which the simulation program currently handles artificial intelligence programs outside of the simulation program heavily taxes the performance of the computer". Mohamed also highlights that embedding an agent inside the simulation model as an event scheduling simulation approach results in "an inefficient simulation processing time that renders the simulation of realistic large model impractical" (Mohamed and AbouRizk 2005).

HLA has solved this challenge in the other industries. For instance in computer science, Zhao (Zhao et al. 2005) and Lees (Lees et al. 2007) have integrated agent-based architecture with HLA. Lees has shown that "significant speedup can be achieved by distributing agent federates across multiple cluster nodes, even in the case of lightweight agents with modest computational requirements." He has also claimed that this framework can work better than other distributed agent simulations including DGensim (Anderson 2000), JAMES (Uhrmacher 2001), MACE3J (Gasser and Kakugawa 2002), and SPADES (Riley 2003) in terms of interoperability, reusability, flexibility, and system speedup.

4 DISTRIBUTED AGENT-BASED SIMULATION WITH HLA

The main objective of this research is employing intelligent agents to create autonomous agents applicable in different construction projects. Each agent is of the generic type, that automates some aspects of decision making processes. For example, one may be a resource allocation agent that enhances the resource utilization. High Level Architecture (HLA) provides the appropriate framework. In the above example, if there is more than one type of resource in the project, there will be a separate resource allocation agent corresponding to each resource. These agents will collaborate together to improve the whole behaviour of the system. The main advantage of the HLA approach is that it promotes interoperability between simulations and aids the reuse of models in different construction projects. This approach enhances the performance of simulation models of large-scale construction projects.

4.1 Agent-Based Federates

Although they may seem quite different, construction projects share some common problems. For example, they:

- usually represent multi-unit projects.
- have resource-driven scheduling.
- are affected by the holidays according to the calendar, as well as by weather conditions.
- can benefit from real-time visualization to consider the constructability issues.

In the traditional approach, we try to deal with these issues in each single project. We propose to design some generic federates based on intelligent agent theory to handle the common issues of construction projects. We employ HLA architecture to these agent-based federates within the simulation model. To demonstrate and validate this approach, two typical construction applications, a module assembly yard and tower cranes, will be studied in this project. At the current stage, we have considered four agent-based federates to resolve the above mentioned issues (Figure 2).

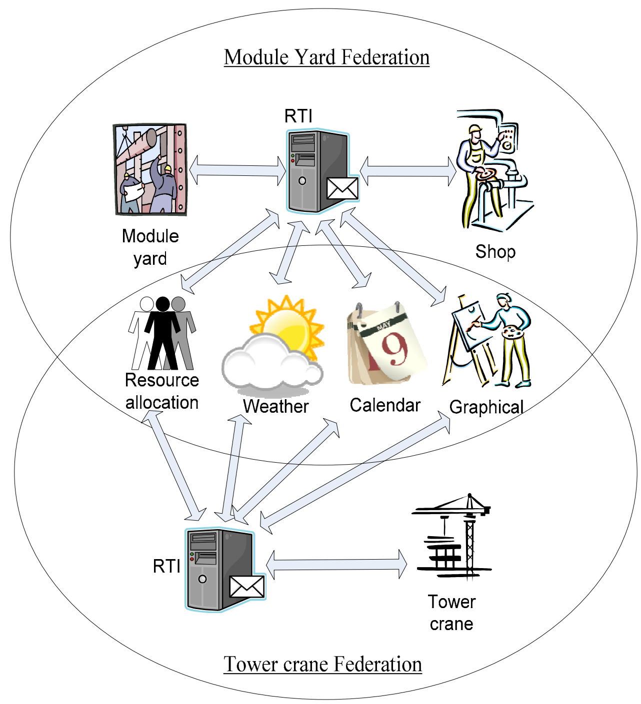


Figure 2: Module yard and tower crane federations

- *Resource allocation agent*: This agent acts as an "intelligent observer" placed within the distributed simulation model. It watches the model's performance, resource utilization, and queue length. It also interferes and makes real-time decisions regarding the overall model performance. This agent will be designed based on the resource leveling methods in job shop scheduling (Gopalakrishnan 1988; Karumanasseri and Abourizk 2002), as well as our results from the previous experience (Taghaddos et al. 2008). If there is more than one type of resource, different instances of this agent have to be cloned, corresponding to different resources.

