SUPPLY CHAIN SIMULATION METHODS ANALYSIS: AN APPLICATION TO THE BEER GAME

Daniel Guimarans[(a)], Julija Petuhova[(b)], Yuri Merkuryev[(c)], Juan José Ramos[(d)]

[(a), (d)] Telecommunication and System Engineering Dept., Universitat Autònoma de Barcelona, Spain
[(b), (c)] Department of Modelling and Simulation, Riga Technical University, Latvia

[(a)]daniel.guimarans@uab.cat, [(b)]julija@itl.rtu.lv, [(c)]merkur@itl.rtu.lv, [(d)]juanjose.ramos@uab.cat

ABSTRACT

The Beer Game has a typical supply chain structure that permits exploring a variety of supply management concepts. Many modelling methods have been used for supply chain analysis and so they can be applied to the Beer Game specific case study. Among them, discrete-event systems simulation has deserved special attention due to its suitability for modelling dynamic systems with a high degree of detailed elaboration and stochastic factors. For this reason, several discrete-event simulation oriented models have been elaborated to tackle the Beer Game and, by extension, multi-echelon supply chains. In the present paper, four of these models are described. Some of their applicability characteristics are also outlined, so a further discussion of their suitability according to simulation purposes can be done. Conclusions extracted from this analysis are presented in this work, aiming to help on choosing the most suitable model according to end user's preferences and purposes.

Keywords: supply chain simulation, Beer Game, Coloured Petri Nets, constraint programming, models analysis, uncertain environment.

1. INTRODUCTION

The Beer Game has a typical supply chain structure when it is represented as a serially connected inventory management systems chain. The Beer Game application enables to explore a variety of simple and advanced supply management concepts, taking into consideration environment uncertainty.

There are a variety of methods which address modelling and analyzing the supply chain. Within them, simulation has become an important tool for analysis and improvement of an entire supply chain operation. Different modelling methods that can be used for supply chain analysis may be classified as follows:

1. Analytical modelling: algebraic methods, automatic control theory, Petri-Nets, queuing theory, Markov chains, etc.
2. Algorithmic modelling:
   (a) Continuous Systems Simulation: differential equations, difference equations, etc.
   (b) Discrete-Event Systems Simulation: event or process oriented simulation, etc.

Analytical models have their place at a tactical level in the design of supply chains. Analytical techniques are able to solve batch sizing and job sequencing problems, yet fail to throw much light on the dynamic behaviour of the supply chain as a whole (Riddals, Bennett and Tipi 2000). Analytical techniques are useful in providing solutions to local tactical problems. Nevertheless, the impact of these solutions on the global behaviour of the whole supply chain can only be assessed using dynamic simulation. In addition, the computational burden associated to such techniques is to be considered an important drawback.

Ishii, Takahashi and Muramatsu (1988) developed a deterministic model for determining the base stock levels and lead times associated with the lowest cost solution for a supply chain. The stock levels and lead times are determined in such a way as to prevent stockout and to minimize the amount of obsolete inventory at each stock point. In this model, they need to decide upon two linearly varying demand rates in order to carry out the computation.

Williams (1981) presents seven heuristic algorithms for scheduling production and distribution operations in an assembly supply chain. The objective of each heuristic is to determine a minimum-cost production and/or product distribution schedule that satisfies the final product demand. However, they fail to illuminate the dynamics of the system.

Cohen and Lee (1989) present a deterministic, mixed integer, non-linear mathematical programming cost-based model. They use an economic order quantity technique to maximize the total after-tax profit for the manufacturing facilities and distribution centres, but dynamics are not included.

Modelling supply chains using continuous system simulation holds great appeal for control theorists. This is because many of the influential characteristics of the problem can be succinctly expressed in a differential equation form (Riddals, Bennett and Tipi 2000). Continuous systems simulation has the advantage of being a conduit into the frequency domain, which offers a framework particularly suited for the study of systems in which oscillations are a salient attribute, e.g. analysis of factors having impact on a seasonal, or other demand fluctuations, amplification as they are passed along the chain. Since differential equation produce “smooth” outputs, they are not suited to modelling of all supply chains. The system must be considered at an aggregated level, in which individual entities in the system (products) are not considered. Rather, they are aggregated into levels and flow rates. Consequently, these methods are not suited for production processes in which each individual entity has an impact on the fundamental state of the system. For the same reasons, continuous systems simulation cannot solve lot sizing and job sequencing problems.

Forrester (1961) developed what he called Industrial Dynamics, which he later extended and renamed System Dynamics. He developed a nonlinear model of a supply chain using first-order differential equations. He analysed the demand fluctuation amplification as it proceed down the multi-echelon supply chain, using continuous time models. However, no sensitivity and cost-based analysis have been carried out on these models, which are solely concerned with the dynamics. Many discrete-event simulation packages available today provide a more advanced simulation capability. Armbuster, Marthaler and Ringhofer (2002) modelled high volume production flows using nonlinear hyperbolic partial differential equations, with Little’s law explicitly built into the formulation. By using the developed models, they are able to analyse multiple products, dispatch polices and control actions.

The emergence of discrete-event systems simulation (DES) was engendered by the deficiencies of differential equation approaches to the solution of even simple man-made problems (Riddals, Bennett and Tipi 2000). Consider, as an example, governing the behaviour of a series of queues at a supermarket. The modelling of phenomena such as queue swapping (when customers jump to shorter queues) or variable service speed (faster when there are more customers) would make impossible the application of differential equations, as well as any other theoretical approach. However, such phenomena can easily be incorporated into a DES model.

