Proceedings of the 2010 Winter Simulation Conference
B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, eds.

REUSABLE TEMPLATE FOR SIMULATION OF OVERHEAD CRANES INTERFERENCES

Daniel Pablo Paz
Luiz Augusto G. Franzese

Paragontech LOGSIS Engineering
Santa Fe 1219, 6th floor
Rosario, SF 2000, ARGENTINA

ABSTRACT

Overhead cranes are critical equipment in heavy industries, ports and construction. In many cases, they may become bottlenecks for the whole production process. At the phase of plant design, or when evaluating modifications to the processes, it is critical to analyze their behavior, specially the interferences that occur among each other. Considering factors like breakdowns, process time variability, buffer capacities and interferences with other vehicles; linear calculation is not applicable and simulation is required. This paper explains the development of a reusable template for simulating a group of overhead cranes integrating it in a major ARENA model.

1 INTRODUCTION

The study of a group of overhead cranes operating, is normally represented in simulation software using components like network links and transporters. Usually cranes can be easily represented as a vehicles components. Additional logic may be necessary to control the interaction among these vehicles in order to avoid blockages and deadlocks.

Even when some simulation packages allow the simulation of two cranes, and there exist stand alone simulators for cranes interferences, we have not found a solution available for integrating the simulation of a group of overhead cranes into a complex simulation model.

This paper describes the development of a reusable template in ARENA, which can be easily integrated into an existing ARENA model, or a new one. This solution brings these advantages:

- Reusability
- Ease to integrate in a complex simulation model
- Ease to configure and use
- An heuristic control logic respecting a practical tradeoff between deadlock avoidance and efficiency in cranes performance

1.1 Defining the problem

Defining a realistic logic for controlling the interaction between cranes, according to the performance they would have when operated by humans, is the key point of this problem. Initially the analysis of every possible interaction between cranes, and every possible way in which a crane could react to those interactions, brings tips for the first approach. Considering the initial position and destination of each crane, there are typical situations that will generate conflict among cranes, resulting in blockages or interferences.

Simulation packages do not have tools for solving potential blockages or deadlocks; therefore specific logic is necessary to solve those situations and get the simulation running in a realistic way.

We have defined the problem considering the factors:

- Conflict situations
- Crane states
- Corrective actions

1.1.1 Description of typical conflict situations

There is a finite number of potential interferences and deadlock situations; each one requires a specific solution. In the real world the crane operators and the supervisor decide according to variables like: crane position, crane speed, priority of movement, request from a machine, urgency of a specific movement, saturation of a buffer, among others, and human factors.

Situation 1: Cranes traveling in opposite directions

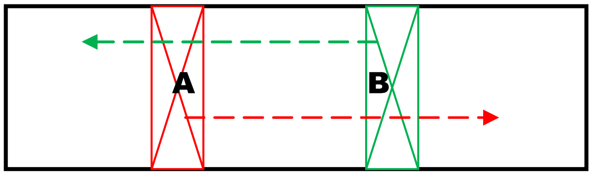

Figure1: conflict situation 1

The crane on the left is trying to get somewhere **behind** the crane on the right, which itself is trying to get to a point **behind** the crane on the left.

Situation 2: Cranes traveling in the same direction

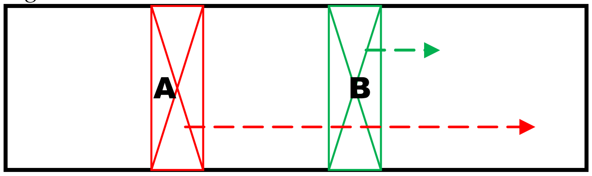

Figure2: conflict situation 2

The crane on the left is moving towards a point further on the right than the point where the crane on the right is heading.

Situation 3: Crane traveling and finding a crane in its way

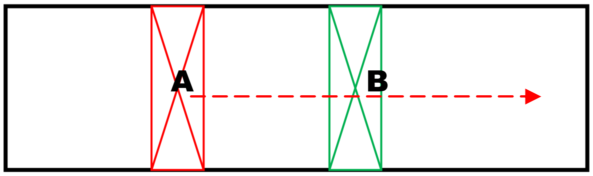

Figure3: conflict situation 3

The crane on the left is moving towards a point **behind** the crane on the right, which is standing still interfering the first one.

Additionally to these basic situations, it is necessary to consider crane's states, in order to evaluate the correct action to solve them properly.