- *Weather generation agent*: This agent is used to model random processes affecting construction, such as weather conditions. In regions such as Alberta, cold (or warm) weather can significantly affect the schedule. Shahin (2007) has developed a framework to quantify and include different weather parameters affecting construction processes. This federate helps to develop a realistic schedule that take into considerations the time of the year, the location of the project, and the impact of weather on the productivity (Shahin 2007). Wales considered a combined discrete-event/continuous simulation to model the weather impact on productivity, and proved the effectiveness of his methodology (AbouRizk and Wales 1997; Wales and AbouRizk 1996). In this study, we would like to combine both approaches to produce a flexible federate. We can also incorporate fuzzy theory for evaluating quantitative factors and assessing the weather impact on the schedule.
- *Calendar agent*: It is a generic federate applicable to many construction projects. It takes into the consideration national holidays and long weekends according to the location (country) of the project.
- *Real-time graphical agent*: This federate demonstrates real-time graphical data while the simulation model is working. This federate can connect to a graphical software, shows the module location, and activities graphically while the simulation federate is running. This federate methodology also has sufficient flexibility, allowing the simulator to incorporate the model and apply his/her decisions while it is running (i.e. the user can allocate the module to a specific bay by independently).

According to Nwana's definition (1996), resource allocation agent and weather generation agent are intelligent federates; i.e. it must possess autonomy, co-operate with other resource allocation agents, and learn throughout the life of the project. The other two agents are collaborative agents that posses autonomy and co-operate with their instances.

4.2 Proposed Framework

Figure 3 illustrates the proposed framework for the distributed agent-Based simulation. Each agent has a sensor and an actuator based on the construction domain. Whenever the construction domain changes, we can still make use of the agent by replacing sensor and actuator of the agent. Agents initialize their parameters based on the OMT of federation and collaborate through the RTI to exchange information.

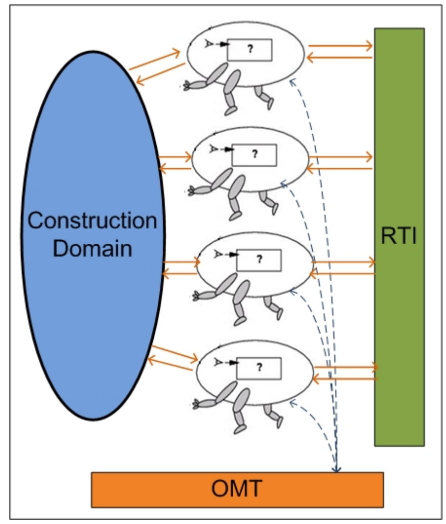


Figure 3: Distributed Agent-Based Framework

4.3 Implementation Environment

This study uses the Construction Simulation Environment (COSYE) software environment, which runs on the Windows.NET, as well as MS office applications, mainly MS Access and MS Visio. COSYE (AbouRizk and Robinson 2006; AbouRizk et. al 2006) is an HLA-based simulation environment, developed at the University of Alberta. This framework is composed of RTI, a generic "base" federate, generic discrete-event and time-stepped federates, a suite of generic modeling elements within a construction context that allow for the graphical development of the HLA federates, and an environment that is optimized for the development of federations for construction applications. During run time, the framework provides the necessary communication, information exchange, and data-sharing protocols using an RTI that assures simulation synchronization, coordination, and consistency between the different federates. The conceptual architecture of the envisioned framework is shown in Figure 4.

