DEVELOPING COMPLEX DISTRIBUTED SIMULATION FOR INDUSTRIAL PLANT CONSTRUCTION USING HIGH LEVEL ARCHITECTURE

Simaan AbouRizk
Yasser Mohamed
Hosein Taghaddos
Farzaneh Saba
Stephen Hague

Department of Civil & Environmental Engineering
University of Alberta
Edmonton, AB T6G 2W2, CANADA

ABSTRACT

Large, complex construction projects, such as industrial plant construction, are not well suited to discrete-event process interaction simulation. The authors present a distributed simulation of industrial plant construction using separate modules inter-linked via the High Level Architecture (HLA). This enables the capture of all features, resources, and processes required to design, build, and maintain a facility, and an HLA-based approach simplifies collaborative development and improves reusability of components. The proposed methodology for simulating industrial construction is validated using COSYE, a Construction Synthetic Environment developed at the University of Alberta.

1   INTRODUCTION

In response to the increasing complexity of construction systems, sophisticated methods and tools for the modeling and analysis of these systems must be developed. Advances in information technology, along with significant progress in simulation techniques, have facilitated the use of such methods in other industries. However, the construction industry has failed to exploit these technological advancements to their full advantage. For example, simulation has been a formal topic in construction engineering research for almost three decades, but still has not been embraced as a practical tool in the industry. One of the primary reasons for this lack of research transfer is that simulation models often do not include the entire project delivery process and are limited to a part of construction processes. These models are built as isolated software systems that are neither able to link with various construction management applications nor to integrate with other construction simulation models under the same simulation environment.

In other industries, different approaches have been taken to address these challenges. The United States Department of Defense (DoD), for instance, has developed the High Level Architecture (HLA) to facilitate the integration of distributed simulation models within an HLA environment. HLA allows the division of a large scale model, known as a federation, into a number of manageable components (i.e., federates), while maintaining interoperability between them. In the last decade, HLA is deployed in a broad range of simulation application areas including transportation and the manufacturing industry.

HLA has been recently employed in the construction industry at the academic and industry level. In this regard, an HLA-based simulation environment, referred to as the Construction Simulation Environment (COSYE), has been developed at the University of Alberta (AbouRizk et al. 2006; AbouRizk and Robinson 2006). COSYE has been applied to model various large-scale construction projects in various

areas. This paper presents COSYE's facilitation of the collaborative development of modeling of industrial plant construction across all project phases and throughout the facility's life cycle.

2   BACKGROUND AND STATE OF THE ART

Simulation has been an important area in construction research for three decades, and to this point a number of construction simulation tools have been developed. The stream of construction simulation tools began with CYCLONE (Halpin 1977) and was followed by MicroCYCLONE (Lluch 1981), RESQUE (Chang 1986), STROBOSCOPE (Martinez 1996), and Simphony (Hajjar and AbouRizk 1996), which is a simulation platform for building special-purpose simulation (SPS) templates and models. Simphony can focus on one particular domain of construction operations and facilitate modeling projects within that domain, which reduces the time required to develop SPS tools.

These construction simulation tools are effective when the level of abstraction is manageable. However, the industrial construction process includes drafting, material procurement and supply, shop fabrication, module assembly, and on-site installation. Using SPS either a simplified version of the entire process at a high level of abstraction, or a detailed version of just a portion of the process, such as fabrication or module assembly, can be simulated. The problem with both of these approaches is that they fail to capture comprehensively and exhaustively the entire industrial construction process. The first approach (a high level of abstraction) does not include an acceptable level of detail reflecting product features and process interactions; the second approach (a detailed version of a portion of the process) simulates a range of interactive, interdependent processes in isolation from each other. Both of these approaches result in unanswered questions and vague areas in the planning and management of construction projects. Moreover, claiming accuracy of results and referring management and planning decisions to predicted results based on an incomplete simulation process is not acceptable.

Wang (2006) pioneered simulating the entire industrial construction process in detail, but faced many limitations in using SPS. A lack of simulation re-use, composability, standardization, computing ability, and versatility in simulating large-scale industrial construction are some of the shortcomings he has identified (Wang et al. 2005).   These limitations are addressed by the High Level Architecture system.

High Level Architecture (HLA), as previously mentioned, was developed in the context of defense applications in the mid-1990s, then standardized by IEEE. The main purpose of HLA is to support component-based simulation so that the development effort is distributed among multiple groups with specific professional interests. HLA also allows end-users to customize their own combination of simulation components (federates) from a repository based on their own requirements and interests. The component models communicate through a Run Time Infrastructure (RTI) using standard HLA-compliant protocols. HLA supports the interaction of simulation components and facilitates their reusability and interoperability. The components are independently modeled and executed, and developers can define their own set of object and interaction classes for data exchange with other simulation components. This underlying common object model or "interchange document," known within HLA terminology as the Federation Object Model (FOM). According to the rules of HLA, each simulation model (federation) should have an HLA Federation Object Model constructed in accordance with an Object Model Template (OMT) (IEEE 2000.Std 1516.2 2000). OMT provides a common framework for HLA object model documentation with a standard format and syntax.