1.1.2 Crane´s States

A single crane may pass trough different states during the simulation. When considering a group of cranes, the problem to analyze is the transition among possible states of the cranes and the interaction among them.

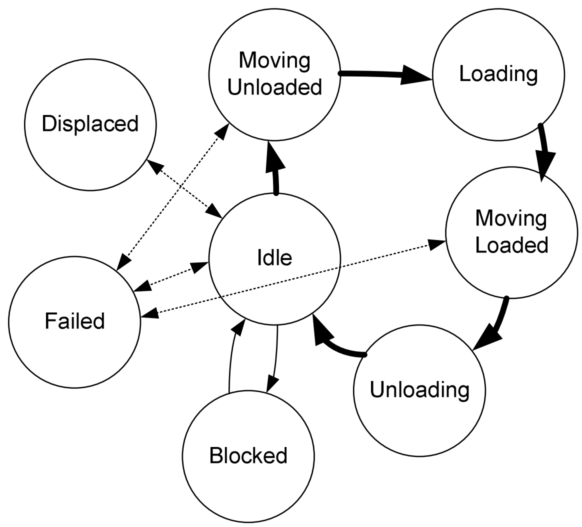

Figure 4: single crane states and normal cycle

For Instance, in situation 1, if the crane A is moving, unloaded, while crane B is moving loaded, a decision could be taken in favor of the latter; or in favor of the unloaded one depending of the chosen criteria. Other approach may be to define crane priorities to solve this situations using a deterministic or probabilistic rule.

In Situation 3, if crane B -waiting on the crane A's way- is idle, the heuristic approach suggest it is advisable to let the crane A displace crane B. In case crane B is in the process of loading, it would be necessary to establish a higher priority to it, forcing crane A to wait until it finishes its operation. At this point, there are different possible decisions: a) crane A waits until cane B finishes loading and then displaces crane B; b) crane A moves to the furthest point near crane B and then decides what to do, according to new cranes B's task.

It is hard to define all the possible interactions among two cranes having a set of possible states, even when there is not an optimal decision a-priori. A heuristic approach is needed to find at least ONE viable solution. When considering a group (i.e. more than two) of cranes traveling from one point to another, loading and unloading, the possibilities grow exponentially.

1.1.3 Description of Corrective Actions

Considering the set of possible conflict situations, the set of crane states, all the combinations among them, and the realistic behavior of cranes operators, it will be possible to define a limited set of corrective actions.

When an interference situation is detected, the crane operator or the plant supervisor has a set of actions to chose from; he will necessary have to set priorities according to a very complex number of decision factors.

The most common corrective actions considered in this model are:

- Make one crane to wait until the other finishes its movement
- Make one idle crane to move backwards in order to allow the other crane to perform its movement
- Move one crane to an intermediate position and wait until the other crane finishes.

There exist other possible solutions like move both cranes coordinated, vary speeds in order to achieve the same points at the same time, itemizing a sequence of tasks, among several others. However real crane operators rarely use those solutions, having limited information about future tasks of the other cranes, and preferring simpler approaches. We assume a set of solutions reflecting the majority of real situations at a realistic level of efficiency.

2 SOLUTION

As seen in the above section, modeling a set of overhead cranes traveling along a single track detailing all their possible reactions under different conflict situations becomes complex as further details are added to describe cranes operation. The solution proposed in this paper deals with both problems presented above: it completely solves the interaction between cranes while it aims for simplicity in *operation assignments*.

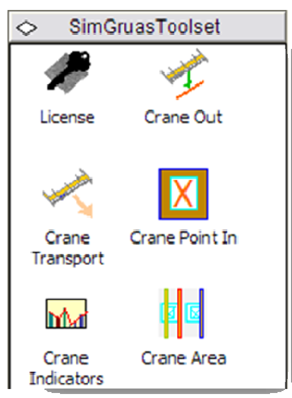

Figure 5: Template components bar

A template is a package of elements and blocks which can be used in any Arena® model; these elements and blocks are predesigned to represent an operation or physical object in particular. In the case of SimGruas, each of the blocks and elements of the template represent the operations and objects we will need to model up to five overhead cranes traveling along a single track.

The user simply gets the objects from the template bar, and builds the multiple overhead crane system as objects and blocks inside a new or existing ARENA model.