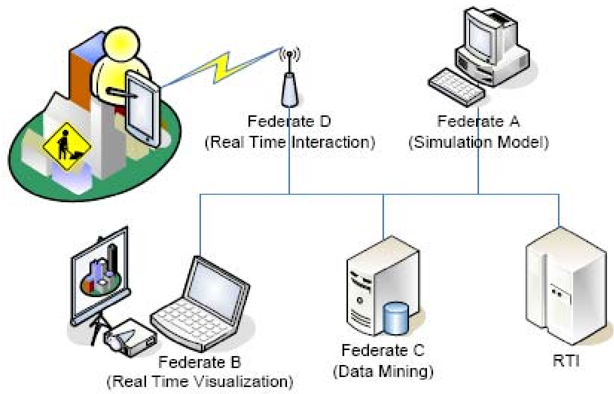


Figure 4: Framework architecture of COSYE

5 APPLICATION AREAS

This research considers simulation models of two different construction projects. The authors have been involved in simulating these projects using Simphony.NET environment. These simulation models are briefly introduced through the case studies and then the new platform using the HLA architecture is described.

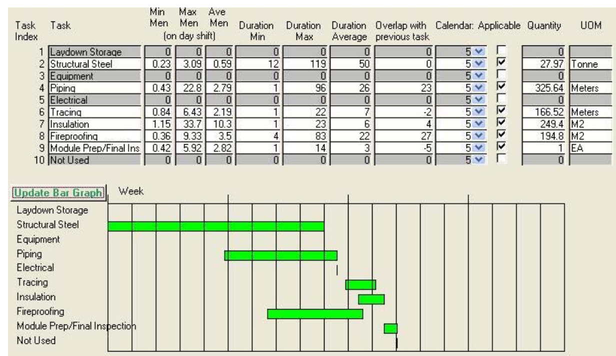
5.1 Simulation-based Scheduling of Module Assembly Yards

The first case study involves simulation-based scheduling of pipe-spool module assembly. Figure 5 displays the layout of the module assembly yard for PCL Industrial in Edmonton, Alberta. This project is affected by a number of uncertain factors and resource constraints in the yard as well as the fabrication shop. These factors also pose a challenge for the scheduler to optimize both the use of available resources (e.g. space, crew) and to meet the project's delivery deadlines (Mohamed et al. 2007).

Figure 5: PCL module yard layout

To assemble a module, first it should be allocated to a suitable space in the yard depending on its type, length, early start (representing the time that all the components are available), estimated duration of the assembly process, and shipping date (Mohamed et al. 2007). The assembly process for the module consists of a number of activities with a range of durations, overlaps with predecessor activities, and required manpower based on historical information (Figure 6).

A Special Purpose Simulation (SPS) template was developed for simulation-based scheduling of the module yard. The model also tried to smooth out the resource utilization curve using some heuristic optimizations (Taghaddos et al, forthcoming 2008). Although this methodology produces significant improvement in the resource utilization of the project, still there is lots of room for the optimization of resource utilization.



Task Index	Task	Min Men (on day shift)	Max Men (on day shift)	Ave Men (on day shift)	Duration Min	Duration Max	Duration Average	Overlap with previous task	Calendar:	Applicable	Quantity	UOM
1	Laydown Storage	0	0	0	0	0	0	0	5	☐	0	
2	Structural Steel	0.23	3.09	0.59	12	119	50	0	5	☑	27.97	Tonne
3	Equipment	0	0	0	0	0	0	0	5	☐	0	
4	Piping	0.43	22.8	2.79	1	96	26	23	5	☑	325.64	Meters
5	Electrical	0	0	0	0	0	0	0	5	☐	0	
6	Tracing	0.84	6.43	2.19	1	22	7	-2	5	☑	166.52	Meters
7	Insulation	1.15	33.7	10.3	1	23	6	4	5	☑	249.4	M2
8	Fireproofing	0.36	9.33	3.5	4	83	22	27	5	☑	194.8	M2
9	Module Prep/Final Ins	0.42	5.92	2.82	1	14	3	-5	5	☑	1	EA
10	Not Used	0	0	0	0	0	0	0	5	☐	0	

Figure 6: Pipe spool module assembly process

5.2 Simulation-based Scheduling of Tower Crane

The second case study involves simulation-based scheduling of tower cranes in construction sites. Tower cranes are the most frequently shared resources on construction sites, and ineffective management of the tower crane may create bottlenecks on the site (Appleton et al. 2006). Figure 7 displays the layout of a PCL construction site in Edmonton as the case study of this project.

