Manoj Kumar et al. International Journal of Recent Research Aspects ISSN: 2349-7688, Vol. 3, Issue 1, March 2016, pp. 63-68

Applying Multi-agent simulation in tracking the Bullwhip effect of supply chain system

Manoj Kumar[1], Dr. R. K. Khola[2]

[1]Reserach Scholar, Suresh Gyan Vihar University, Jaipur, Rajasthan, India
[2]Professor, Department of E-C Engineering; Suresh Gyan Vihar University, Jaipur, Rajasthan, India.

Abstract: **We have applied a fast, easily extensible, discrete-event multi-agent simulation toolkit in Java. This approach is being applied to serve as the basis for a wide range of multi-agent simulation tasks in the supply chain environments. MASON carefully delineates between model and visualization, allowing models to be dynamically detached from or attached to visualizers, and to change platforms mid-run. We describe the multi-agent simulation system, its motivation, and its basic architectural design. We then compare the impact of the simulation approach to related multi-agent libraries in the public domain, and discuss applications of the system in bullwhip effect reduction in supply chain system. In this paper, we propose a framework for business process simulation based on multi-agent cooperation. Social rationality of agent is introduced into the proposed framework. Adopting rationality as decision making strategies, flexible scheduling of activities is achieved.**

Keywords: **MAS, Simulation Bullwhip effect, Demand forecasting.**

I. INTRODUCTION

In order to understand the source and solution to modern-day business problems, linear and mechanical thinking should give way to non-linear and organic thinking, more commonly referred to as systems thinking. The approach of systems thinking is fundamentally different from traditional thinking methodology and analysis. By definition, analysis means breaking up a problem into constituent parts and finding the solution to each individual part separately.

Multi-agent systems are receiving increasing research attention as affordable computer brawn makes simulation of these environments more feasible. One source of interest has come from social and biological models, notably ones in economics, land use, politics, and population dynamics.

As markets tend to be more and more customer-oriented, the uncertainty connected with end customer demand and its consequences in the supply chain have become an important subject for research. The bullwhip effect is caused by this uncertainty, and several researchers have identified causes to this effect and have tried to propose methods to minimize it.

Chen et al. 1998 and Lee and Padmanabhan 1997 have discussed the main causes of the bullwhip effect. In this paper, we will try to reduce the bullwhip effect using information sharing strategies (centralized information) and breaking order batches (changing the frequency of reordering using two inventory control policies).

Due to the uncertainty and complexity inherent in a supply chain and in inventory control systems, simulation was found a suitable tool to analyze the bullwhip effect (Banks and Malave 1984). Especially the combination of the high-level simulation tool Arena and the procedural programming language Visual Basic for Applications (VBA), proved its usefulness to simulate the systems presented in this paper. The model logic can be represented comprehensibly in Arena, while the more complex calculation algorithms can be programmed in VBA.

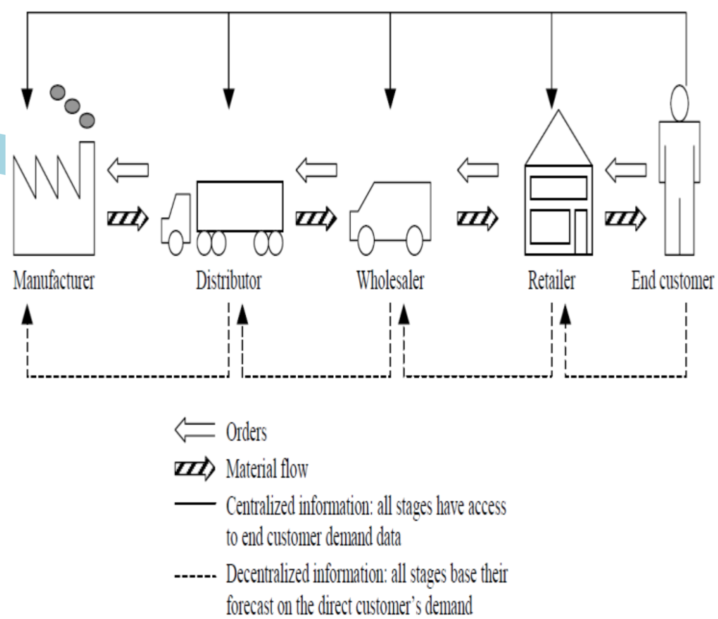

Figure 1: Conceptual Model

If several events are scheduled to occur at a certain 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.