The Run-Time Infrastructure (RTI) is the backbone of the federation. It provides software services such as synchronizations, communication, and data exchange services between federates to support and HLA-compliant simulation. The Construction Simulation Environment (COSYE) provides a powerful RTI which conforms to the HLA specifications. The employment of COSYE facilitates modeling of industrial construction and overcoming the above-mentioned challenges of traditional construction simulations.

3 MODELING INDUSTRIAL CONSTRUCTION PROCESSES IN THE COSYE ENVRONMENT

Industrial construction is the construction process of industrial plants (i.e., refineries and oil-processing plants, buildings). These industrial plants are usually built based on modular construction in the province of Alberta, Canada. This technique provides numerous benefits for both constructor and client, including shorter project duration, reduced number of workers onsite, reduced project costs, improved safety and quality, more flexible construction processes, and eased site congestion (Taghaddos et al. 2008).
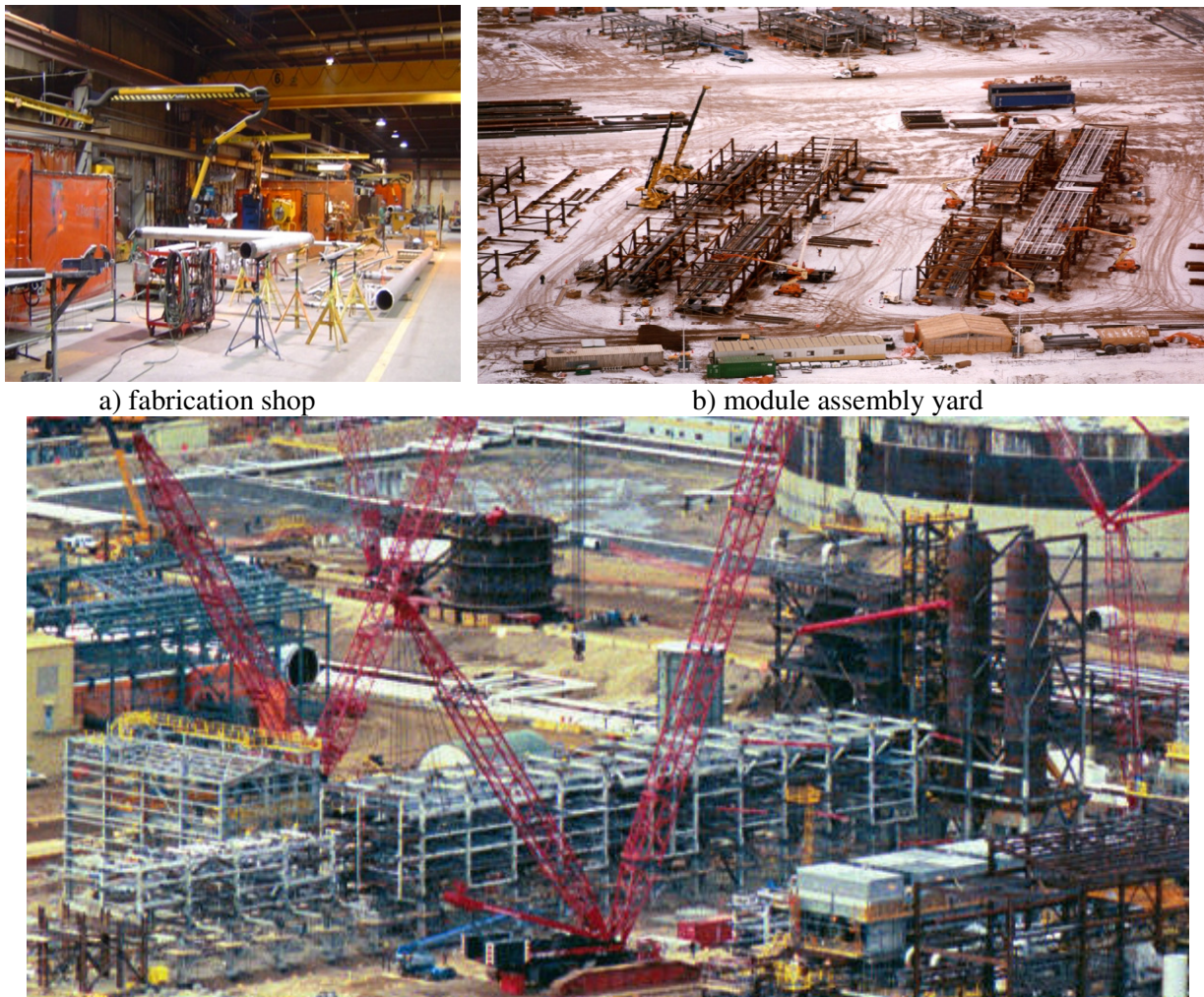