The template is understandable and simple to use having the following components:

- **Crane Area:** represents the single track along which the cranes will travel, defining zones for every crane
- **Crane Point In:** represent the point at which a crane can load objects
- **Crane Transport:** defines a loaded movement from origin to destination point
- **Crane Out:** lets the crane unload and leaves the template to the model
- **Crane Indicators:** shows different indicators of the cranes performance.

2.1 Logic and Algorithm

The logic behind SimGruas is based on a heuristic approach to the problem presented, where every interaction among cranes are solved.

The definition of the template blocks must follow these assumptions:

- The track is divided in "zones"; a crane must seize all required zones to move from an initial point to final point
- When assigning the zones for the cranes (each crane will be able to move throughout a subset of zones from the track), a particular zone of the track may be assigned to at most two contiguous cranes.
- For every subset of zones there must be a zone where the crane can be placed without interfering with any other crane, which is to say that that zone can only be attended by that crane.
- Every crane movement must be held inside its subset of zones, defined in the *Crane Area* block
- The logic analyzes the interaction of cranes according to their current states, and then proceeds to perform a proposed solution

Each conflict situation is detected before any crane's movement, and there is a typical action defined for each situation. Some conflict situations require assigning priorities to the cranes; this may be solved by manually assigning a priority number for each crane, or by a random priority assigned automatically.

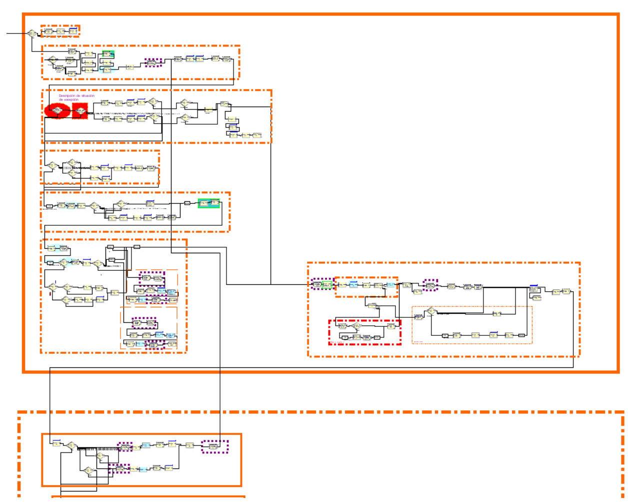

Figure 6: Part of the template's logic for a single crane movement

2.1.1 Algorithm Implementation

As an idle crane is required by a ¨Crane Transport¨ step, the crane's state is analyzed by the logic. If the crane appears to be ready to move, a set of objects directly related to "zones" availability will be seized(by the execution of two different loops), from the current crane location to its assigned loading point. At this moment, a second evaluation of states is performed, and if the crane is ready to move, the second loop is performed from loading point to unloading point. Seized zones are released as cranes advances.

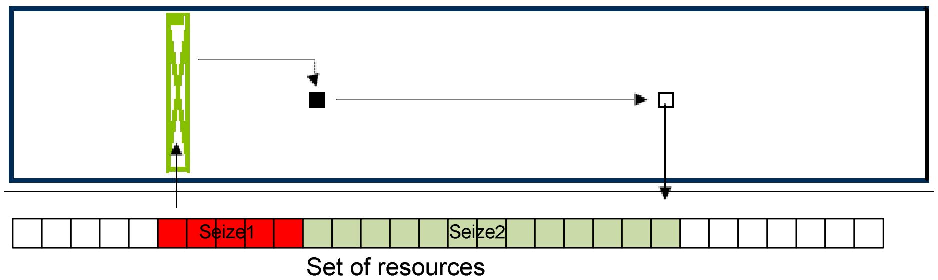

Figure 7: zone seizing logic

2.2 Reusability

By a simple combination of *Crane Point In*, *Crane Transport* and *Crane Out* blocks, where you just have to describe the coordinates of the origin and destination of the movement, and the crane that will perform it, the model will run, and you can just simply forget about the control logic of every possible interaction between the cranes, since they are already solved by the template itself.