There are many factors said to cause or contribute to the bullwhip effect in supply chains; the following list names a few:

- **Disorganization** between each supply chain link; with ordering larger or smaller amounts of a product than is needed due to an over or under reaction to the supply chain beforehand.
- **Lack of communication** between each link in the supply chain makes it difficult for processes to run smoothly. Managers can perceive a product demand quite differently within different links of the supply chain and therefore order different quantities.
- **Free return policies**; customers may intentionally overstate demands due to shortages and then cancel when the supply becomes adequate again, without return forfeit retailers will continue to exaggerate their needs and cancel orders; resulting in excess material.
- **Order batching**; companies may not immediately place an order with their supplier; often accumulating the demand first. Companies may order weekly or even monthly. This creates variability in the demand as there may for instance be a surge in demand at some stage followed by no demand after.
- **Price variations** – special discounts and other cost changes can upset regular buying patterns; buyers want to take advantage on discounts offered during a short time period, this can cause uneven production and distorted demand information.
- **Demand information** – relying on past demand information to estimate current demand information of a product does not take into account any fluctuations that may occur in demand over a period of time.

Let's look at an example; the actual demand for a product and its materials start at the customer, however often the actual demand for a product gets distorted going down the supply chain. Let's say that an actual demand from a customer is 8 units, the retailer may then order 10 units from the distributor; an extra 2 units are to ensure they don't run out of floor stock.

The supplier then orders 20 units from the manufacturer; allowing them to buy in bulk so they have enough stock to guarantee timely shipment of goods to the retailer. The manufacturer then receives the order and then orders from their supplier in bulk; ordering 40 units to ensure economy of scale in production to meet demand. Now 40 units have been produced for a demand of only 8 units; meaning the retailer will have to increase demand by dropping prices or finding more customers by marketing and advertising. Although the bullwhip effect is a common problem for supply chain management understanding the causes of the bullwhip effect can help managers find strategies to alleviate the effect.

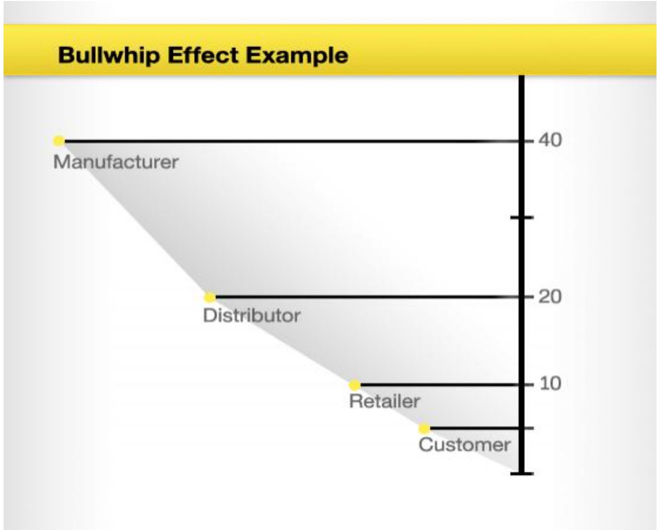

Figure 2 Bullwhip effect example

II. MULTI-AGENT THEORY

Agent is a software entity which functions are proactive and autonomous in a particular environment. Multi-agent system (MAS) is a kind of intelligent system that interconnects separately developed agents, thus enabling the ensemble to function beyond the capabilities of any singular agent in the set-up [3].