a) fabrication shop   b) module assembly yard

Figure 1: Supply chain of Industrial construction

In modular construction, a "module" refers to a pre-constructed unit that can easily be inter-connected with other such units to make a structure. Pipe spool modules are usually assembled off-site in a yard near the spool fabrication shop, referred to as the module assembly yard and made up of several bays where modules are assembled. A module is typically made of pre-assembled components such as spools, structural steel frames, racks of pipes, cables, equipment, or other components. A spool is composed of several pipes and connections that are welded together in an indoor facility (i.e., fabrication shop). Figure 1 displays a view of a fabrication shop, the module assembly yard, and the construction site of PCL Industrial Management Inc.

The industrial construction process covers a wide range of operations: drafting, material supply, shop fabrication and module assembly, and site construction. In COSYE, each of these operations is modeled separately as a main federate. These main simulation federates include the procurement, fabrication shop, module yard, and site manager federates. In addition, some domain independent federates are designed as supportive federates to serve one or several federates in the industrial construction federation. These fede-rates include the calendar, resource allocation, and visualization federates. Figure 2 lists the designed fe-derates in the industrial construction federation. The FOM of the industrial construction federation and some of the developed federates are briefly explained in the following sections.
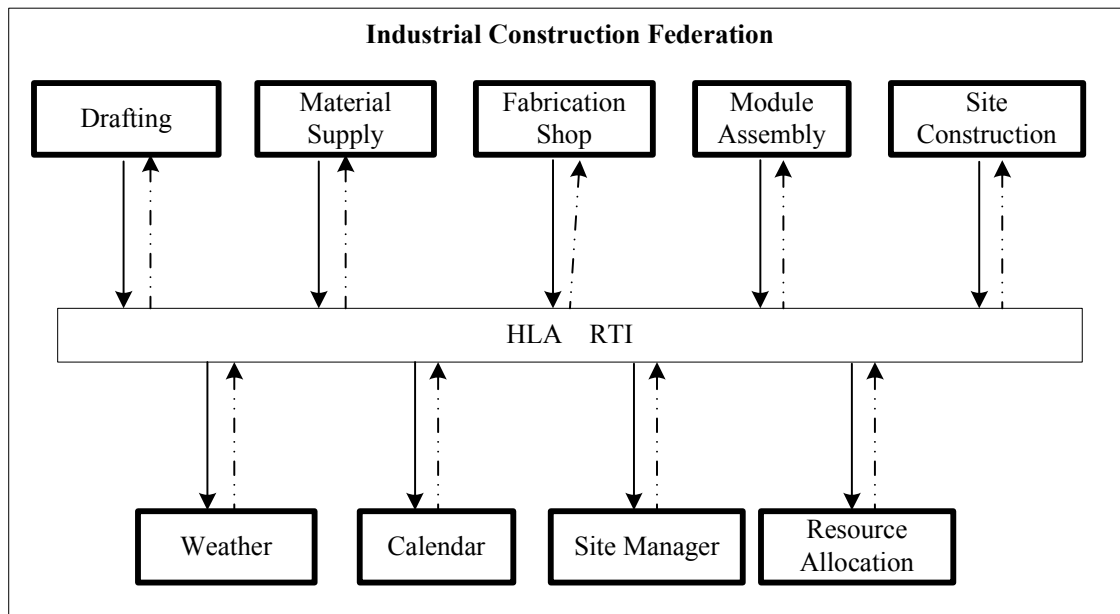

Figure 2: Industrial construction federation

4   FEDERATION OBJECT MODEL OF THE INDUSTRIAL CONSTRUCTION FEDERATION

The Federation Object Model (FOM), which should be developed based on the Object Model Template (OMT), provides the interchange document between simulation components. However, the OMT is not instructive as to how the commonly used and accepted object classes, attributes, and interactions might be identified, or whether or not they are semantically comprehensive or representative of the knowledge do-main. In other words, OMT does not offer a methodology that promises reusable and flexible simulation object model components, limiting the capacity and capabilities of HLA (Base Object Model Study Group 2001).

