A Distributed Simulation Model of the Maritime Logistics in an Iron Ore Supply Chain Management

Afonso C. Medina[1], Luis G. Nardin[2], Newton N. Pereira[1], Rui C. Botter[1] and Jaime S. Sichman[2*]
[1]*Centro de Inovação em Logística e Infraestrutura Portuária, Universidade de São Paulo, São Paulo, Brazil*
[2]*Laboratório de Técnicas Inteligentes, Universidade de São Paulo, São Paulo, Brazil*

Keywords: Distributed Simulation, Maritime Logistics, Iron Ore Industry, Supply Chain Management.

Abstract: Supply chain management (SCM) has increased its importance in the last decades, accordingly demanding new approaches to support its decision making processes. Simulation has been advised as an adequate approach for fulfilling such demand. However, develop monolithic simulation models representing the whole supply chain can be costly and time consuming. In the iron ore supply chain in which the seaports have the same features, the use of generic models and distributed simulation may be a real alternative in order to reduce the development time and costs. This paper presents a distributed simulation model of the maritime logistics in an iron ore supply chain applied to support fleet management decisions. Such model was used to perform an experiment in order to determine the maximum possible cargo volume supported by a ship fleet.

1 INTRODUCTION

Supply chain is a phenomenon that can be defined as a set of three or more entities directly involved with the flow of products, services, finances, and information from a producer to a customer (Mentzer et al., 2001). Such phenomenon became more critical and impacting because of the globalization process that has been happening in the last decades. Consequently, global active companies realized that the efficiency of their own businesses is highly dependent on the collaboration and coordination with their suppliers as well as with their customers (Hieber, 2002). Therefore, they identified the need of implementing a more efficient and effective supply chain management (SCM).

Among different industries, the mining industry is one that depends mostly on SCM. Generally, because its raw materials are placed in remote locations and their customers are geographically distant from the production centers. For instance, the biggest iron ore mines are located in African countries, Australia and Brazil. However, China is the main customer consuming more than 50% of the world's iron ore production (Hoyt et al., 2007). In addition, the iron ore supply chain has some characteristics similar to other commodities, such as a few numbers of products, a high cargo volume, high lead times and low price, which makes the pipeline management a key factor of success (Beresford et al., 2011).

Hence, the maritime transport is a key component in the whole mining industry supply chain. Thus, mining companies face several complex decisions in order to improve their maritime logistics efficiency. Among the available techniques supporting such decision making process, simulation is one of the most adequate as it is capable of providing *what-if* analysis and answer quantitatively questions that typically arise in these situations (Terzi and Cavalieri, 2004). Moreover, it handles better complex scenarios in contrast to optimization tools as mathematical programming (Ingalls, 1998).

Accordingly, many mining companies have been developing simulation models focusing on specific maritime logistics nodes in order to tackle particular problems, such as terminal capacity (Bugaric et al., 2012), flowability of products from/to the terminal (Everett, 1995), and closed-loop maritime transportation (Silva et al., 2011). Nonetheless, a model with extended boundaries shall be developed incorporating material and information flow among all the nodes involved in the maritime logistics in order to allow drawing more accurate conclusions (Jain et al., 1999).

Basically, supply chain simulation uses two different approaches: monolithic or distributed (Taylor et al., 2002). In the former, the whole chain is rep-

---

[*] Jaime S. Sichman is partially supported by CNPq and FAPESP, Brazil.

Medina A., Nardin L., Pereira N., Botter R. and Sichman J. (2013).
A Distributed Simulation Model of the Maritime Logistics in an Iron Ore Supply Chain Management.
In *Proceedings of the 3rd International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, pages 453-460
DOI: 10.5220/0004488504530460

resented in a single model. In the latter, the supply chain is represented by several models, each one corresponding to a different part of the chain. Therefore, because the companies have already invested on the development and validation of simulation models to tackle specific problems, it is more reasonable to reuse them through the distributed simulation approach (Fujimoto, 1999).