There are two fundamental approaches used in modeling multi-agent systems: qualitative (some form of logic, e.g. BDI) and quantitative (e.g. Bayesian). Utility theory is a quantitative one to model MAS. Utility function is a mapping from states of the world to real numbers, indicating the agent's level of happiness with that state of the world. Agents in the competitive MAS potentially have different utility function.

In MAS, as to bounded resources and capability, agent does not stand alone. In accordance with behavior in reality, agent must take action based on certain strategy or rationality. Traditionally, designers have sought to make their agents rational so that they can "do the right thing". Rationality is how the rational decision is made among multiple strategies in the interaction of multi-agent [4].

The predominant theory of rational decision making in agents is that of the economic principle of maximizing the expected gain of actions [5]. Decision theoretic rationality dictates that the agent should choose an action which will maximize the expected utility of performing that action given the probability of reaching a desired state in the world and the desirability of that state [6]. The action that maximizes individual utility may conflict with overall interest (social utility), or redundant actions could be taken due to local utility preference. Hence rationality needs to be considered not only from the individual's point of view, but also from the social perspective. Jennings and Campos proposed the principle of social rationality [7] as follows:

If a member of a respective society can perform an action whose joint benefit is greater than its joint loss, then it may select that action. Here, joint benefit is defined as the benefit provided to the individual plus the benefit afforded to society as a result of an action

Where *Ui*(*aj*) is the individual utility of agent *i* when it takes an action *aj*,*λi* is the weighting given to the individual utility of agent *i*; Σ*Uk'*(*aj*) is the sum of utilities of other agents in the system when action *aj* is taken by agent *i* , *λsoc* is the weighting given to the social utility part of the function.

At a coarse level, equation (1) can be rewritten as:

*U(i,j)=k1*selfUtility*(*ps*)+*k2*publicUtility*(*pp*). (2)

Where *U*(*i, j*) is the utility of agent *i* when it takes action *j*; *k1, k2* are the weighting given to individual utility and public utility respectively. *ps* and *pp* are the key influence parameters for individual utility function and public utility function, e.g. activity's duration, waiting time, priority. The values of *k1, k2* can be altered to implement a wide range of decision making strategies [8].

The proposal of social rationality is to ensure the proceeding of task planning when resource competition appears [9]. Social rationality can be used to guide an agent's decisions. In process simulation, when different activity instances could not share limited resources, competition appears. Thus agent social rationality can be introduced into process simulation to represent the decision making strategies of organizations/departments. Related organization will prefer the activity instances which maximize their predefined rationality utility functions.

III. PRIOR WORK

Potter et al. (2006) considered circumstances where orders were positioned only in multiples of an unchanging consignment extent, for both deterministic and stochastic demand rates. They derived a congested form expression for bullwhip when demand was deterministic. This was authenticated through a straightforward model of a production control system. An expression for bullwhip in a ‘‘pass on orders'' situation with stochastic demand was also derived and validated. Using simulation, they showed the impact of altering batch size on bullwhip in a fabrication control system. Their results showed that a manager might achieve economies through batching while minimizing the impact on bullwhip through the careful selection of the consignment dimension.

Radhakrishnan et al. (2009) developed a novel and efficient approach using Genetic Algorithm which clearly determined the most possible excess stock level and shortage level that was needed for inventory optimization in the supply chain so as to minimize the total supply chain cost. Inventory management was one of the significant fields in supply chain management. Efficient and effective management of inventory throughout the supply chain significantly improved the ultimate service provided to the customer. Hence there was a necessity of determining the inventory to be held at different stages in a supply chain so that the total supply chain cost was minimized. Minimizing the total supply chain cost was meant for minimizing holding and shortage cost in the entire supply chain. This inspiration of minimizing Total Supply Chain Cost could be done only by optimizing the base stock level at each member of the supply chain. The dilemma occurring here was that the excess stock level and shortage level was very dynamic for every period.