Therefore, the development of FOM not only should follow HLA rules and the Object Model Tem-plate, but also should seek alternative methodologies that encourage and strengthen simulation interope-rability and reusability and facilitate the development of robust, standardized, and generic FOMs while preserving logical connections, both syntactically and semantically, among the simulation components. In the industrial construction federation development the high level construction ontology has been used as the FOM reference library. The COSYE research team has followed the same ontology throughout vari-ous construction engineering developments in order to increase technical and syntactical interoperability (Figure 3). The ontology is the result of entire COSYE team member consensus; including industrial con-striction federation, structural steel and tunneling federation.
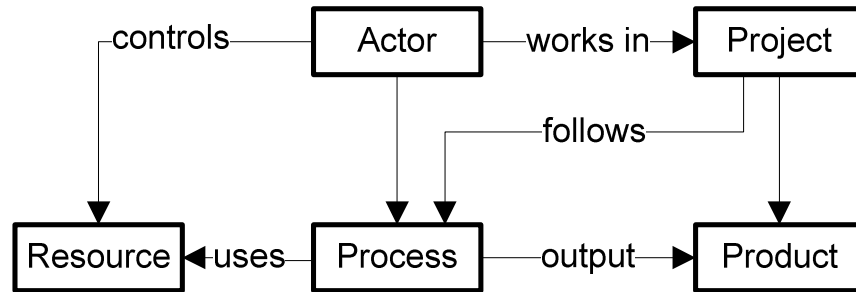
Figure 3: High level ontology of construction operations (adapted from (El Diraby et al. 2003))

Each of these concepts are elaborated for specific federations. As an example, the product input-output chain within the industrial construction federation is shown in Figure 4.
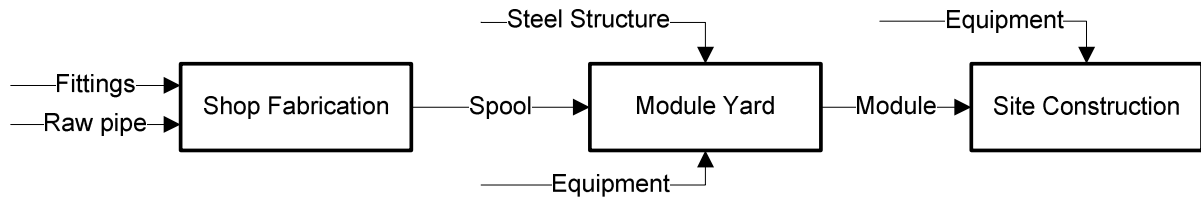
Figure 4: Product chain within industrial construction federation

5   FABRICATION SHOP FEDERATE

The objective of this federate is to simulate the process of fabricating spools in the fabrication shop. The unique features of individual pipe spools and uncertain and complex fabrication process and module assembly dependency on fabrication shop production necessitates accurate simulation modeling of fabrication process.

Each spool is a composition of a number of pipes and fittings which are assembled together. In order to fabricate a spool, it should be decomposed into smaller components which are easier for fabrication. The decomposition of pipe spools depends on unique features of spool such as configuration, material, size and thickness. There are several stations in the spool fabrication shop, including cutting, fitting, welding, Quality Control (QC), stress release, Post Weld Hydro Test (PWHT), painting, and shipping. Figure 5 depicts the typical processes of a spool fabrication shop.

The federate reads the spools' information from the database. After cutting process, fitting and welding process can be done several times on one assembly to complete the spool fabrication process. The fabrication shop simulation model reveals the process bottlenecks and project duration, and at the same time different if-then scenarios can be applied for process improvement.

Based on the simulation, whenever all the spools for one module are fabricated, a message is sent to the module yard federate so that it can start the module yard assembly process.
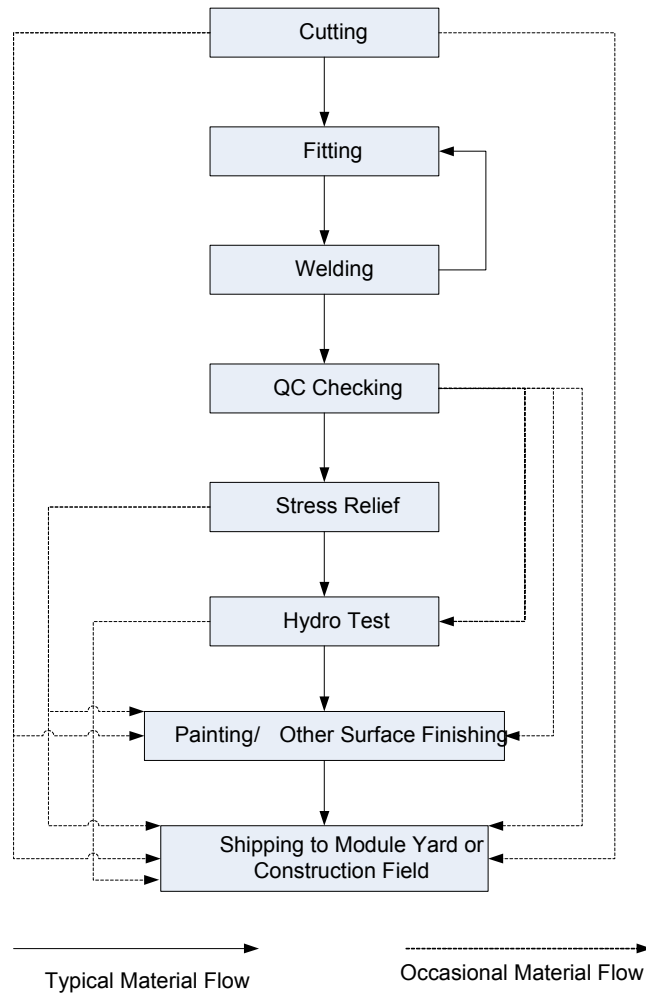
Figure 5: Typical fabrication process