DES has the following two characteristics: (1) it represents individual events, e.g., the arrival of an individual customer order; (2) it incorporates uncertainties, e.g., customer orders arrive at random points in time, machines break down at random points of time, etc. (Kleijnen 2005). Most systems dynamics models are non-stochastic, but their behaviour often becomes incomprehensible due to nonlinear feedback loops. Most econometric models are also based on the deterministic, nonlinear differential equations. DES provides more accurate simulation capabilities against above described techniques and so it has been considered an important method in supply chain modelling. Banks, Buckley and Jain (2002) described a lot of DES based studies, like commercial packages developed by IBM for supply chain management simulation for both operational and strategic planning levels. For more details on DES one could refer to the many textbooks, e.g., Law and Kelton (2000), Banks, Carson and Nelson (2004), Ho and Cao (1991), etc.

Different described methods of mathematical modelling are suitable for different problems solving and all have their place in the design and management of supply chains. However, the analysis of advantages and disadvantages of the proposed methods specifies DES as the more appropriate method for modelling dynamic systems with a high degree of detailed elaboration and stochastic factors, such as supply chains. For this reason, only models included in the DES category have been considered in this paper.

The remainder of this paper is structured as follows: next section presents a rough description of the Beer Game and introduces the studied models. Section 3 presents a discussion about models suitability according to end user purposes. Finally, some conclusions are outlined in the last section.

2. THE BEER GAME

The Beer Game is a role-playing simulation developed at Massachusetts Institute of Technology in the 1960’s to clarify supply chains’ behaviour (Jarmain 1963). The Beer Game model considers a simplified beer supply chain, consisting of a single retailer, a single wholesaler which supplies the retailer, a single distributor which supplies the wholesaler, and a single factory with unlimited raw materials which makes (brews) the beer and supplies the distributor (Figure 1).

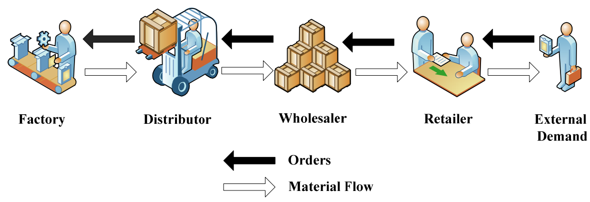

Figure 1: Beer Game’s Supply Chain Structure

Each component has unlimited storage capacity and the manufacturer has also unlimited raw materials. There are a fixed supply lead time and order delay time between each participant.

Every week, each component in the supply chain tries to meet the demand of the downstream participant. Any orders which cannot be fulfilled are recorded as backorders. These unmet orders are to be satisfied as soon as possible, since no orders can be ignored. At each period, each member orders some amount from its upstream supplier. It takes one week for this order to arrive at the supplier. Once the order arrives, the supplier attempts to fill it with its available inventory and it takes an additional delay time, commonly set to two weeks, before goods arrive to the customer who placed the order. Usually, each supply chain component has no knowledge of the external demand or the orders and inventory of the other members. Nevertheless, in some cases, all components may share information in order to optimise supply chain's behaviour (Simchi-Levi, Kaminsky, and Simch-Levi 2003).

At each period, each component owns the inventory at that facility and goods in transit to the downstream participant. Each location is charged $1 per item that it owns as inventory holding cost. In addition, any backordered item is charged $2 per week. The external demand is uncertain and the goal of the retailer, wholesaler, distributor, and factory, is to minimise total cost, either individually, or for the whole system.

2.1. Computerised Beer Game

The Computerised Beer Game is a Windows based program written in C++, developed by Kaminsky and Simchi-Levi (1998), providing an interactive tool for teaching some supply chain behaviour characteristics.

The Computerised Beer Game follows the original rules of the Beer Game, with few exceptions aimed to enhance teaching possibilities. The end user can only take one role, usually the distributor, while the computer manages all remaining components according to the chosen policies (Figure 2). These characteristic allows the player focusing on single managerial decisions rather than understanding the whole chain behaviour, for which few information is known. Furthermore, demand in the Computerised Beer Game may be chosen to be either completely deterministic, as in the original game description, or random, following a statistical distribution.

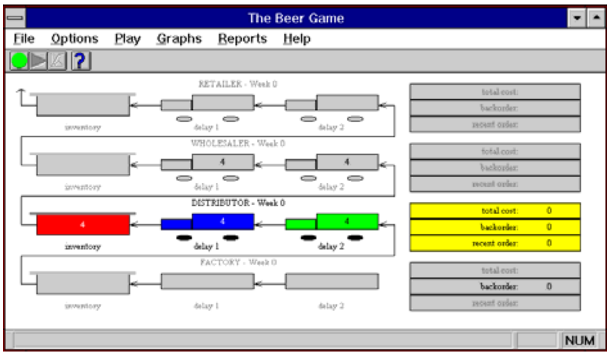

Figure 2: User Interface of the Computerised Beer Game

However, main changes with respect to the original beer game are the options to play with global information, centralized information and/or shortened lead time. When playing the global information scenario, all information is always known, including customer demand and inventories. In the centralized information version, the player can only take the role of the manufacturer. Because the system is centralized, only this component can place orders, while goods are moved downstream as quickly as possible. As in the previous option, all information is always available. This option permits to compare centralized and decentralized policies if costs are correctly adjusted, since no backorders are allowed in the decentralized scenario. Finally, the short lead time version allows reducing the delivery delay from two weeks to one.