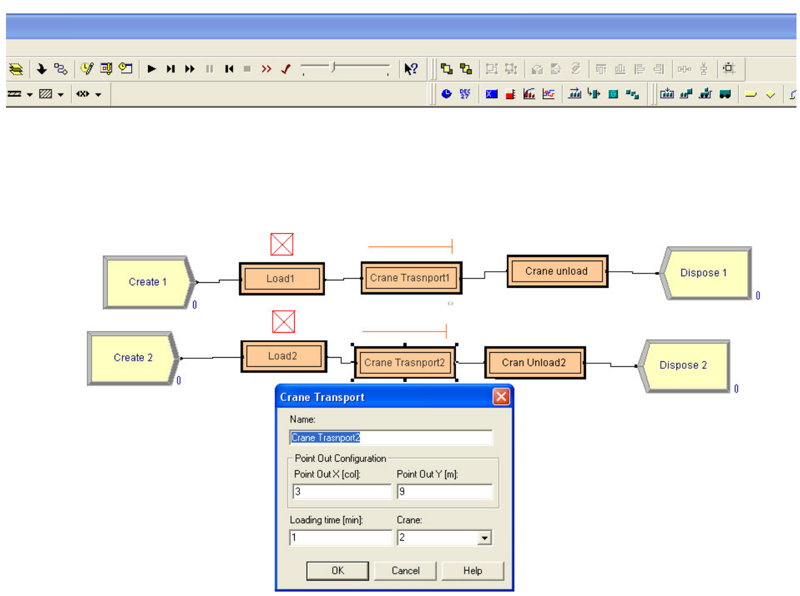

Figure 8: User's view. Logic for the movement of two cranes

Another characteristic of the template is that you can add it to existing models without any problems at all –just taking care on the names of the blocks and variables that will be created–, and you will be able to model the overhead cranes interacting with a greater system.

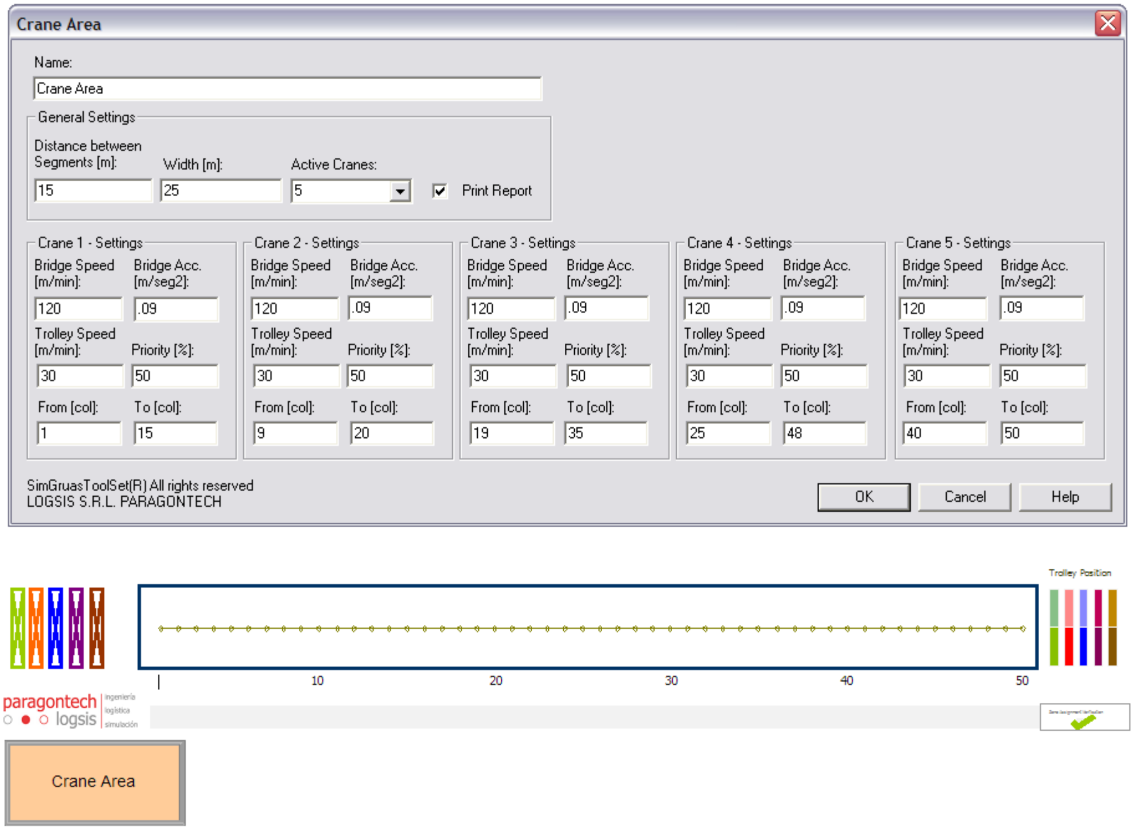

Figure 9: Template user view in model

3 APPLICATION

SimGruas Toolset®, was developed as a template by **Paragontech LOGSIS SRL** for commercial use in the popular simulation software Arena®, from Rockwell Simulation™. Demo videos are available at `<http://www.logsis.com.ar>`