This paper presents a distributed simulation applied to the maritime logistics of an iron ore supply chain in order to support fleet management decisions. The underlying distributed model is composed of several nodes representing the seaports and the navigation routes. Each seaport is represented by a specific model, which is derived from a generic port model able to represent any iron ore seaport from the product's reception via railways to their load and dispatch through the vessels. All these models interact among themselves through a navigation model supported by a framework based on the High Level Architecture (HLA) (Dahmann et al., 1998; IEEE, 2000). The model was applied to simulate real-world scenario in order to evaluate the fleet handling capacity.

This paper is organized in six more sections. Section 2 contextualize and motivates this work, followed by a brief review of supply chain management supported by simulation in Section 3. Next, the generic port and the navigation models are detailed in Section 4, which integration are described in conjunction with the specification of the implemented software architecture in Section 5. Section 6 describes the validation models' process and an experiment performed using the validated model. Finally, Section 7 concludes the paper and provides some possible future works.

2 MOTIVATION SCENARIO

The global iron ore industry is dominated by few players and the Brazilian mining company Vale[1] is the world's largest producer and second largest exporter (Choenni et al., 2011). In order to support its operations, Vale owns several seaports in Brazil, such as Port of Tubarão (TU) - Espírito Santo, Port of Ponta da Madeira (TMPM) - Maranhão, Port of Septiba (CPBS) and Port of Itajaí (TIG) - Rio de Janeiro.

Since 2007, Vale has been developing discrete event simulation models in partnership with research laboratories in order to have a decision support tool for terminals flow capacity management. Particularly, they have developed a Generic Port Model (described in more detail in Section 4.1) able to represent any iron ore seaport, which was validated with real operations data and is used nowadays to support the company's Director of Ports decisions.

Back to 2008, the maritime transport costs was constantly varying[2] and Vale was paying a high price to transport the iron ore to its customers. Thus, in an attempt to take an active role in the maritime logistics and reduce its vulnerability, the company decided to act as shipowner transporting part of its own production using its own ship fleet. As a consequence, it contracted the construction of 35 Valemax class ships, each able to carry up to 400,000 tons of cargo (Pereira and Brinati, 2012). Along with it, the company envisioned the necessity to have a decision support system to assist its integrated actions concerning the new ship fleet and the remaining supply chain. In this scenario, the ship fleet functions as links interconnecting the seaports. Nevertheless, such links need to be flexible in the sense that they may change in order to represent more realistically the maritime transport system dynamics.

Since the simulation approach fulfills these maritime transport dynamism and Vale had already developed a Generic Port Model that represents any of its seaports, the more adequate solution would be integrate all its models through the distributed simulation approach instead of developing a new monolithic simulation model representing the whole supply chain.

3 LITERATURE REVIEW

SCM has been studied extensively in the last decades because of its increasing impact on enterprise operations. Generally, the approaches used to address their complex formulations are physical experimentations, mathematical programming and simulation (Thierry et al., 2010). Usually, physical experimentation is scope limited due to technical issues and budget availability. The mathematical programming approach uses methods, such as linear, mixed integer linear and dynamic programming with an objective function set to minimize cost or maximize profit. Despite its wide use in the supply chain's optimization (see (Sarmiento and Nagi, 1999) for a review of its application on different production-distribution problems), it becomes impractical in complex scenarios involving stochasticity aspects and dynamically interconnected nodes such as existent in supply chain. On the other hand, the simulation approach uses different methodologies to build models capable to represent the complex supply chain's nodes interconnections,

---

[1] http://www.vale.com.

[2] Source: Maritime Transport Cost Database (http://stats.oecd.org/).

which are then used for replicating and analyzing the system's behavior.

Among these approaches, simulation can better represent the SCM inherent complexities. For this reason, it has been applied in several supply chain studies. Ingalls (Ingalls, 1998) discusses advantages and disadvantages of using simulation in SCM and stresses its importance to deal with the random signals caused by demand forecast.