The Computerised Beer Game is mainly aimed to education and training on supply chain management. Although results obtained are equivalent to other models and the interactive role may be switched off, so all participants are controlled by the computer, its graphical interface significantly slows down its performance. Therefore, it might not be a good option if the simulation is aimed to analytical purposes, for which other models may provide the same results faster.

2.2. Coloured Petri Nets Model

Coloured Petri Nets (CPN) formalism has proven to be a successful tool for modelling the characteristics for any type of discrete event oriented system. CPN shows several advantages such as the conciseness of embodying the static structure and the dynamics, the availability of the mathematical analysis techniques as well as its graphical nature (Jensen 1997).

The Beer Game has been modelled using timed Hierarchical Coloured Petri Nets (Jensen 1997) following the general scheme presented in Panic, Vujosevic and Makajic-Nikolic (2006). The top level of the model is the whole supply chain represented in the Beer Game, including a customer and the four described agents: retailer, wholesaler, distributor and manufacturer (Figure 5).

The customer is presented by a place, whose initial marking specifies its demand in time. Retailer, wholesaler and distributor are modeled by a *Supplier* sub-model (Figure 3). Finally, the manufacturer is represented by a *Manufacturer* sub-model (Figure 4). Furthermore, this hierarchical model allows including additional suppliers between the customer and the manufacturer, each of them modelled by an instance of the *Supplier* sub-model. This is possible due to in the Beer Game it is assumed all participants make decisions according to the same rules. Thus, all agents included in the supply chain are equal from a modelling perspective.

Although the CPN model may be extended by adding additional parameters, only orders, backorders and deliveries have been considered. These variables are enough to trace system's behaviour in order to show and analyze the bullwhip effect.

An instance of the sub-model *Supplier* has been used to represent the retailer, wholesaler and distributor. According to the order received in place `Demand (downstream)`, the current inventory at place `Stock` and backordered items in place `Backorders`, a supplier makes its own order. This process is modelled through the instantaneous transition `Place order` and the function `order(b,k,n)`, where different policies may be used. These policies may be implemented in a deterministic way or kept open to allow interaction with end users, according to simulation goals. Transition `Fulfil demand` is used to model requested amount's delivery. If the inventory stores enough goods, the complete demand, including last received order and backordered items, is satisfied and the remaining (function `rest(n,b)`) is kept in the stock. Otherwise, all available goods are delivered and the difference is backordered (`rest(b,n)`). The duration of transition `Fulfil demand` is @+2, since the Beer Game rules establish that deliveries last 2 weeks. The associated guard function ensures this transition is only fired when an order exists, or there are backorders to satisfy, and there are goods in stock.

The *Manufacturer* sub-model is similar to the *Supplier* one. In fact, the manufacturer acts as other suppliers, but deciding what amount to produce in the following period instead of placing an order to its upstream agent. Again, this decision is made according to the current demand, backorders and inventory. The associated transition is `Manufacture`, which has associated a duration of 2 weeks (@+2) as MIT Beer Game rules state.