Similar to the module yard, an SPS template was developed for simulation-based scheduling of the tower cranes using priority logic. In this model, prioritized work packages are identified on a weekly basis throughout the project, and the attributes of each work package including early start date, number of lifts, source and destination coordinates and elevation are defined.

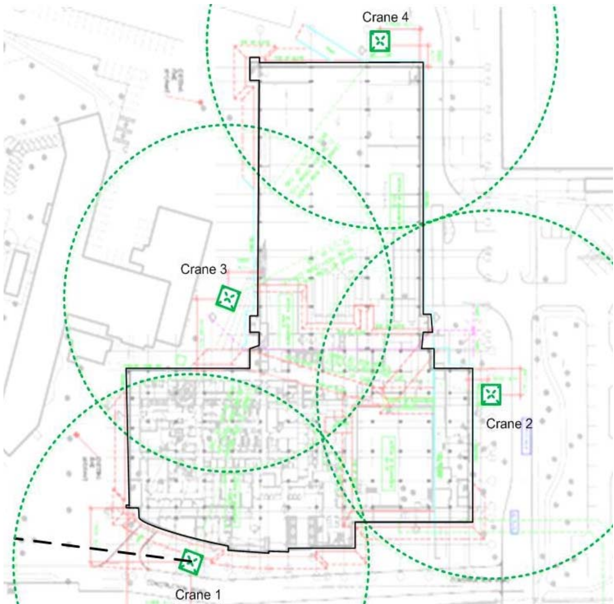
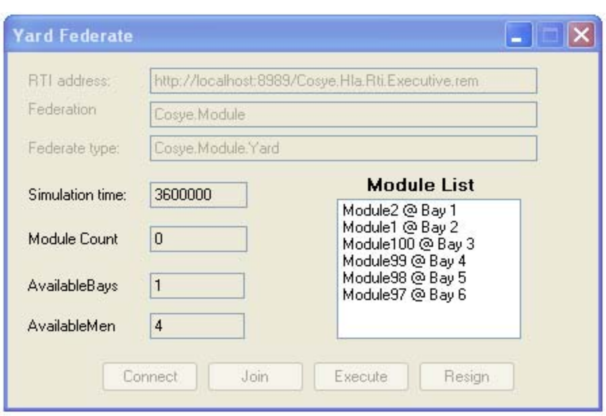


Figure 7: PCL construction site layout

5.3 Current Implementation

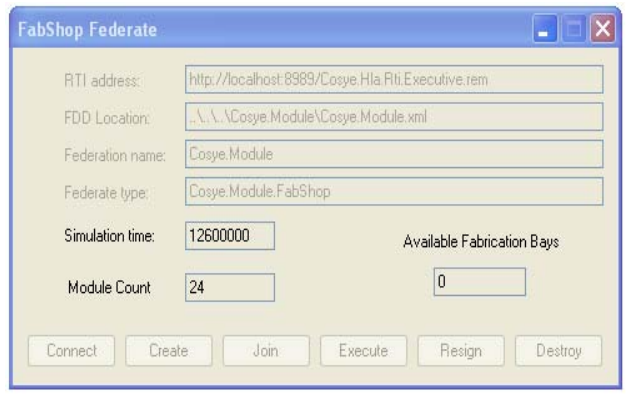
Part of module yard federation, including shop, module yard and graphical federates, has already been developed in the COSYE environment. This federation is furthermore linked to the spool fabrication shop federate in order to resemble the real assembly project (Figure 8(a), (b)). In practice, a module assembly process in the module yard starts whenever the required components are fabricated in the shop. Having an integrated model solves the serious scheduling challenge of linking module yard to fabrication shop.

Figures 8(a), (b), (c) represent three different federates designed at the current stage for the module yard federation. Each federate is an individual simulation model with its own resources and is at different simulation time at the captured snap shot.