Specifically in the maritime logistics, simulation has been used for a long time in the study of port systems (Wadhwa, 1992; Hassan, 1993; Botter et al., 1998; Bugaric et al., 2012). Nonetheless, these studies are focused on dimensioning and planning port's capacity, while maritime logistics is a much more complex system that involves other components other than ports. Terzi and Cavalieri (Terzi and Cavalieri, 2004) present a survey of 80 works related to supply chain simulation and they classify them in two major paradigms: *local* and *parallel or distributed* simulation. The *local* simulation paradigm consists in the use of a single model to represent all the supply chain and it usually runs in a single machine. On the other hand, the *parallel or distributed* paradigm considers the existence of several models representing a more complex system and it runs in separate machines. They advocate that SCM would have several advantages by applying the distributed simulation paradigm, such as no need for new development as the models of specific nodes are already available and the possibility of having these models running geographically distributed. Moreover, in the case of complex systems made up of autonomous entities, multiagent-based simulation techniques may be used (Marietto et al., 2003).

Although the recognition of its advantages, most of the literature on parallel or distributed simulation focus on frameworks to parallelize or distribute the simulation, yet just a few present the application of the paradigm in real scenarios (Terzi and Cavalieri, 2004). Among these few, Duinkerken et al. (Duinkerken et al., 2002) is the only one that reports the application of distributed simulation in the study of the maritime logistic. However, the main focus of their work is still to demonstrate the benefits of using a distributed structure in the transparency and maintainability of the simulation model. Hence, to the best of our knowledge, this work is the first one applying the distributed simulation approach in the analyses of a real scenarios supporting fleet management decisions in the maritime logistics domain.

4 SIMULATION MODELS

This section presents the simulation models used in the development of the maritime logistics distributed simulation model. In Section 4.1, it is presented the Generic Port Model capable to represent any iron ore seaport. Next, the Navigation Model responsible for simulating the behavior of vessels navigation is presented in Section 4.2.

4.1 Generic Port Model

Generic models are a well know approach to develop simulation models (Robinson et al., 2004; Monks et al., 2009; Mackulak et al., 1998; Lung et al., 1994). They are based on the concept that some class of models may represent a wide range of similar scenarios, hence they could be used on a frequent and recurrent basis. In contrast, a particular model is developed to answer questions for a very specific system and they cannot be easily reused, not even in similar scenarios. Usually, a generic model is more expensive to develop in the short-term than a particular model, but as the former has more chance to be reused in new projects, it generally provides a better return on investment in the long-term (Doss and Ülgen, 2004).

Envisioning this long-term benefit, the Generic Port Model (GPM) was developed based on the generic models approach as the result of a three years project. Such development involved interviews, workshops, and meetings with professionals of the maritime sector, more specifically, specialists of seaport sector. Its main purpose is to be a comprehensive simulation model to represent any iron ore seaports operational behavior. As a consequence, it is structured in a way to support decisions regarding terminals capacity. Such aim is achieved through the possibility to represent different iron ore seaports configurations and scenarios, including seaport's equipments and stockyards configurations, company's demands and products, as well as vessels and ground vehicles arrivals.

The GPM is structured in 4 subsystems:

- *Ships Arrival* – Controls all the ships arrival processes, such as cargo required and estimated time to arrival. Moreover, it is also responsible for controlling all processes from the actual vessel arrival in the bay up to the begin of the loading operation, such as tide control, berth allocation, navigation through the channel to the harbor, berthing, and vessel pre-loading and loading operations.
- *Cargo Transfer* – Performs the vessels loading operation including the selection and transportation of products from stockyard to the vessels in

a guaranteed maximum loading rate (or minimum operation time).

- *Storage* – Represents all resources used to store the products at stockyard and, if present, the performance of some semi-industrial operations like screening, pellet production or products blending.
- *Dispatch and Ground Reception* – Responsible for cargo reception through trains guaranteeing mass balance between arrived and dispatched cargo. It generates the trains arrival sequence and control the processes from full train arrival to empty train departure. Additionally, it controls cargo transfer from train to stockyard considering car dumpers and stackers. It attempts to ensure the best disposition of the products at the stockyard in order to guarantee the maximum ships loading rate.