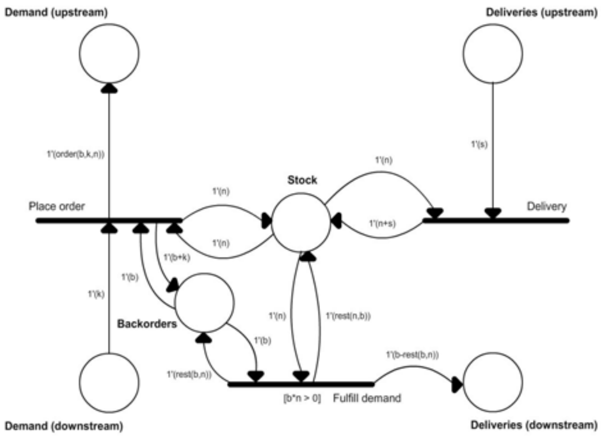

Figure 3: Sub-Model Supplier

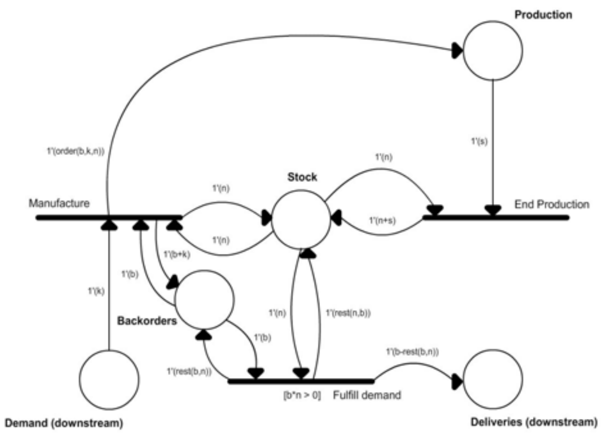

Figure 4: Sub-Model Manufacturer

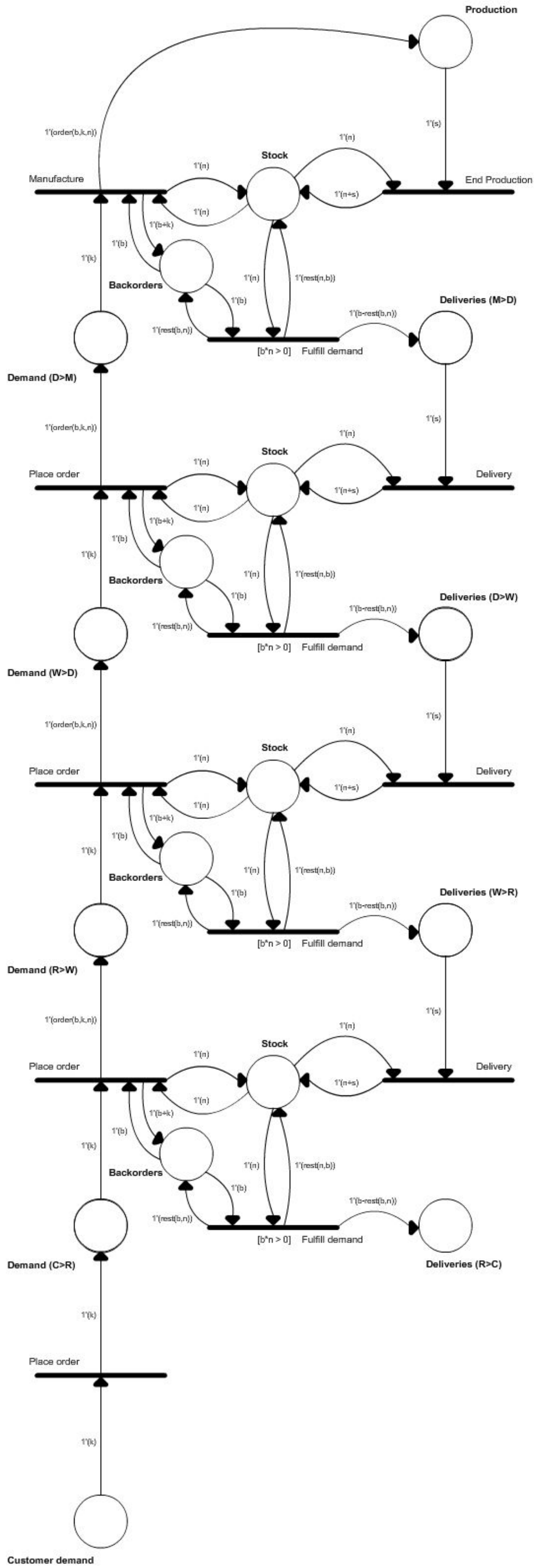

Figure 5: Complete Beer Game CPN Model

This model may be implemented in several platforms supporting CPN simulation, such as CPN Tools (2010), or using any programming language (Guasch, Piera, Casanovas and Figueras 2002). According to implementation details, the model is likely to be used with different purposes. As an example, all policies may be implemented so all participants behave in a deterministic way. On the other hand, function `order` may be left in blank for one, some, or all participants, allowing end users interaction. Thus, an interactive CPN model is implemented, especially suitable for educational and training purposes, comparable to the original Beer Game. Taking into account extra variables may be included in the model, this CPN model is more likely to be extended according to training preferences and dynamics to be studied.

2.3. Constraint Programming Model

Constraint Programming (CP) is a powerful paradigm for representing and solving a wide range of combinatorial problems. Problems are expressed in terms of three entities: variables, their corresponding domains and constraints relating them. The problems can then be solved using complete techniques such as depth-first search for satisfaction and branch and bound for optimization, or even tailored search methods for specific problems. Rossi et al. (2006) presents a complete overview of CP modelling techniques, algorithms, tools and applications.

The CP model may be seen as a specific implementation of the CPN model described in the previous section. Constraints among variables are defined as a set of rules, relating each component's variables with its upstream/downstream participant and their values at each period.

Four sets of variables have been defined, one for each component $i=[1..4]$ (1: retailer, 2: wholesaler, 3: distributor, 4: manufacturer) and the final customer being $i=0$. Periods are denoted by the variable $t=[1..t_{max}]$, where t_{max} is usually set to 50 weeks. Thus, for component i at period t, variables defined are: $\mathtt{INV}_{it}$ is the current inventory, $\mathtt{DEL1}_{it}$ and $\mathtt{DEL2}_{it}$ are the goods in transit, $\mathtt{BO}_{it}$ is the number of backordered items, $\mathtt{DEM}_{it}$ is the demand to be satisfied, $\mathtt{ORD}_{it}$ is the order placed by the component, $\mathtt{SHIP}_{it}$ is the amount shipped by the component in the current period and $\mathtt{COST}_{it}$ is the associated cost to this turn. Variables are related according to the following rules:

$$\mathtt{DEM}_{1t}=\mathtt{ORD}_{0\ t} \quad (1)$$
$$\mathtt{DEM}_{it}=\mathtt{ORD}_{i-1\ t-1} \quad (2)$$

$$\mathtt{INV}_{i\ t-1} + \mathtt{DEL1}_{i\ t-1} \geq \mathtt{DEM}_{it} + \mathtt{BO}_{i\ t-1} \rightarrow \quad (3)$$
$$\mathtt{INV}_{it}=\mathtt{INV}_{i\ t-1}+\mathtt{DEL1}_{i\ t-1}-\mathtt{SHIP}_{it}$$
$$\mathtt{SHIP}_{it}=\mathtt{DEM}_{it}+\mathtt{BO}_{i\ t-1}$$
$$\mathtt{BO}_{it}=0$$

$$\mathtt{INV}_{i\ t-1} + \mathtt{DEL1}_{i\ t-1} < \mathtt{DEM}_{it} + \mathtt{BO}_{i\ t-1} \rightarrow \quad (4)$$
$$\mathtt{INV}_{it}=0$$
$$\mathtt{SHIP}_{it}=\mathtt{INV}_{i\ t-1}+\mathtt{DEL1}_{i\ t-1}$$
$$\mathtt{BO}_{it}=\mathtt{BO}_{i\ t-1}+\mathtt{DEM}_{it}-\mathtt{SHIP}_{it}$$

$$\mathtt{DEL1}_{it}=\mathtt{DEL2}_{i\ t-1} \quad (5)$$
$$\mathtt{DEL2}_{it}=\mathtt{SHIP}_{i+1\ t} \quad (6)$$

$$\mathtt{COST}_{it}=1*(\mathtt{INV}_{it}+\mathtt{DEL1}_{it}+\mathtt{DEL2}_{it})+2*\mathtt{BO}_{it} \quad (7)$$

The total cost for component i for the whole simulation period is then calculated trivially:

$$TCOST_i = \sum_{t=1}^{t_{max}} COST_{it} \quad (8)$$

The value of $\mathtt{ORD}_{it}$ in (2) is determined according to the chosen policy. Several policies may be implemented, usually depending on inventory and demand parameters.