6   MODULE ASSEMBLY YARD FEDERATE

The module assembly yard federate is designed to simulate the processes of the module assembly yard. This federate involves developing a realistic schedule for a module assembly yard that satisfies project constraints and uncertainties; these factors pose a challenge for the scheduler to optimize the use of available resources (e.g., space, crew) and meet the project's delivery deadlines (Taghaddos et al. 2009). Constraints include limitations of space, skilled crew and construction equipment, and physical constraints based on the yard layout and the assembly process. For example, modules can only be shipped out by a transporter once there is empty space in the front of the module in the bay. Otherwise, a crane is required to lift the module and place it in the transporter, which is very costly.

A typical module assembly yard may contain a few hundred modules when it faces a high workload. Scheduling such a multi-project environment with so many constraints involved is very complicated when using network-based scheduling tools (e.g., Primavera). The optimum allocation of resources and the resource leveling are also serious challenges in CPM-based approaches (Mohamed et al. 2007). Therefore, a simulation-based scheduling system for the module assembly yard has been developed and has matured over the last 5 years to be usable for the industry. This simulation model reads the data from a database, simulates the assembly process in the yard, and generates graphical reports. Each module in the devel-

oped discrete event simulation model is an entity that has to capture and release some resource (space, crew, equipment). Overlap between different activities are modeled based on cloning of entities. Once an entity (module) can capture different types of resources (e.g., space in different bays in the yard), an optimization model should be integrated with the simulation model to choose the best resources.

The previous stand-alone simulation model was modified to comply with HLA rules and is now the module yard federate. In the previous model, all the modules were generated at the beginning of the simulation model and scheduled at their Early Start (ES) time. Now, in contrast, a module in the industrial construction federation is generated at the fabrication shop federate once all the required spools are fabricated in the spool fabrication shop. Then the fabrication shop federate updates the state of the module from 'fabshop' to 'ModuleYard'. Once this change is reflected to the module yard federate via the RTI, the federate starts the assembly process at the earlier of either the Early Start (ES) time of the module or the current time of the module assembly federate.

The most important enhancement in the module yard federate over the previous simulation models is separating the optimization component as an independent federate, referred to as the Resource Allocation (RA) federate (Taghaddos 2010). All the modules that are looking for space wait in a queue and determine their utility function over different resource alternatives. This utility function represents numerous affecting factors in the space allocation problem including the modules' total float, the amount of blocking in a bay because of the late delivery of another module in a bay, and the amount of waste space (i.e., not enough to place a module) in a bay. Once the RA federate assigns the available resources to the bidding entity by maximizing the overall utility function (i.e., social welfare), it sends some interactions to the module yard federate to declare the winning modules. After receiving an interaction from the RA federate, the modules in bays schedule an event to capture space. After capturing the bay, a number of activities (i.e., structural steel, piping, electrical, tracing, insulation) must take place before a module can be shipped to the site.

7 SITE CONSTRUCTION FEDERATE

The site construction federate is another main simulation federate designed to simulate crane operations and modular construction in the site. Once a module is assembled in the module assembly yard, it is shipped to the construction site by a transporter. Then it has to be lifted to its predetermined position, once a proper mobile crane in an accessible location with suitable configuration and rigging is available and the predecessor modules (e.g., the bottom modules) are placed in advance. There are also several other constraints in this problem, elaborated in Taghaddos et al. (2010). The simulation model reads the objects, cranes, locations, pick points, crane options, and other general information from the database. The information flow of this federate is described in Figure 6. The simulation model considers the cranes, locations and pick-point as the main resources.

As in the module yard federate, the simulation model of the site construction federate is developed based on the standalone simulation model of the site construction. This federate also takes advantage of the RA federate to allocate the resources to the modules optimally. The bid price for each module is calculated by approximating the real cost incurred in the construction, which is the summation of the actual lifting cost, delay cost and the idle cost of the crane (Taghaddos et al. 2010).