In order to guarantee the model reuse, the subsystems were modeled in a way that they may be configured for use in a wide range of situations. Table 1 lists a partial set of the input and the output parameters divided by subsystems in order to demonstrate the flexibility of the model.

Strategically, its development started with a conceptual model of a simpler terminal. Along the project, more complex features were incorporated into the model considering an iterative process in which every new feature incorporated was validated using real data. This iterative process allowed the incorporation of a great degree of complexity into the model requiring a reduced validation time at each improvement.

Nowadays, the model is used as a decision support tool in a recurrent basis for answering questions concerning capacity, new capital investments and design of new port systems.

4.2 Navigation Model

The Navigation Model (NM) was developed to represent the vessels movement behavior and available routes. This model performs an interconnection among all the instantiated GPMs in the system. The vessels are represented as entities that flow from one GPM to another through specific routes configured in the NM. Since the iron ore production usually flows from an export to an import seaport and the vessels return empty from the import seaport to the export seaport, the NM considers the existence of three classes of routes: (i) a route that interconnects an export to an import seaport, (ii) a ballast route that interconnects an import seaport to a *check point* and (iii) a route that interconnects a *check point* to an export seaport.

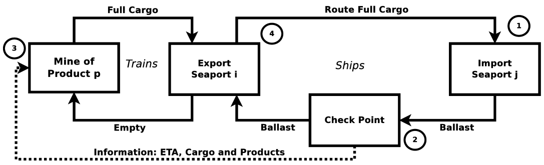

Figure 1: Process at iron ore pipeline.

In addition to the vessels and routes, the NM has an additional entity named *check point*, which represents the point that a ballast vessel (ship without cargo) receives the definition of its next export and import seaport destinations as well as the products to transport.

Figure 1 presents the steps followed by a vessel beginning in the moment it leaves an import seaport up to the moment it leaves an export seaport towards its assigned import seaport. The sequence is:

1. At the import seaport, the vessel is unloaded and after its unberthing, it starts its navigation in ballast to the *check point*.
2. At the *check point*, the vessel is allocated to its next export and import seaports and the product(s) to transport are assigned (Algorithm 1 describes the decision algorithm). At this moment, the mine is informed about the vessel's Early Time Arrival (ETA), cargo and assigned product(s).

Algorithm 1: Allocate vessel to route.

```
Vessel $v$ arrives at Check Point
$MinGAP \leftarrow$ MINIMUM INTEGER
for all Route ($r$) that $v$ can operate do
    $i \leftarrow r$ export seaport
    $j \leftarrow r$ import seaport
    for all Product $p$ possible on $r$ do
        Estimate the product gap using Equation 1
            $gap(p,r) = (D(t)*(1+\Delta)-A(t))/(D(t)*(1+\Delta))$    (1)
        where
          $t$ = simulation time
          $D(t)$ = demand of $p$ at $j$ until $t$
          $A(t)$ = attempt demand of $p$ at $j$ until $t$
          $\Delta$ = accepted gap
        if $gap(p,r) > MinGAP$ then
            Estimate gain using
              $Max\ gain(p,r)$ = vessel cargo capacity
            $VRP \leftarrow v$ to $r$ with product $p$
            $MinGAP = gap(p,r)$
        end if
    end for
end for
return $VRP$
```

Table 1: Input and output parameters of the GPM's main subsystems.

Subsystem	**Input parameters**	**Output parameters**
Ship arrivals	- Demand by destiny and product - Ships class distribution - Time to stock the cargo at the yard before the ship arrival - Berthing/unberthing times - Channel (capacity, times, tides)	- Ships queue, turnaround, berthing/unberthing times - Waiting times - Export demand/ships attended - Dispatch/demurrage calculation
Cargo transfer	- Ship unloaders (capacity and maximum rates by product) - Berths - Layout disposition (berth x yard and resources x berths)	- Occupancy of resources (reclaimers, ship loaders, berths, lines) - Cargo transfer by product, line and berth
Storage	- Number of yards, piles, capacity by product - Layout disposition, access to berths and car dumpers - Screening (number, production rates, yards) - Pellet production (number, production rates, yards)	- Occupancy of resources (yards, stacker, reclaimers, lines, pellet unit)
Dispatch and ground reception	- Number of car dumpers - Rates by product - Setup times - Railway yard capacity - Trains capacity by product	- Occupancy of resources (car dumpers, stacker) - Total cargo transferred to yards - Trains queue time