Rule (3) is only applied to update parameters when there is enough stock to fulfil the current demand. On the other hand, rule (4) is used whenever the component i is not able to meet demand requirements. Constraints (5) and (6) update transportation variables at each period.

Departing from a given initial state and a list of customer's demand along periods, CP propagation rules determine immediately all remaining variables. Therefore, the model is able to provide instantaneously results corresponding to a complete simulation. Moreover, since propagation rules are not unidirectional, the model may provide a mechanism to infer other participants' policies, inventory bounds and even final customer's demand. With this goal, initial information concerning the evolution of own variables should be provided, instead of final customer's demand. So, the model may reconstruct demand patterns from other participants, even for the final customer, and provide a good mechanism for analysis and a first step to get a reliable forecasting tool.

The CP model, implemented using the CP platform ECLiPSe (Apt and Wallace 2007), is instantaneous for common simulation periods. Since propagation is very fast, the model is especially suitable for analysis purposes. Furthermore, it is likely to be parallelized, so different scenarios may be defined and run simultaneously to get different data. Nevertheless, the model provides a low level of interaction, so it might not be a good option for decision making training.

2.4. Arena

Simulation techniques are used when analytical solving is impossible. Most of all analytical approaches do not succeed analysing complex, dynamic systems like supply chains. For this reason, the Beer Game has been modelled using DES software Arena by Rockwell Automation (2010). Arena software is effective when analysing complex, medium to large-scale projects, involving highly sensitive changes related to supply chain, manufacturing, processes, logistics, distribution, warehousing and service systems. Arena proposes an event-oriented modelling approach and consists of "libraries" of modelling objects that make it significantly easier and faster to develop models. Arena exploits two Windows technologies that are designed to enhance the integration of desktop applications. The first, ActiveX Automation allows applications to control each other and themselves via a programming interface. The second technology addresses the programming interface issues to a Visual Basic programming environment. These two technologies work together to allow Arena integrating with other programs that support ActiveX Automation, e.g., Excel, AutoCAD, or Visio.

The Beer Game supply chain and its inventory control systems have been implemented combining Arena software and Visual Basic for Applications (VBA). The model logic is represented comprehensibly in the Arena flowchart-style environment, while the more complex calculation algorithms are programmed in VBA. The traditional four-stage Beer Game structure has been modelled in combination with two information sharing strategies (centralised and decentralised) and two inventory control policies (s-S and Stock-To-Demand).

In accordance with specific features of the Beer Game, the proper event processing schedule implementation in Arena environment is shown in Figure 6. If several events are scheduled to occur at a certain supply chain stage at the same simulation time, there is a fixed order in which the events should be processed:

1. order or backorder arrival from upstream stage (stock replenishment);
2. fulfilling of backorders (only if an order has arrived);
3. new demand fulfilling.

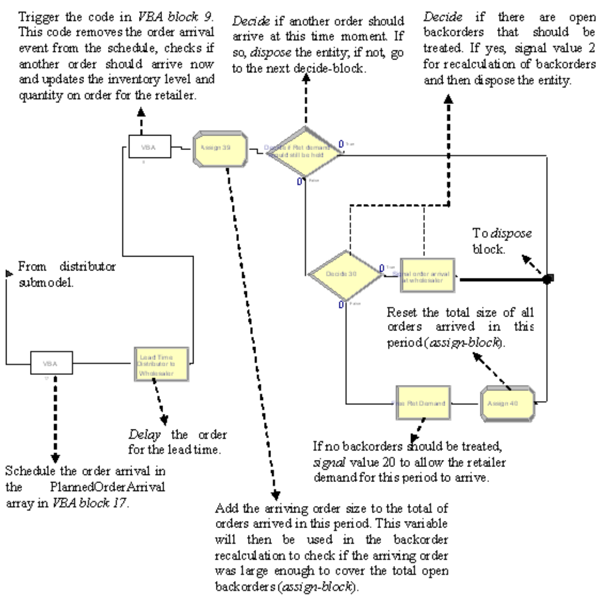

Figure 6: Submodel of Order Shipment to Wholesaler

As Arena's simulation engine do not always process the events in this order (Kelton and Sadowski 2002), a procedure has been developed to guarantee that events are processed in the mentioned sequence. `Wait` and `Signal` blocks form implementation's basis of this procedure in Arena.

Various experiments may be performed with the created model and achieved results may be analysed by the Output Analyzer. This is a component of Arena that provides an easy-to-use interface, simplifying data analysis and allowing viewing and analysing simulation data quickly and easily.