Zarandi et al. (2009) presented a Multi-Agent System (MAS) for reduction of the bullwhip effect in fuzzy supply chains. First, it was shown that, even using an optimal ordering policy, without data sharing the bullwhip effect still exists in the supply chain. Then a multi-agent system was proposed to manage the bullwhip effect. The multi-agent system had four different types of agents. The multi-agent system applied Tabu Search algorithm for fuzzy rules generation and a new data filtering method for extraction of training and testing data from the supply chain data warehouse. The results showed that the proposed MAS were capable of managing the bullwhip effect efficiently.

Cimino et al. (2010) discussed that simulation engines of commercial discrete event simulation software used specific rules and logics for simulation time and events management. Difficulties and limitations came up when commercial discrete event simulation software were used for modeling complex real world-systems (i.e. supply chains, industrial plants). The objective of this paper was twofold: first a state of the art on commercial discrete event simulation software and an overview on discrete event simulation models development by using general purpose programming languages were presented; then a Supply Chain Order Performance Simulator (SCOPS, developed in C++) for investigating the inventory management problem along the supply chain under different supply chain scenarios was proposed to readers.

Ghane et al. (2010) presented Robust-Intelligent controller based on sliding mode control theory and neural network to reduce the bullwhip effect in supply chain. A state space model used to design and evaluate the performance of the proposed controller. The neural network control strategy was studied to overcome the "chattering" of the sliding mode controller. The numerical simulations were curried out to check the effectiveness of proposed robust-intelligent controller. The obtained results demonstrated that the proposed controller could effectively suppress the bullwhip effect. Furthermore it was shown that the chattering of sliding mode controller was smoothed when it was integrated with a neural network control strategy.

Duc et al. (2010) measured bullwhip effect in a two-stage supply chain with one supplier and two retailers. The customer demand was assumed to be followed an AR(1) model and is forecasted at each retailer by using the minimum mean square error forecasting method. In addition, the retailers employed the base stock inventory policy. Among the findings of this research, it was interesting to note that the bullwhip effect in supply chains would be minimized as the retailers have the same market share.

Shi et al. (2010) proposed an analytical model to quantify the bullwhip effect by integrating the information sharing and the risk pooling strategies. The developed technique showed that the increase in variability across a three-stage supply chain could be reduced while information sharing and risk pooling were adopted simultaneously. Further, the numerical analysis suggested that their approach outperforms the existing approaches which employed information sharing or risk pooling separately in terms of controlling the bullwhip effect.

IV. MULTI-AGENT SIMULATION

Multi-Agent Based Simulation Systems (MABS) have provided new perspectives on modeling and simulating complex problems. While traditional simulation systems have been limited to a certain class of applications, MABS have employed the powerful concepts of adaptation, emergence and self-organization to model complex, real-world problems. Many domain specific MABS have been developed over the past two decades. Even though these systems have addressed important issues in domains such as social or traffic simulations they are not reusable outside of their application area. On the other hand, the multi-agent system community has spent effort developing generic frameworks for MABS.

These frameworks provide the basic building blocks, i.e., architectures, software components and libraries for the development of a variety of agent-based simulation systems. Unfortunately, none supports the development of MABS where the environment is *open* (i.e., inaccessible, non-deterministic, dynamic and continuous). This represents a major weakness since realistic simulations require the modeling of dynamic environments that can only be partially perceived by the agents.

Over the past several years we have developed a framework for the development of large scale multi-agent based simulation systems for complex domains. The framework called DIVAs (Dynamic Information Visualization of Agent systems) offers reusable architectures, abstract classes, software components and libraries to support the development of enterprisescale simulation systems. DIVAs is based on the premise that agents and environment play an equally important role in MABS. Agents are situated in an open environment that is partially perceived, and the environment is totally decoupled from agents. Such a clear separation of duties leads naturally to extensible, reusable architectures. In addition, DIVAs offers means to dynamically access and modify agent and environment properties at run-time, a unique feature that none of the existing frameworks offers.