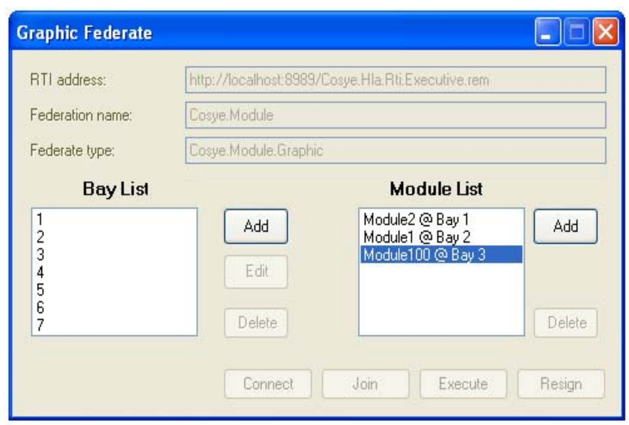


(a) Module yard federate

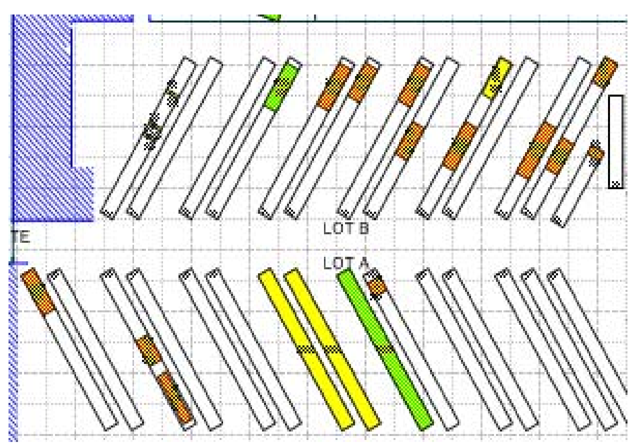
The real-time graphical federate is developed using Visual Studio 2005 Tools for Office (VSTO). This federate connects to Microsoft Visio 2007 and shows the module location and activities graphically while the simulation federate is running (Figure 8 (c), (d)). This federate demonstrates the capability of this systems to collaborate with a third party graphical system.