The other feature of this federate is its connection with the module yard federate. The arrival time of the modules to the yard depends on the work load and capacity of the assembly yard. Moreover, the schedule in the yard should be adjusted based on the lifting schedule in the yard. For example, if a module is delivered to the site and then there is no crane to lift the module, it has to be shipped again to storage, incurring extra cost for transporting, loading, unloading and storing the module. Therefore, it would save on costs if the module was sitting in the yard and the effort in the yard was put towards another module more urgently needed on the construction site.
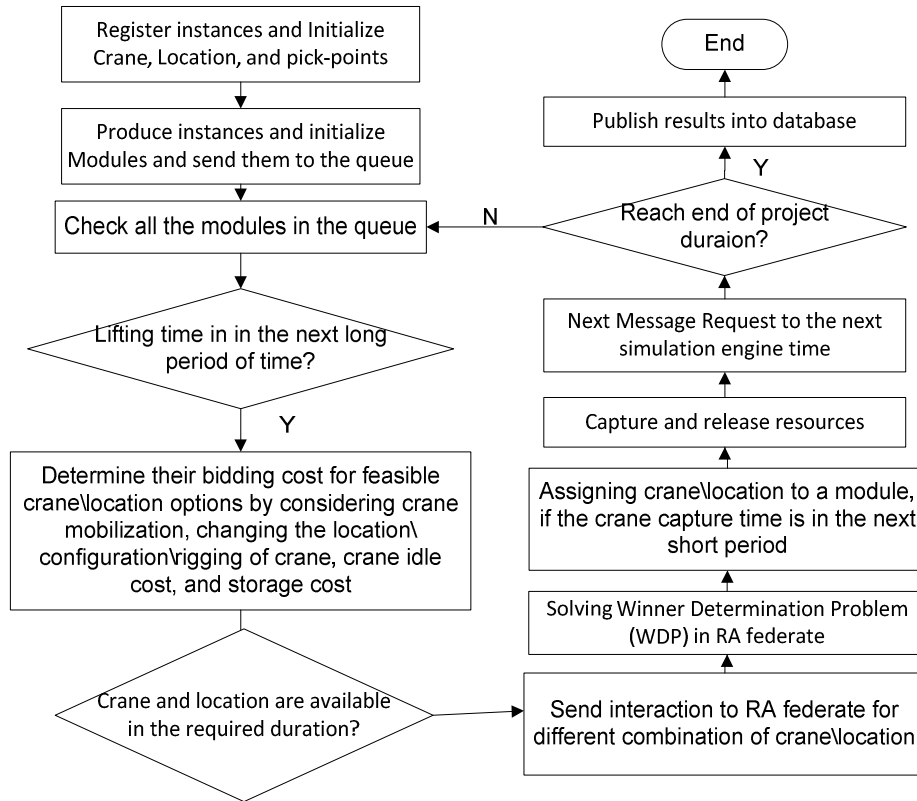
Figure 6: Typical site manager process

8 CALENDAR FEDERATE

The main role of this generic federate is to take into consideration national holidays and long weekends, as well as the number of working hours and overtime hours during the project. This federate provides a form, shown in Figure 7, to input the working hours and overtime hours during the week. This form also enables the simulator to determine the holidays (e.g., long weekends) during the project. The calendar federate synchronizes federate time with calendar time. Thus, all federates that are interested in advancing time according to the calendar can register for the updates of this federate. In practice, there is also the potential of having two different instances of this federate to provide two different calendars in the federation for different activities. For example, the working hours and shifts involved in pouring concrete may differ from the ones for piping activity.

Figure 7: Calendar federate

9 RESOURCE ALLOCATION FEDERATE

This federate is designed to act as an auctioneer to allocate resources among the bidding agents based on the Multi Agent Resource Allocation (MARA) structure. MARA is an excellent structure for allocation resources among different agents and finding a feasible or optimum resource allocation procedure. MARA is not any specific algorithm; however, it allows employing various combinatorial optimizations algorithms. In the centralized allocation approach of MARA, a single element (i.e., auctioneer) decides on the allocation of resources among agents, once the agents submit their preferences over alternative allocations, referred to as bidding stage (Chevaleyre et al. 2006). Then the auctioneer makes the final assignment by solving the Winner Determination Problem (WDP).

The Resource Allocation (RA) federate is supposed to solve the Winner Determination Problem (WDP) in the combinatorial optimization in a generic format. Currently this federate can allocate two types of resources (e.g., crane and location as the resources in the site construction federate) to a number of bidding agents based on a greedy algorithm or ascending-auction algorithm. This federate can easily be expanded to allocate n type of resources to several agents using a combinatorial optimization. Figure 8 shows the schematic view of communication between the RA federate and other simulation federates (e.g., module yard federate) in the industrial federation. Initially, this federate operated through a database, but currently it works through interaction. This federate currently works with the module yard federate as well as the site construction federate (Taghaddos 2010).