Two **major steel makers** in Argentina are currently using SimGruas® as a decision tool for evaluating important plant investments. In most cases, the simulation has avoided the generation of bottlenecks in critical processes. SimGruas® is being applied as a part of new simulation models, as well as being added to existing simulation models of existing plants.

In demonstrations of real cases, line supervisors and operators have observed that models created with SimGruas showed a realistic behavior. When validating models using historical data, simulation results showed good fit in cranes occupancies and interferences.

The application of SimGruas allowed the detection of potential problems in plants design, avoiding significant production loses at the stage of plant design. Users have modified entire projected layouts, obtaining designs with expected plant performance increased in more than 10%.

3.1 Example

In order to simplify the understanding of SimGruas, a simple problem is presented:

- Products enter from a production line at one end of the building by pallet (one pallet every 4 minutes)
- Loading time: 1 min. Unloading time: 1 min.
- The warehouse area is located at the center of the building
- Products may be stored randomly. Waiting time is 24 to 48 hs.
- Products are dispatched at the other end of the building. They may be picked randomly.

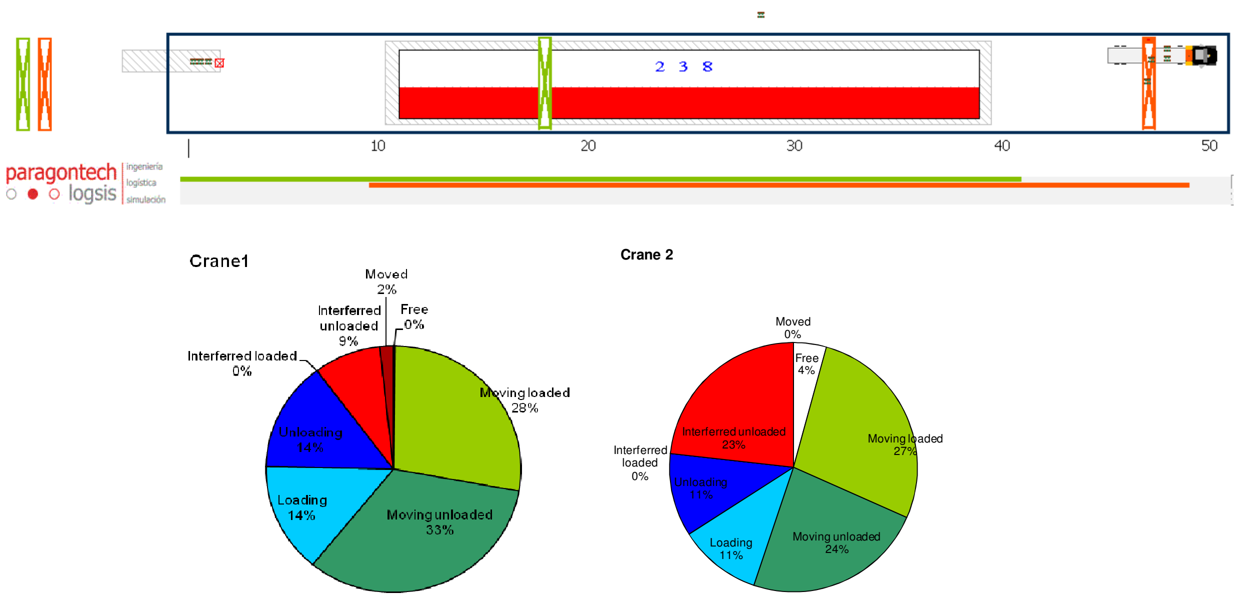

Figure 10: Example and results

4 ASSUMPTIONS AND LIMITATIONS

Real efficiency in this kind of problems may be higher; however we assume that a real operator will have a real efficiency lower than 100% in his decisions. The model considers interactions by pairs, according to limited information of one or two steps ahead. A very well coordinated movement of a group of cranes from a control center is not supported. The model is intended to solve object movements in single trips with no precise positioning; complex crane movements, like positioning objects, are not supported.