The Arena model is suitable for research purposes, since it incorporates both dynamic and stochastic nature of the supply chain operations and the simulation execution speed without animation is quite fast. The analysis of system's behaviour under specified conditions may be easily performed and allows foreseeing thousands of situations which could result from supply chain operations. Each of the supply chain stages is modelled as a separate module and it is possible to change both supply chain structure (e.g., add or remove a stage) and supply chain management concepts (e.g., centralized or decentralized information sharing strategy).

However the model is not suited for training and education due to low level of interaction with the end user. To modify the model parameters, end user should have previous experience on working with Arena software.

2.5. Excel

An alternative to DES specific software is developing a supply chain simulator implemented in a general-purpose high-level programming language, e.g., C++, Java or VBA. Programming languages are mainly used for simulation in order to avoid additional expenses of commercial software purchasing and maintenance.

The user interface of the Beer Game supply chain model is developed by means of Microsoft Excel spreadsheets (Figure 7), which is widely applied and easy accessible software. The programming logic is implemented using VBA (Ternovoy 2004).

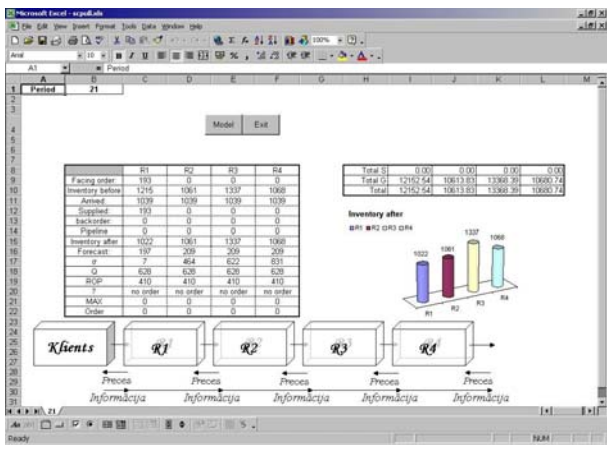

Figure 7: Supply Cain Model User Interface

Supply chain's control variables' values, as well as initial data, are defined by the user using a MS Excel interface, and then the simulator is run for the specified number of periods by means of a VBA procedure (Figure 8).

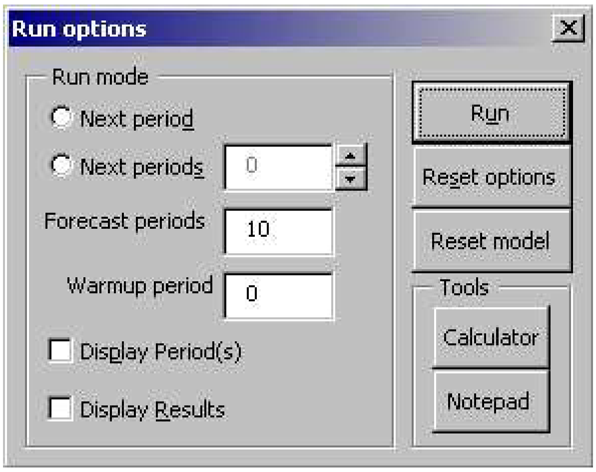

Figure 8: Simulation Parameters

The simulation procedure `Next_n` runs the model in accordance with defined user settings. First, constants and variables are defined and information is read from the data files. Then, the simulation of the processes scheduled is performed and repeated four times for all supply chain's stages, i.e. once for each component. Events management is performed in accordance with the Beer Game structure. Eventually, simulation results appear in separate spreadsheets showing different types of charts, histograms and tables.

The functional possibilities of the Excel simulation model are quite similar to the Arena model, but since the special simulation software is not needed and experiments are easily configured, it can be used for educational and training purposes. However, the speed of executing a simulation run in Excel is quite slow and the number of simulated periods is restricted, so research tasks are difficult by using this model.

3. MODELS SUITABILITY

Although all described models yield the same results, choosing one or another relies on simulation's goals. Each model's characteristics make it more suitable according to pursued purposes: research or training/education. However, all models may be used to simulate the Beer Game regardless of their characteristics, being only a matter of efficiency which one is best suited to a specific simulation goal.

Table 1: Models' Characteristics Summary

Model	Interactive	Scalable	Speed	Infer Policies
Comput.	X			
CPN	X	X		
CP		X	X	X
Arena		X	X	
Excel	X	X		

Table 1 summarizes characteristics for all described models in the previous section. Even though different parameters might be defined, only those relevant for purposes considered in this paper have been included, i.e. interactivity, scalability, execution speed and the capability of inferring demand patterns from a data set. Interactivity is a characteristic of Computerised, CPN and Excel models. In all of them, different parameters can be chosen or modified on runtime, or the model is likely to be modified easily to include some level of interaction with the end user. On the other hand, the CP model is completely deterministic and results are only derived from the initial data and set-up. Arena model requires some particular skills in order to permit some interaction with the end user. Models' modularity determines their capability of being scaled to represent larger systems. In this sense, Arena, Excel, CPN and CP models are clearly modular, since different components are defined separately and may be concatenated with little changes. The computerised Beer Game is proprietary software and so the system it represents cannot be modified. According to the simulation speed, CP and Arena models are clearly faster than other models. This characteristic makes them especially efficient when running a set of simulations with different parameters, as demanded for analysis purposes. Finally, only the CP model allows performing simulations with different initial data than the final customer demand and participants' policies. This characteristic provides a mechanism especially important from a forecasting perspective. As mentioned in the CP model description, it also permits reconstructing demand patterns and inferring other participants' policies and inventory bounds. Therefore, this characteristic might be an interesting tool for analysis, either for research or management.