The RA federate is called from the module yard federate and site construction federate on a regular basis to maximize the social welfare of the bidding agents. Moreover, the module yard federate inherits from the Resource Allocation Base (RAB) federate to automate its communication with the RA federate. Figure 9 shows results in the RA federate for one of the auctions of the site construction.
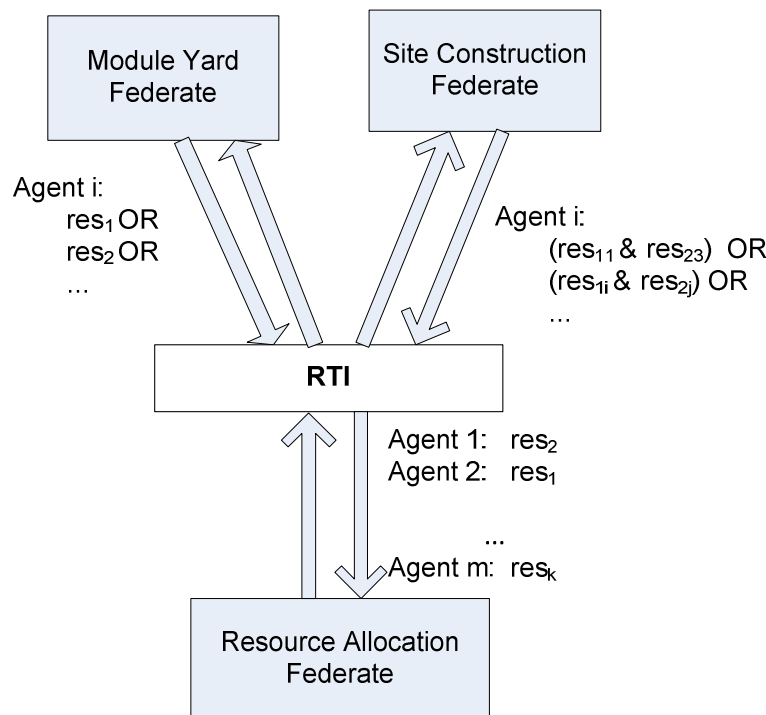

Figure 8: Communication between RA and simulation federates

Bidding Stage			
ID	Resource 1	Resource 2	Price
Module 101	Crane 1	Loc 2	6000
Module 101	Crane 2	Loc 3	6300
Module 102	Crane 1	Loc 5	4500
Module 101	Crane 2	Loc 5	7500

Allocation Stage		
ID	Resource 1	Resource 2
Module 101	Crane 2	Loc 3
Module 102	Crane 1	Loc 5

Operation:	Min

Federation Name:	Site Construction

Figure 9: Resource allocation federate

10   VISUALIZATION FEDERATE

The visualization federate helps to visualize the assembly process in the module yard as well as in the site interactively, while the module yard and site construction federates are running. Consequently, this federate assist the different parties to have a common understanding of the field processes and operations. This federate is developed using Blender, an open source environment that can be integrated with the .NET framework. The Blender gaming engine updates a module location after receiving the respective message from the site construction simulation. The viewer federate then sends reflected attribute values to the visualization model, which automatically updates the site during run time, putting each module in its predefined location. This federate has been recently replaced by a 3D visualization federate, which is under development.

11   BENEFITS OF COSYE FOR INDUSTRIAL CONSTRUCTION MODELING

Within the COSYE environment, large scale industrial construction was decomposed into several federates developed by different individuals based on their interest and expertise. Decomposition saves development time and effort and helps the developer to focus on a portion of the process, while the interaction with other industrial construction federates ensures comprehensive simulation of the entire industrial construction. Decomposition also makes the reuse of other simulation components easier; supportive federates such as weather and calendar can be built once and be reused with other federations interested in the same functionality. Also, because decomposition happens both on a development and an execution level, the execution of simulation components on different engines provides sufficient computing capacity for execution of the entire federation.

This decomposition also allows independence of the federates. Each of the industrial construction federates, according to their needs, could use different simulation world views with different time scales. In industrial construction mostly the process-interaction discrete event simulation world view provided by Simphony.NET services was used; however, for resource allocation federate artificial intelligent was employed based on the decent structure of Multi Agent Resource Allocation (MARA). With regard to the time scale, in the shop fabrication process, the task duration might be on the order of seconds; however, in other federates larger time scales have been used. As may be imagined, this eases development considerably.

12   FUTURE UPDATES: COSYE MODELING SERVICES

At present, the only way to develop COSYE federates is by writing C# or Visual Basic code within the Visual Studio environment. This makes development in COSYE difficult when compared to a visual modeling environment such as Simphony. Recently, work has begun on a software component called COSYE Modeling Services which will act as a bridge between the COSYE and Simphony platforms. The idea is to create a set of Simphony modeling elements that provide access to the various HLA services. These elements would be interoperable with the existing Simphony General Template. Federate developers would then be able to create federates visually by using the General Template to model the internal discrete event processes of their federates and the new set of elements provided by COSYE Modeling Services to coordinate communication with the RTI. This work presents many challenges, not least of which is the fact that Simphony was not developed with HLA compliance in mind.