3. In the mean time between the vessel arrival to the export seaport, the mine dispatches the trains to attend the ETA considering also that the total cargo should be available at the seaport in advance to the beginning of vessel's loading operation.
4. Once at the export seaport, the vessel is loaded with the assigned products and navigates to its designed import seaport.

Basically, Algorithm 1 checks the fulfilled demands on each import seaport up to the current simulation time. This is performed by calculating a parameter named gap. For any product p and any route r, $gap(p,r)$ represents the difference between the product demanded by the import seaport in the period and the total product p already delivered to that seaport. Thus, among all possible routes, the *check point* allocates the ballast ship to the route in which the gap falls mostly behind. In order to choose the best option, the *check point* also considers the constraints about demand and vessels capacity, export seaport product availability and allowance of vessels in the seaports. However, if no reduction is identified on any products gap, then the vessel waits at the *check point* and verifies periodically until a gap is identified.

5 DISTRIBUTED SIMULATION

The distributed simulation model performs the interconnection of several General Port Models and one Navigation Model. It is structured according to the High Level Architecture (HLA) (IEEE, 2000), which is a general purpose architecture to support distributed simulation allowing the communication among simulation models running on distributed heterogeneous computational platforms. A HLA distributed simulation may have one or several *Federations*. Each *Federation* is composed of one or more *Federates*, each representing a simulation model. The *Federates* are interconnected through the *Runtime Infrastructure* (RTI), which provides common communication and synchronization services to them. In order to architect our distributed simulation adherent to the HLA, two kinds of *Federates* are defined: *Arena Federate* and *Distributed Simulation Coordinator*.

5.1 Arena Federate

The *Arena Federate* is the architectural module that encapsulates a single simulation model in the distributed environment. It is composed of a *Federate* and an *Arena Adapter*. The *Federate* corresponds to the actual simulation model, which is represented by a GPM running in the Arena® discrete-event simulator. The *Arena Adapter* enables the communication between the *Federate* and the other components of the distributed environment. Besides, the *Arena Adapter* implements two HLA components: *Federate-Ambassador* that implements the communication interface with the *Federate* and *RTI-Ambassador* that enables the communication between the *Federate* and the RTI.

The communication with Arena® is performed through DLL (Dynamic Link Library) functions developed in C++ language. The set of Arena®'s communication functions implemented are: *initProcess*, *shutdownIPC*, *readIPCQueue*, *writeIPCQueue*, *userInitializeMaxTimeAdvance* and *userGetMaxTimeAdvance*.

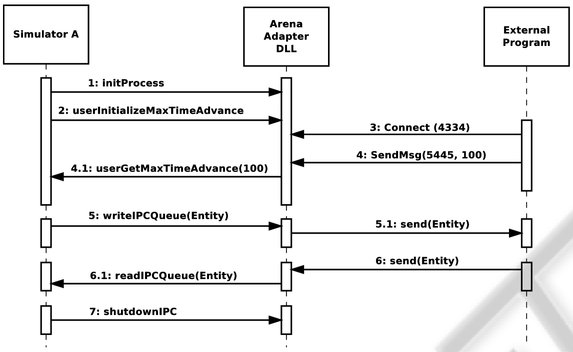

Figure 2: Communication sequence diagram.