(b) Fabrication shop federate



(c) Real-time graphical federate



(d) Module Yard graphical representation at Visio

Figure 8: The current stage of module yard federation

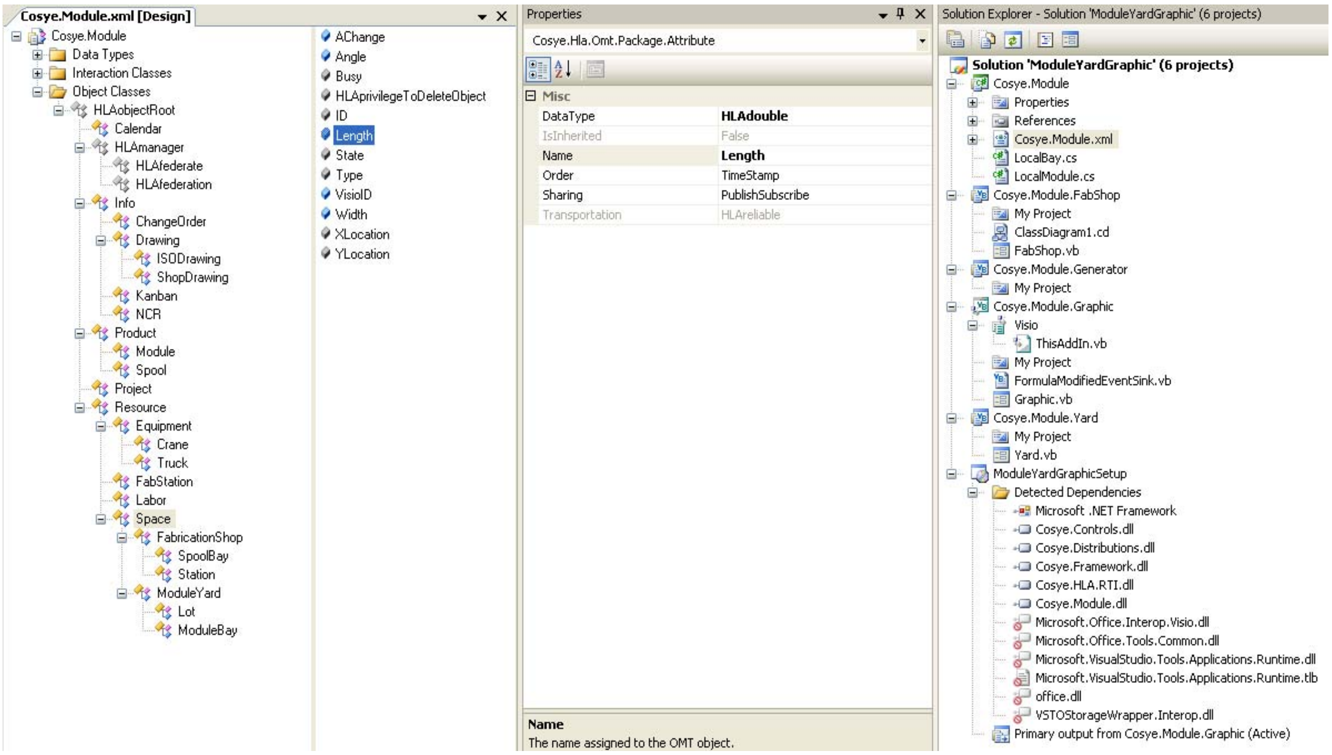


Figure 9: Object Model Template (OMT) of module yard federation

The Object Model Template (OMT) of the module yard federation is shown in the Figure 9. It defines the objects and interactions, which are shared for the whole federation, as well as the attributes, the parameters, their types, and their sharing method. The OMT is very helpful to initialize the structure the agents.

5.4 Future Work

At the next stage, the other three agent-based federates have to be designed similar to the real-time graphical federate. The main challenge in this research is to develop resource allocation and weather generation federates as intelligent agents; i.e. they must possess learning as well as autonomy and co-operation. The sensor and actuator of the agents should also be determined based on the construction domain.

6 CONCLUSION

This paper described a framework for distributed agent-based simulation with high level architecture (HLA) for construction projects. These models are composed of several individual modeling components (federates) that can cooperate with each other for the simulation model (interoperability). They are developed in a generic way that can be reused on other construction projects. This study proposes a number of agent-based federates, including resource allocation agent, weather generation agent, calendar agent, and real-time graphical agents. These federates are generic and can serve on various construction projects to administer various aspects of the project, enhancing the project's productivity. The proposed approach is partially developed and its feasibility is under investigation.

ACKNOWLEDGEMENTS

The author wishes to acknowledge and thank PCL Industrial and PCL Construction companies for their kind collaboration.

REFERENCES

AbouRizk S.M., Al-Hussein M, Mohamed Y. 2006. Development of Synthetic Environments for Select Construction Applications, *Collaborative Research and Development Grant*, Feb. 2006.

AbouRizk, S.M. 2006., Robinson Fayek, A. 2006. Synthetic Environments for Construction Planning and Control. *NSERC Industrial Research Chair Research Program*, Feb. 2006.

AbouRizk, S.M. and Wales J.R. 1997. Combined discrete-event/continuous simulation for project planning. *Journal of Construction Engineering & Management* 123(1):10.

Anderson, J. 2000. A generic distributed simulation system for intelligent agent design and evaluation. *Proceedings of the Tenth Conference on AI, Simulation and Planning*, AIS-2000 :36-44.

Appleton, B.J.A., Niaz, M.A., AbouRizk, S.M., Al-Hussein, M., Mohamed, Y. 2006. Priority rating logic simulation model for tower crane operations management. *Journal of Construction Research* 7(1):133-47.

Bellifemine FL, Caire G, Greenwood D. 2007. Developing multi-agent systems with JADE . *Wiley*, NY.

Bruzzone, A.G., Mosca, R., Revetria, R., Bocca, E., Briano, E. 2005. Agent directed HLA simulation for complex supply chain modeling. *Simulation* 81(9):647-55.

Cicirelli, F. 2007. An agent infrastructure for distributed simulations over HLA and a case study using unmanned aerial vehicles. *ANSS 40th Simulation Symposium*,: 231-8.