CONCLUSION

The industrial construction process is a large-scale project with several supply chains involved including drafting, material supply, shop fabrication and module assembly and site construction. Modeling the entire system at a proper level of detail cannot be achieved using traditional construction simulation modeling tools. In this project, High Level Architecture (HLA) was employed to decompose the simulation model into smaller and more manageable components, known as federates. These federates are developed independently and connected together in the Construction Synthetic Environment (COSYE), which is an HLA-based simulation environment developed at the University of Alberta.

Currently the industrial construction federation includes main federates (i.e. procurement, fabrication shop, module yard, site manager) and some supportive federates (i.e. calendar, resource allocation, visualization). The supportive federates may serve one or a number of federates. For instance, resource allocation federate currently serves the module yard federate. This system is successfully implemented and will continue to be enhanced with further details in the near future, as will the COSYE environment.

REFERENCES

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

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

Chang, D. Y. 1986. *RESQUE: A Resource Based Simulation System for Construction Process Planning*. Ph.D. Dissertation, The University of Michigan, Ann Arbor, MI, USA.

Chevaleyre, Y., P.E. Dunne, U. Endriss, J. Lang, M. Lemare, N. Maudet, J. Padget,, S. Phelps, J.A. Rodruez-Aguilar, and P. Sousa. 2006. Issues in multi-agent resource allocation. *Informatica*, 30(1), 3-31.

Defense Modeling and Simulation Office (DMSO). 1999. High Level Architecture: Federation Development and Execution Process (FEDEP) Model.

El Diraby, T., B. Fies, and C. Lima. 2003. An ontology for construction knowledge management. Canadian Society for Civil Engineering - 31st Annual Conference: 2003 Building our Civilization, Montreal, H3H 2R9, Canada, Moncton, NB, Canada, 1949-1956.

Hajjar, D., and S.M. AbouRizk. 1996. Building a special purposes simulation tool for earth moving operations. In *Proceedings of the 1996 Winter Simulation Conference,* ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, 1313-1320. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Halpin, D. W. 1977. CYCLONE - Method for modeling job site processes. *ASCE J Constr Div*, 103(3), 489-499.

Kuhl, F., R. Weatherly, and J. Dahman. 1999. *Creating Computer Simulation Systems: An Introduction to the High Level Architecture*, Englewood Cliffs, NJ: Prentice Hall.

Lluch, J. F. 1981. *Analysis of Construction Operations Using Microcomputers*, Ph.D. Dissertation, Georgia Institute of Technology, Atlanta, GA, USA.

Martinez, J. C. 1996. *STROBOSCOPE: State and Resource Based Simulation of Construction Process*, Ph.D. Dissertation, University of Michigan, Ann Arbor, MI, USA.

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

Taghaddos, H. 2010. *Developing a Generic Resource Allocation Framework for Construction Simulation*, Ph.D. Dissertation, University of Alberta, AB, Canada.

Taghaddos, H., S.M. AbouRizk, Y. Mohamed, and U. Hermann. 2010. Simulation-based multiple heavy lift planning in industrial construction. *Proceedings of the 2010 ASCE Construction Research Congress (CRC)*.

Taghaddos, H., S.M. AbouRizk, Y. Mohamed, and U. Hermann. 2009. Integrated simulation-based scheduling for module assembly yard. *Proceedings of the 2009 ASCE Construction Research Congress (CRC)*, 1270-1279.

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

Wang, P., Y. Mohamed, and S. M. AbouRizk. 2005. Production-based large scale simulation modeling for construction projects, 33rd CSCE Annual Conference 2005, Canadian Society for Civil Engineering, Montreal, H3H 2R9, Canada, Toronto, ON, Canada, 118-1.

AUTHOR BIOGRAPHIES

SIMAAN M. ABOURIZK holds an NSERC Senior Industrial Research Chair in Construction Engineering and Management at the Department of Civil and Environmental Engineering, University of Alberta, where he is a Professor in the Hole School of Construction. He received the ASCE Peurifoy Construction Research Award in 2008. His email address is <abourizk@ualberta.ca>.

YASSER MOHAMED is an Assistant Professor in the Hole School of Construction at the Department of Civil and Environmental Engineering, University of Alberta.

HOSEIN TAGHADDOS is a Construction Coordinator for PCL Industrial Management Inc. in Edmonton, Alberta and recently received his Ph.D. from the Hole School of Construction in the Department of Civil and Environmental Engineering at the University of Alberta.

FARZANEH SABA is a Ph.D. student at the Hole School of Construction in the Department of Civil and Environmental Engineering at the University of Alberta.

STEPHEN HAGUE is the Senior Programmer for the Hole School of Construction, Department of Civil and Environmental Engineering, University of Alberta.