Figure 2 depicts the communication sequence between Arena® and an external program. The sequence starts with the initialization of *Simulator A* that calls the *initProcess* and *userInitializeMaxTimeAdvance* DLL functions (messages 1 and 2) in order to initialize, respectively, the communication and the *Simulator A* clock. Next, it waits the signal from the *External Program* to start the simulation execution (message 3). After receiving the signal, the model is executed synchronized to the simulation time received from the *External Program* (messages 4 and 4.1). During the simulation, if *Simulator A* wants to send an entity to another simulator, it calls the element TASKS, which triggers a call to the *writeIPCQueue* DLL function that receives the entity data and send it to the *External Program* (messages 5 and 5.1). In order to transfer such data to another simulator, the *External Program* sends the data to the target's *Arena Adapter*, which triggers the *readIPCQueue* DLL function to send the new entity data to the model's ARRIVALS element. Once received, the entity is created in the simulation (messages 6 and 6.1). At the end of the simulation, *Simulator A* calls the *shutdownIPC* DLL function that finalizes the communication and the simulation (message 7).

5.2 Distributed Simulation Coordinator

The *Distributed Simulation Coordinator* (DSC) is a *Federate* developed in Java programming language and it has the following functions: (i) store the scenarios information, (ii) coordinate the transfer of ships (entities) from one *Arena Federate* to another, and (iii) synchronize the clock among the several *Arena Federate*. Each *Federation* has only one DSC and it is structured in an architecture composed of layers: *High Level Architecture*, *Distributed Simulation Planning* and *Graphical User Interface*.

The *High Level Architecture* layer enables DSC to communicate with the *Arena Federates* in the *Federation*. In this layer, the communication and clock synchronization mechanisms are implemented. Among these functions, the synchronization is the most critical, which in this work was implemented as a conservative clock synchronization.

The *Distributed Simulation Planning* performs all the route planning functions according to the configured demands in the scenario. It also controls the restrictions imposed concerning the possibility to allocate a ship to a route.

The *Graphical User Interface* (GUI) layer allows the users to interact with the system inputing data and extracting simulation results. This layer is supported by a database responsible to store the configured scenarios and output simulation results.

5.3 Distributed Integrated Model

The *Distributed Integrated Model* is an integration of all the components described in the previous sections, which may be configured differently in order to support different scenarios. In this work, the scenario used to simulate the seaports network and fleet is comprised of one *Federation* composed of 1 *Distributed Simulation Coordinator* and 7 *Arena Federates*, 4 representing export seaports using the Generic Port Model (Ports A, B, C and D), 2 representing the import seaports (Ports E and F) and 1 representing the navigation system which uses the Navigation Model. This integrated model was used to perform the experiments described in the next section. Therefore, it is worth noting that since this model can be configured differently in order to represent different scenarios in the same context of maritime logistics, it is reusable.

6 EXPERIMENT

In order to perform the experiment, the developed distributed model was first validate using real data, which process overview is provided in Section 6.1. Then, the proposed experiment was conducted and its results analyzed as described in Section 6.2.

6.1 Validation

The validation process was carried out considering a base scenario and one year of real operations data for its calibration. The base scenario was composed of the components described in Section 5.3 and 28 vessels ranging from 89,000 DWT to 270,000 DWT. The validation process was performed by comparing the output of 10 simulation runs against the real data output taking into account the following parameters: imported and exported total cargo by product, number of berthing, number of unberthing and queue time. The model was assumed to be valid when the error of the simulated and real average parameters values was under 2%.

6.2 Simulated Scenario

Based on the validated model, an experiment was performed using an IBM Blade Center S[3] (1 Blade 2-processors Intel Xeon 2.0 GHz) configured with 04 virtual machines (1 processor 2.0 GHz and 2 GB memory) running Rockwell Arena® v12 on Windows XP and 03 PCs (1 processor Intel iCore i7 2.93 GHz and 4 GB memory) running Rockwell Arena® v12 on Windows 7.

The experiment aimed to identify the maximum transport capacity of a fictitious ship fleet composed of 32 ships ranging from 150,000 DWT to 400,000 DWT and considering 4 export seaport (Ports A, B, C and D), and 2 import seaport (Ports E and F). The strategy used to achieve this objective was to perform several simulations beginning with a small supply and demand cargo and increasing them proportionally until the ships waiting time on the *check point* was zero. The latter condition indicates that during the whole simulation none ship was idle, which means that the maximum ship fleet transport capacity was reached. Table 2 presents the supply and demand cargo in which the condition described was achieved.