As shown in Figure 3, in DIVAs, an agent consists of four main modules [15]. The Interaction Module handles an agent's interaction with external entities and separates environment interaction from agent interaction. An agent communicates with other agents through the Agent Communication Module. It receives environmental data (e.g., agent states, environment object states, external event information) from the Environment Perception Module. The Knowledge Module is partitioned into External Knowledge Module (EKM) and Internal Knowledge Module (IKM). The EKM serves as the portion of the agent's memory dedicated to maintaining knowledge about entities external to the agent, i.e., acquaintances, environment objects situated in the environment. The IKM serves as the portion of the agent's memory dedicated for keeping information that the agent knows about itself (i.e., current state, physical constraints, social limitations). The Task Module manages the specification of the atomic tasks that the agent can perform (e.g., walk, run). The Planning and Control Module serves as the brain of the agent; it uses information provided by the other modules to react to critical situations, plan, initiate tasks, make decisions, and achieve the agent's goals.

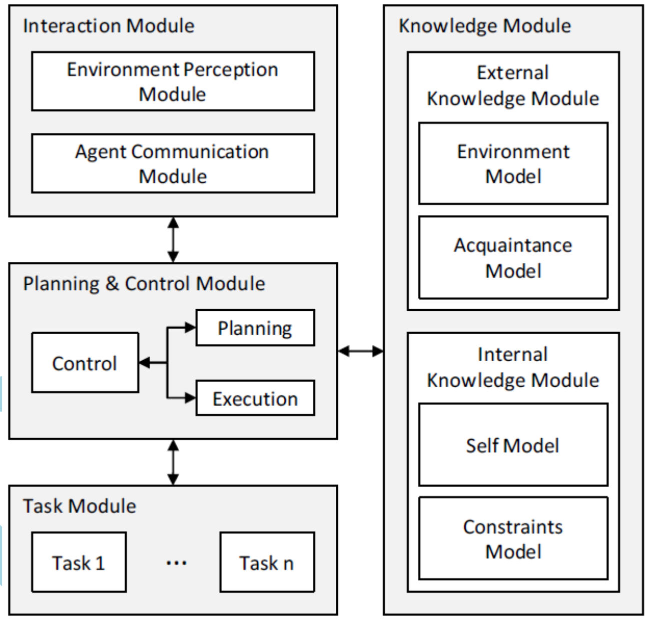

Fig. 3. Agent architecture showing the agent's main components.

V. EXPERIMENTAL IMPLEMENTATION

Aiming to illustrate the applicability of MAS platforms, a washing machine production line will be used as case study to accommodate an agent-based control system that will be modelled and simulated in the Multi-agent simulation environment. The use of simulation in this work has supported the task of specification of an agent-based control system for the process control, adjusting the definition of the autonomous agents' behaviour and the interaction among them.

A. Description of the Case Study

The case study used in this work is a part of a washing machine production line, following a product-driven control approach. This simplified production line is composed by 11 machines that are linked together by conveyors, as illustrated in Fig. 1, including two particularities:

- The first one is centred on a workstation (WS) where a marriage operation occurs, consisting in joining two different components (i.e. Rear Tub and Drum) that arrive from two independent conveyors.
- The other is the existence of an operation that can be performed in one of two available and similar machines (i.e. tub welding machines).

All other operations are single machine operations that are placed on a sequential order, each one having a processing time, according to the type of product to be processed. The production line also comprises a station (WS9, functional tests), where a quality control check is made to all produced products. This station is in charge to run a proper quality check program and the product is labelled with the inspection results for posterior analysis.

The products enter the line with a process plan that must be fulfilled. The process plan is set to the product taking into consideration the variables (e.g. type of the rear tub) and operation parameters (e.g. thickness of welding process) according to the type of washing machine to be manufactured.