Table 2: Models' Suitability According to Simulation Purposes

Model	Training / Education	Research	
		Analysis	Strategies develop.
Computerised	X		
CPN	X	X	
CP		X	X
Arena		X	
Excel	X		

All parameters related in Table 1 characterize models' performance and adaptability, and help on choosing one or another depending on simulation purposes. For example, if the simulation is training or education oriented, requested characteristics mainly include a high interactivity, in order to permit the end user making his own decisions and introducing them into the system. These decisions could be programmed in advance for all models, so not interactivity is allowed at all, but it would reduce the teaching or training experience to a single analysis after the simulation ends.

For this reason, interactive models among described ones, i.e. Computerised Beer Game, CPN and Excel, are considered to be best suited for management training or education. On the other hand, research tasks demand another kind of characteristics, especially related to execution speed. Usually, separate sets of simulations are run combining different demand patterns and policies, in order to analyse system's behaviour. With this goal in mind, it is more appropriate to choose a model which allows running a complete simulation in a low time, such as CP or Arena models. Although the execution time depends on the implementation platform and computer's characteristics, both models have demonstrated being fast enough for research tasks. Another desirable model characteristic is scalability, since it allows modifying system's dimensions with analytical purposes. In this sense, both CPN and CP models permit changing the number of components easily while ensuring a complete integration of all variables. Finally, the capability of the CP model to infer other components' policies, demand patterns and inventory bounds, combined with its low execution time, make it especially suitable for strategies and policies development, as well as a first step for developing forecasting tools. Table 2 presents which models best fulfil different simulation goals. Nevertheless, conclusions presented in Table 2 are to be considered more a guide than a rule, since all models may be used for all selected purposes according to end user preferences, as mentioned.

4. CONCLUSIONS

The Beer Game has a sequential supply chain structure that permits exploring many supply management concepts. In the present paper, four discrete-event simulation models representing the Beer Game have been described. Some of their characteristics have also been outlined and related to their suitability according to simulation purposes. This analysis may form a helpful basis for choosing the most suitable model according to end user's preferences and purposes.

The Computerised Beer Game is mainly aimed to education and supply chain management training. It allows a high level of interaction with the end user, although the interactive role may also be switched off aiming to increase the simulation speed. However, its graphical interface significantly slows down model's performance, so it might not be the best option if the simulation has analysis purposes.

CPN provide a mechanism to specify a conceptual model, likely to be implemented either on CPN simulation platforms or by using high-level programming languages. Thus, it possesses a set of characteristics that permit using it for educational and training purposes, as well as for analysis and research tasks.

The CP model may be considered as an implementation of the proposed CPN model. Since constraint propagation is very fast, it provides almost instantaneous results for common simulation periods. This way, thousands of different scenarios could be simulated sequentially, or even in a parallelized environment, getting results in reasonable times. On the other hand, there is a lack of interactivity with the end user that makes it difficult to be used with real-time decision making training.

The Arena model incorporates advantages from using specific simulation software, such as the capability of including both dynamic and stochastic nature of supply chains operations. Since it is able to provide results in a quite reasonable time, this model is suitable to be used for research purposes. However, using Arena on developing and modifying supply chain models requires particular skills on working with simulation software environment.

Using Excel for developing a simulation model becomes an alternative to using Arena. Both models possibilities are similar, having the advantage that no special simulation software is needed. Thus, Excel model may also be used for educational and training purposes. However, its simulation speed is quite low compared to Arena, so research tasks are limited when using this model.

Described models' main purposes are clearly determined by parameters analysed in section 3, among others. As an example, for training simulations is highly desirable to run a model which allows interacting with the end user. Thus, he can check almost instantly consequences derived from his decisions. On the contrary, research tasks often require a high execution speed, so different scenarios may be simulated and analysed in low times. Scalability is another desirable characteristic, even though it is not critical due to Beer Game models simplicity. It should be remarked a characteristic that only the CP model possesses: inverse deduction and reconstruction of policies and inventory bounds. CP paradigm allows this characteristic, since constraints are not unidirectional, unlike rules included in other models.

Finally, it should be remarked that all four presented models are simulation oriented. Simulation models are the so-called input-output models, i.e., they yield the output of the system for a given input. Therefore, simulation models are "run" rather than "solved" (Hurrion 1986). For this reason, no one of presented models may be used for optimization purposes without significant changes. In any case, system may be studied and optimised, but depending entirely on the end user. Among models described in the present paper, only the CP model is suitable to be easily modified to include an objective function in order to be optimised. However, supply chains nature would induce to explore a huge search space, characteristic of NP problems. Tackling this kind of problem would require using complex algorithms, i.e. heuristics and metaheuristics, combined with simulation and other operational research techniques.

REFERENCES

Apt, K., and Wallace, M.G., 2007. *Constraint Programming using ECLiPSe*. Cambridge: Cambridge University Press.

Armbuster, D., Marthaler, D., and Ringhofer, C., 2002. Efficient simulations of supply chains. *Proceedings of the 2002 Winter Simulation Conference*, 1345-1348. December 8-11, San Diego (USA).

Banks, J., Buckley, S., and Jain, S., 2002. Panel session: opportunities for simulation in supply chain management. *Proceedings of the 2002 Winter Simulation Conference*, 1652-1658. December 8-11, San Diego (USA).

Banks, J., Carson, J.S., and Nelson, B.L., 2004. *Discrete-Event System Simulation*. New Jersey: Prentice Hall.

Bessiere, C., 2006. Constraint propagation. In: Rossi, F., van Beek, P. and Walsh, T., eds. *Handbook of Constraint Programming*. Amsterdam: Elsevier, 29-83.