Using this experiment scenario, the simulation was executed 10 times and it was identified that the ship fleet was able to transport 20,236,144 tons of iron ore as presented in Table 3.

Table 2: Cargo supplied and demanded per seaport.

Seaport	Supply (tons)	Demand (tons)
Port A	4,398,195	–
Port B	11,194,190	–
Port C	7,055,643	–
Port D	18,393,006	–
Port E	–	39,187,197
Port F	–	2,607,837

Table 3: Maximum ship fleet transport capacity.

Cargo	Quantity (tons)
Unloaded	20,236,144
Loaded	23,386,314
In Transit	3,150,170

7 CONCLUSIONS

Simulation has been advised as an adequate approach to analyze SCM behaviors. However, develop monolithic models representing the whole supply chain can be costly and time consuming in the long-term. In the iron ore industry in which the seaports have the same features, the use of generic models and distributed simulation can be considered a real alternative in order to reduce its modeling development time.

Hence, this work presented a Generic Port Model, a Navigation Model and a Distributed Integrated Model in order to develop a maritime logistics distributed simulation based in the High Level Architecture. The main identified technical challenges along the distributed simulation implementation were the interoperability among the simulators and the clock synchronization tasks.

The developed system was applied in the fictitious scenario in order to determine the maximum transport capacity of a specific ship fleet. The results showed the usefulness of the model as a tool to support decision making and the applicability of distributed simulation. Additionally, its usefulness for decision making training of operational, tactical and strategical personnel was identified during its development.

As future work, we intend to carry out further simulations incorporating other supply chain nodes, such as distribution center, mine production and railway. Specifically concerning the distribution centers, we expect to study an interesting aspect which is how the service level of the system – measure by queue time at its output side – is affected by the number of vessels at the loop circuit between import seaports and the distribution centers and estimate the trade off between the number of ships and services level at the distribution center.

---

[3] http://www-03.ibm.com/systems/bladecenter/hardware/chassis/blades/index.html.

REFERENCES

Beresford, A., Pettit, S., and Liu, Y. (2011). Multimodal supply chains: iron ore from australia to china. *Supply Chain Management: An International Journal*, 16(1):32–42.

Botter, R. C., Mendes, A. B., and de Souza, R. F. (1998). Simulation model for the redesign of a supply distribution system. In *Proceedings of the 1998 Summer Computer Simulation Conference*.

Bugaric, U. S., Petrovic, D. B., Jeli, Z. V., and Petrovic, D. V. (2012). Optimal utilization of the terminal for bulk cargo unloading. *Simulation*, 88(12):1508–1521.

Choenni, A., Lehmann, J., Sewerski, M., Huber, R., and Kirk, R. (2011). Iron ore: The competition between brazil and australia for asian markets. Technical report, Fundação Getúlio Vargas.

Dahmann, J., Fujimoto, R. M., and Weatherly, R. M. (1998). The DoD High Level Architecture: an update. In *Proceedings of the 30th conference on Winter simulation*, pages 797–804, Los Alamitos, US. IEEE Computer Society Press.

Doss, D. L. and Ülgen, O. M. (2004). A case for generic, custom-designed simulation applications for material handling and manufacturing industries. In *Proceedings of the 2004 Brooks Automation's Worldwide Automation Symposium*.

Duinkerken, M. B., Ottjes, J. A., and Lodewijks, G. (2002). The application of distributed simulation in TOMAS: Redesigning a complex transportation model. In Yücesan, E., Cheng, C.-H., Snowdon, J. L., and Charnes, J. M., editors, *Proceedings of the 2002 Winter Simulation Conference*.

Everett, J. E. (1995). Improving the quality of iron ore shipments. In *International Congress on Modelling and Simulation*, pages 320–325.

Fujimoto, R. M. (1999). *Parallel and Distribution Simulation Systems*. John Wiley & Sons, Inc., New York, US, 1st edition.

Hassan, S. A. (1993). Port activity simulation: an overview. *SIGSIM Simul. Dig.*, 23(2):17–36.