5 CONCLUSIONS

A reusable template is a powerful tool for a fast overhead crane simulation, dramatically increasing productivity of simulation developers. For instance, a simulation project involving interferences among overhead cranes was reduced from 3 months (normally required time) to 2 weeks using the template.

As overhead cranes interferences may represent significant production losses and very important economic losses, its simulation is a very profitable investment. The usage of a reusable template allows to develop simulation models in very short periods of time bringing reliable results.

ACKNOWLEDGMENTS

To P. Racca , P. Picasso and P. Bernard , for their valuable contributions during their internships, with programming techniques and algorithms.

REFFERENCES

Ali, M.S.A.D., N.R. Babu, and K. Varghese. 2005. Collision free path planning of cooperative crane manipulators genetic algorithm. *In Journal of Computing in Civil Engineering ASCE,* 19(2):182-193.

Canonaco, P., P. Legato, R.M. Mazza, and R. Musmanno. 2008. A queuing network model for the management of berth crane operations. *Computers and Operations Research* 35:2432–2446.

Daganzo, C. F. 1989. The crane scheduling problem. *Transport Research, Part B*. 23(3): 159–175.

Gould, Les. 1994. *Modern Materials Handling* 49(10):48-49.

Guo, X., and S.Y. Huang. 2008. Performing A* search for yard crane dispatching in container terminals. *In Proceedings of the 20th IEEE International Conference on Tools with A.I.* 1:263-267.

Jung, S.H., and K.H. Kim. 2006. Loading scheduling for multiple quay cranes in port container terminals. *Journal of Intelligent Manufacturing* 17:479-492.

Kelton, W., R. Sadowski and D. Sturrock. 2007. *Simulation With ARENA*. McGraw Hill

Kim, K.H., and K.Y. Kim. 1999. An optimal routing algorithm for a transfer crane in port container terminals. *Transportation Science* 33 (1): 17–33.

Kim, K. H., and Y. M. Park. 2004. A crane scheduling method for port container terminals. *European Journal of Operations Research* 156(3): 752–768

Linn, R.J. and C.Q. Zhang. 2003. A heuristic for dynamic yard crane deployment in a container terminal. *IIE Transactions* 35(2):161-174.

Ng, W.C. 2005. Crane scheduling in container yards with intercrane interference. *European Journal of Operational Research* 164: 64–78.

Ng, W.C., and K.L. Mak. 2005. Yard crane scheduling in port container terminals. *Applied Mathematical Modelling* 29: 263-275.

Pedgen, C. D., R. E. Shannon and R. P. Sadowski. 1995. *Introduction to Simulation Using SIMAN.* McGraw Hill.

Pinedo, Michael. 1995. *Scheduling: Theory, algorithms, and systems*. Englewood Cliffs, New Jersey: Prentice-Hall, Incorporated.

Reddy, H.R., and K. Varghese. 2002. Automated path planning for mobile crane lifts. *In Computer-aided Civil and Infrastructure Engineering* 17: 439-448.

Sammarra, M., J.F. Cordeau, H. Laporte, and M. F. Monaco. 2007. A tabu search heuristic for the quay crane scheduling problem. *Journal of Scheduling* 10: 327– 336.

Sivakumar, PL., K. Varghese, and N.R. Babu. 2003. Automated path planning of cooperative crane lifts using heuristic search. *In Journal of Computing in Civil Engineering ASCE,* 17(3): 197-207.

AUTHOR BIOGRAPHIES

DANIEL P. PAZ is project manager in simulation applied to evaluation of strategic investments, with a degree in Industrial Engineering from the Universidad Nacional de Rosario (Argentina) and an MBA at the Universidad de Barcelona (Spain). Founder of Paragontech LOGSIS, leader company in simulation applied to manufacturing and logistics in Argentina. He has participated in simulation projects for leading companies in Argentina, Brazil, Venezuela, and Peru. He can be contacted by email at: <dpaz@logsis.com.ar>.

LUIZ AUGUSTO G. FRANZESE is a simulation consultant with a degree in Production Engineering and an MSc. in Logistics, who has completed 250 successful projects with simulation. He founded PARAGON Tecnologia in 1992, the pioneer and leading consulting company in simulation in South America, and is partner of Paragontech LOGSIS. He has trained more than 1,200 professionals in simulation. He can be contacted by email at: <augusto@paragon.com.br>.