Fujimoto, R.M. 2003. *Distributed Simulation Systems*. 124 pp.

Gasser, L. and Kakugawa, K. 2002. *MACE3J: Fast flexible distributed simulation of large, large-grain multi-agent systems*. 745 pp.

Gopalakrishnan, K.N. 1988. The design of a capacity requirements planning system and a decision support system for strategic planning. PhD Thesis, University of Alberta, Edmonton, AB, CA.

Hajjar, D. and AbouRizk, S.M. 2002. Unified modeling methodology for construction simulation. *Journal of Construction Engineering and Management* 128(2): 174-85.

Karumanasseri, G. and Abourizk, S.M. 2002. Decision support system for scheduling steel fabrication projects. *Journal of Construction Engineering and Management* 128(5):392-9.

Kuhl, F., Weatherly, R., Dahmann J. 1999. *Creating Computer Simulation Systems: An Introduction to the High Level Architecture* .

Lees, M., Logan, B., Theodoropoulos, G. 2007. Distributed simulation of agent-based systems with HLA. *ACM Transactions on Modeling and Computer Simulation* 17(3).

McCabe, B.Y. 1997. An automated modeling approach for construction performance improvement using simulation and belief networks. *PhD thesis*, University of Alberta, Edmonton, AB, CA.

Mohamed, Y. and AbouRizk, S.M. 2005. Framework for building intelligent simulation models of construction operations. *Journal of Computing in Civil Engineering* 19(3):277-91.

Mohamed, Y., Borrego, D., Francisco, L., Al-Hussein, M., AbouRizk, S.M., Hermann, U. 2007. Simulation-based scheduling of module assembly yards: Case study. *Engineering, Construction and Architectural Management* 14(3):293-311.

Mukherjee, A. 2005. A multi-agent framework for general purpose situational simulations in construction management. *PhD Thesis*, University of Washington, Washington DC.

Nwana, H.S. 1996. Software agents: An overview. *Knowledge Engineering Review* 11(3):1-40.

Ren, Z. and Anumba, C.J. 2004. Multi-agent systems in construction-state of the art and prospects. *Automation in Construction* 13(3):421-34.

Riley, P. 2003. *MPADES: Middleware for parallel agent discrete event simulation*. 162 p.

Sawhney, A. 2003. Agent-based modeling and simulation in construction. *Proceedings of the 2003 Winter Simulation Conference* 2:1541-7.

Shahin, A.A. 2007. A framework for cold weather construction simulation. *PhD Thesis*, University of Alberta, Edmonton, Alberta

Taghaddos, H., AbouRizk, S.M., Mohamed, Y., Hermann, U. Forthcoming 2008. Simulation-based resource leveling in multi-project construction. *CSCE 2008 Annual General Meeting & Conference*.

The Institute of Electrical and Electronics Engineers, Inc. 2000. IEEE 1516 for Modeling and Simulation (M&S) High Level Architecture (HLA) -Framework and Rules, |*IEEE Standard*.

Uhrmacher, A.M. 2001. Dynamic structures in modeling and simulation: A reflective approach. *ACM Transactions on Modeling and Computer Simulation* 11(2):206-32.

Van Tol, A.A. 2005. Agent embedded simulation modeling framework for construction engineering and management applications. *PhD Thesis*. University of Alberta, Edmonton, Alberta.

Van Tol, A.A. and AbouRizk, S.M. 2006. Simulation modeling decision support through belief networks. *Simulation Modelling Practice and Theory*, 14(5):614-40.

Wales, R.J. and AbouRizk, S.M. 1996. An integrated simulation model for construction. *Simulation Practice and Theory* 3(6):401-20.

Weiss, G. 1999. Multiagent systems: A modern approach to distributed artificial intelligence. *PhD Thesis*, MIT, Cambridge, Mass.

Wooldridge, M. and Jennings, N.R. 1995. Intelligent agents: Theory and practice. *Knowledge Engineering Review* 10(2).

Zhao, Z., Albada, D.V., Sloot, P. 2005. Agent-based flow control for HLA components. *Simulation* 81(7):487-501.