Hieber, R. (2002). *Supply Chain Management: A Collaborative Performance Measurement Approach*. Research reports for industrial practice. vdf, Hochschulverlag AG an der ETH Zürich.

Hoyt, D., Lee, H., and Singh, S. (2007). Rio tinto iron ore: Challenges of globalization in the mining industry. In *Global Supply Chain Management Forum*. Stanford Graduate School of Business.

IEEE (2000). IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules. *IEEE Std. 1516-2000*, 0:i–22.

Ingalls, R. G. (1998). The value of simulation in modeling supply chains. In *Proceedings of the 30th conference on Winter simulation*, pages 1371–1376, Los Alamitos, US. IEEE Computer Society Press.

Jain, S., Lim, C.-C., Gan, B.-P., and Low, Y.-H. (1999). Criticality of detailed modeling in semiconductor supply chain simulation. In *Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future - Volume 1*, pages 888–896, New York, US. ACM.

Lung, C.-H., Cochran, J. K., Mackulak, G. T., and Urban, J. E. (1994). Computer simulation software reuse by generic/specific domain modeling approach. *International Journal of Software Engineering and Knowledge Engineering*, 4(1):81–102.

Mackulak, G., Lawrence, F., and Colvin, T. (1998). Effective simulation model reuse: a case study for amhs modeling. In *Simulation Conference Proceedings, 1998. Winter*, volume 2, pages 979–984.

Marietto, M. B., David, N., Sichman, J. S., and Coelho, H. (2003). Requirements analysis of agent-based simulation platforms: State of the art and new prospects. In Sichman, J. S., Bousquet, F., and Davidsson, P., editors, *Multi-Agent-Based Simulation II*, volume 2581 of *Lecture Notes in Computer Science*, pages 125–141. Springer.

Mentzer, J. T., DeWitt, W., Keebler, J. S., Min, S., Nix, N. W., Smith, C. D., and Zacharia, Z. G. (2001). Defining supply chain management. *Journal of Business Logistics*, 22(2):1–25.

Monks, T., Robinson, S., and Kotiadis, K. (2009). Model reuse versus model development: Effects on credibility and learning. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, pages 767–778.

Pereira, N. N. and Brinati, H. L. (2012). Onshore ballast water treatment: A viable option for major ports. *Marine Pollution Bulletin*, 64(11):2296–2304.

Robinson, S., Nance, R. E., Paul, R. J., Pidd, M., and Taylor, S. J. E. (2004). Simulation model reuse: definitions, benefits and obstacles. *Simulation Modelling Practice and Theory*, 12(7–8):479–494.

Sarmiento, A. M. and Nagi, R. (1999). A review of integrated analysis of production-distribution systems. *IIE Transactions*, 31(11):1061–1074.

Silva, R. C. d. S., Brito, T. B., Botter, R. C., and Pereira, N. N. (2011). Modeling of a closed-loop maritime transportation system with discrete event simulation and multi-criteria decision analysis. In Ao, S. I., Douglas, C., Grundfest, W. S., and Burgstone, J., editors, *Proceedings of the World Congress on Engineering and Computer Science 2011 Vol II, WCECS 2011, October 19 - 21, 2011, San Francisco, USA*, Lecture Notes in Engineering and Computer Science, pages 956–961. International Association of Engineers, Newswood Limited.

Taylor, S., Bruzzone, A., Fujimoto, R., Gan, B. P., Strassburger, S., and Paul, R. (2002). Distributed simulation and industry: potentials and pitfalls. In *Simulation Conference, 2002. Proceedings of the Winter*, volume 1, pages 688–694.

Terzi, S. and Cavalieri, S. (2004). Simulation in the supply chain context: a survey. *Comput. Ind.*, 53(1):3–16.

Thierry, C., Bel, G., and Thomas, A. (2010). The role of modeling and simulation in supply chain management. *SCS M&S Magazine*, 4:1–8.

Wadhwa, L. C. (1992). Planning operations of bulk loading terminals by simulation. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 118(3):300–315.