Cohen, M.A., and Lee, H.L., 1989. Resource deployment analysis of global manufacturing and distribution networks. *Journal of Manufacturing and Operations Management*, 2: 81-104.

CPN Tools, 2010. *CPN Tools, Computer Tool for Coloured Petri Nets*. Available from: wiki.daimi.au.dk/cpntools.

Forrester, J.W., 1961. *Industrial Dynamics*. New York: MIT Press and Wiley.

Guasch, A., Piera, M.A., Casanovas, J. and Figueras, J. (eds.), 2002. Simulación de Sistemes Orientados a Eventos Discretos. In: *Modelado y Simulación. Aplicación a Procesos Logísticos de Fabricación y Servicios*. Barcelona: Edicions UPC.

Ho, Y.C., and Cao, X.R., 1991. *Perturbation Analysis of Discrete Event Dynamic*. Boston: Kluwer Academic Publisher.

Hurrion, R.D., 1986. *Simulation: Applications in manufacturing - International trends in manufacturing technology*. Bedford: Springer-Verlag.

Ishii, K., Takahashi, K., and Muramatsu, R., 1988. Integrated production, inventory and distribution systems. *International Journal of Production Research*, 26(3): 473-482.

Jarmain, W.E., 1963. *Problems in Industrial Dynamics*. Cambridge: MIT Press.

Jensen, K., 1997. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use*. Volumes 1, 2, 3. Berlin: Springer-Verlag.

Kaminsky, P., and Simchi-Levi, D., 1998. A new computerized beer game: a tool for teaching the value of integrated supply chain management. In: Hau Lee and Shu Ming Ng, eds. *Global Supply Chain and Technology Management*. Miami: The Production and Operations Management Society, 216-225.

Kelton, W.D., and Sadowski, R.P., 2002. *Simulation with Arena*. New York: McGraw-Hill.

Kleijnen, J.P.C., 2005. Supply chain simulation tools and techniques: a survey. *International Journal of Simulation & Process Modelling*, 1(1-2): 82-89.

Law, A.M., and Kelton, W.D., 2000. *Simulation modeling and analysis*. International Series in Industrial Engineering and Management Science. Singapore: McGraw-Hill.

Panic, B., Vujosevic, M., and Makajic-Nikolic, D., 2006. A Supply Chain Simulation Model Based on Hierarchical Coloured Petri Nets. *Proceedings of the International Scientific Days – Competitiveness in the EU – Challenge for the V4 Countries*, 1431-1436. May 17-18, Nitra (Slovakia).

Riddalls, C.E., Bennett, S., and Tipi, N.S., 2000. Modelling the dynamic of supply chain. *International Journal of System Science*, 31(8): 969-976.

Rockwell Automation, 2010. *Arena Simulation*. Available from: www.arenasimulation.com.

Rossi, F., van Beek, P., and Walsh, T. (eds.), 2006. *Handbook of Constraint Programming*. Amsterdam: Elsevier.

Simchi-Levi, D., Kaminsky, P., and Simchi-Levi, E., 2003. *Designing and Managing the Supply Chain: Concepts, Strategies and Case Studies*. New York: McGraw-Hill.

Ternovoy, V., 2004. *Simulation-Based Analysis of the Supply Chain Operation*. Master thesis (supervisor Petuhova, J.). Riga Technical University.

Williams, J.F., 1981. Heuristic techniques for simultaneous scheduling of production and distribution in multi-echelon structures: theory and empirical comparisons. *Management Science*, 27(3): 336-352.

ACKNOWLEDGMENTS

This work has been supported by the Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) of the Generalitat de Catalunya (project 2009 SGR 629).

AUTHORS BIOGRAPHY

Daniel Guimarans is a PhD Student and Assistant Teacher at the Telecommunication and Systems Engineering Department at the Universitat Autònoma de Barcelona. He is an active member of the Logisim Research Group. His research interests are constraint programming, simulation and hybridisation of different techniques to solve industrial combinatorial problems.

Julija Petuhova is Doctor of Engineering, lecturer of the Institute of Information Technology at Riga Technical University. Her research interests include simulation methodology of logistics systems, supply chain dynamics, practical applications of simulation modelling, training and education via simulation-based business games. She is a member of the Latvian Simulation Society and has a wide experience in performing research projects in the simulation area at national level.

Prof. Yuri Merkuryev is Habilitated Doctor of Engineering, Professor of the Institute of Information Technology at Riga Technical University, Head of the Department of Modelling and Simulation. His professional interests include methodologies and practical implementation of discrete-event simulation, supply chain modelling and management, and education in the areas of modelling, simulation and logistics management. He is Programme Director of the Master-level curriculum “Industrial Logistics Management” at Riga Technical University. Prof Merkuryev has a wide experience in performing research and educational projects in the simulation area, at both national and European levels, and regularly participates in organising international conferences in the simulation area. He has promoted 5 PhD theses and authors about 250 scientific publications, including 5 books and numerous journal and conference papers, as well as a text-book in Logistics Information Systems. He is also co-editor of a collection of simulation-based case studies in logistics, published in 2009 by Springer-Verlag. Prof. Merkuryev is a Board member of the European Council of the Society for Modeling and Simulation International, President of the Latvian Simulation Society, and Board member of the Latvian Transport Development and Education Association.

Juan José Ramos is an Associated Professor at the Telecommunication and Systems Engineering Department at the Universitat Autònoma de Barcelona. He is coordinator and an active member of the Logisim Research Group. His research interests are modelling, simulation and optimisation of logistics problems.