B. Implementation Details

The agent-based model to control this production line was developed in NetLogo. The agent-based system is composed by 3 types of agents: Product Agents (PA), Quality Control Agents (QCA) and Resource Agents (RA). The Rear Tub and Drum are examples of PA agents, the machines and conveyors are examples of RA agents and WS9 is a QCA. The behaviour of the PA agent is very simple. Basically the PA is created with a process plan containing the details and sequence of operations that must be fulfilled. During its lifecycle the PA agent will interact with the RA agents in order to guarantee the execution of the product according to the process plan. The results of the operations' execution are stored for posterior analysis and to support traceability..

VII. CONCLUSION

In this case, and since the line is not well balanced and only one tub welding machine is available, a congestion in the upstream sequence of the production line appears, and consequently the MLT is significantly increased due to the time spent by the pallets stocked in the line. Also the WIP parameter is increased.

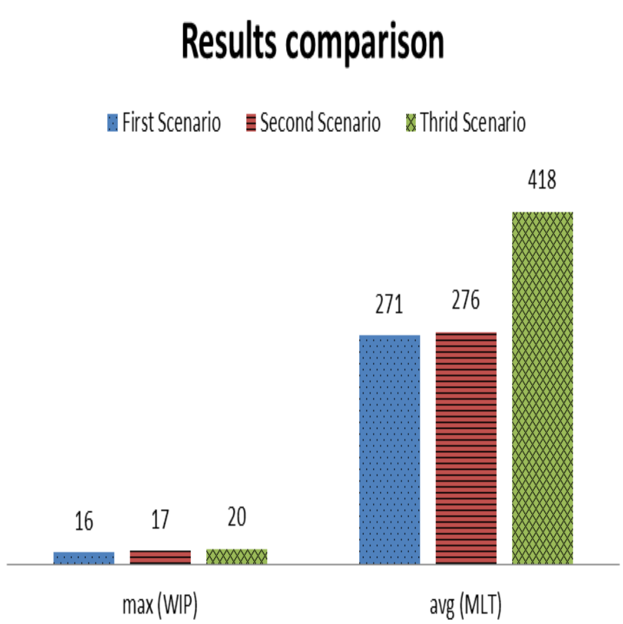

Fig. 5 summarizes the WIP (maximum value) and MLT parameters for the three scenarios simulated..

REFERENCES

[1]. Rachel Mason-Jones, and Denis R. Towill, "Coping with Uncertainty: Reducing "Bullwhip" Behaviour in Global Supply Chains", Supply Chain For u m An International Journal N°1 - 2000 pp. 40-46.

[2]. Christer Carlsson and Robert Full´er, "Reducing the bullwhip effect by means of intelligent, soft computing methods", Proceedings of the 34th Hawaii International Conference on System Sciences – 2001, pp. 1-10.

[3]. Ki-Chan Kim, Il Im, "The Effects of Electronic Supply Chain Design (e-SCD) on Coordination and Knowledge Sharing: An Empirical Investigation", Proceedings of the 35th Hawaii International Conference on System Sciences – 2002, pp. 1-10

[4]. B. Yang And N. Burns, "Implications of postponement for the supply chain", International Journal of Production Research, 2003, vol. 41, no. 9, pp. 2075–2090.

[5]. Lixin Miao, Xiongbo Guo and Zhiwei Miao, "The bullwhip effect in the supply chain and the corresponding solution," Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE, 2003, pp. 1054-1059 vol.2.

[6]. Roger D.H. Warburton, An Analytical Investigation of the Bullwhip Effect, Production and Operations Management, Vol. 13, No. 2, Summer 2004, pp. 150–160.

[7]. Dragana Makajić-Nikolić, Biljana Panić, Mirko Vujošević, ''Bullwhip Effect and Supply Chain Modelling and Analysis Using CPN Tools", Fifth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools'', Aarhus, Denmark, October 8-11, 2004, 219-234.

[8]. Hongnian Yu, H. Comma, Shuang Cang, Wende Chen and Y. Yusuf, "Reducing bullwhip effect using delayed generalized predictive control (DGPC)," Systems, Man and Cybernetics, 2004 IEEE International Conference on, 2004, pp. 1510-1515 vol.2.

[9]. Mihaela Ulieru, "Building Holonic Supply Chain Management Systems: An e-Logistics Application for the Telephone Manufacturing Industry", IEEE Transactions On Industrial Informatics, Vol. 1, No. 1, February 2005, pp. 18-30.

[10]. H. X. Sun and Y. T. Ren, "The impact of forecasting methods on bullwhip effect in supply chain management," Engineering Management Conference, 2005. Proceedings. 2005 IEEE International, 2005, pp. 215-219.

[11]. Rachel Croson, Karen Donohue, "Upstream versus downstream information and its impact on the bullwhip effect" System Dynamics Review , Vol. 21, No. 3, pp. 249-260, 2005.

[12]. Rachel Croson, Karen Donohue, "Behavioral Causes of the Bullwhip Effect and the

Observed Value of Inventory Information", Management Science 00(0), pp. 1–14, 2005

[13]. de Kok Fred Janssen, Jan van Doremalen, Philips Electronics Synchronizes Its Supply Chain Interfaces Vol. 35, Issue 1, pp. 37–48, ©2005 INFORMS

[14]. T. Kawagoe and S. Wada, "The bullwhip effect: a counterexample," Intelligent Agent Technology, IEEE/WIC/ACM International Conference on, 2005, pp. 124-127.

[15]. S. Geary, S.M. Disney, D.R. Towill, "On bullwhip in supply chains—historical review, present practice and expected future impact", International Journal of Production Economics, Vol. 101, 2006, pp. 2–18

[16]. M. H. Fazel Zarandi, M. Pourakbar and I. B. Turksen, "An Intelligent Agent-based System for Reduction of Bullwhip Effect in Supply Chains," *Fuzzy Systems, 2006 IEEE International Conference on*, Vancouver, BC, 2006, pp. 663-670.

[17]. Z. Lei, Y. j. Li and Y. q. Xu, "Chaos Synchronization of Bullwhip Effect in a Supply Chain," *Management Science and Engineering, 2006. ICMSE '06. 2006 International Conference on*, Lille, 2006, pp. 557-560.

[18]. Andrew Potter, Stephen M. Disney, "Bullwhip and batching: An exploration", International Journal of Production Economics Vol 104 (2006) pp. 408–418.

[19]. Janusz K. Grabara, Marta Starostka-Patyk, "The Bullwhip Effect In Supply Chain", Advanced Logistic Systems, ALS vol. 3 - Chapter 1, 2007, pp. 127-133.

[20]. Alony, I, Munoz, A, The Bullwhip effect in complex supply chains, International Symposium on Communications and Information Technologies ISCIT 2007, 17-19 Oct, 1355-1360.

[21]. P. Radhakrishnan, Dr. V.M.Prasad, Dr. M. R. Gopalan, "Genetic Algorithm Based Inventory Optimization Analysis in Supply Chain Management", 2009 IEEE International Advance Computing Conference (IACC 2009) Patiala, India, 6-7 March 2009, pp. 418-422.

[22]. M. H. F. Zarandi, M. Avazbeigi and I. B. Turksen, "An intelligent fuzzy Multi-Agent System for reduction of bullwhip effect in supply chains," *Fuzzy Information Processing Society, 2009. NAFIPS 2009. Annual Meeting of the North American*, Cincinnati, OH, 2009, pp. 1-6.

[23]. Dr. Lee, Tzong-Ru (Jiun-Shen), Powerful Supply Chain: apply areas of social and science management to supply chain management to enhance the coordination of supply chain Participants, In proceeding of 2010 8th International Conference on Supply Chain Management and Information Systems (SCMIS), Date 6-9 Oct. 2010, pp. 1-6.