

UTILIZING SIMULATION TO EVALUATE BUSINESS DECISIONS IN SENSE-AND-RESPOND SYSTEMS

Paul Huang
Young M. Lee
Lianjun An
Markus Ettl
Steve Buckley

Karthik Sourirajan

School of Industrial Engineering
Purdue University
West Lafayette, IN 47906, U.S.A.

IBM T.J. Watson Research Center
1101 Kitchawan Road
Yorktown Heights, NY 10598, U.S.A

ABSTRACT

Simulation can be an effective way to evaluate alternative decisions in Sense-and-Respond systems prior to taking actions to resolve existing or anticipated business situations. In Sense-and-Respond systems, business situations arise within predefined contexts that specify what aspects of the business need to be monitored and what information is needed to make decisions. We have designed a decision support system that dynamically configures simulation models based on business context and interactively presents simulation results to business analysts. In this paper, our decision support system is applied to the IBM Demand Conditioning process, in which mismatches between supply and demand are identified and corrective actions are initiated.

1 INTRODUCTION

The term “Sense-and-Respond enterprise” refers to a company that adaptively evolves to keep up with a rapidly changing business environment. Adaptive enterprises are driven by the need to preserve or expand competitive advantages that are threatened by intense competition and changing customer preferences. Stephan Haeckel (1999) of IBM Advanced Business Institute described the concept of Sense-and-Respond enterprise.

A Sense-and-Respond enterprise needs to continuously monitor its business processes, detect existing or anticipated business issues, and resolve these issues in nearly real time. To do so, advanced business process monitoring capability and decision-making support are essential.

This paper describes a Sense-and-Respond system we developed for IBM’s Personal Computing Division (PCD) to support their Demand Conditioning process. Demand Conditioning is a decision-making process that proactively monitors discrepancies between demand and supply and

recommends corrective actions before a potential exception becomes a threat to customer service.

We use simulation techniques to support business users to evaluate alternative decisions in our Sense-and-Respond system prior to taking any corrective actions to resolve a business situation. Different business situations may require different simulation models. Our Sense-and-Respond system deploys a Service Oriented Architecture (SOA) to accommodate different simulation techniques that can be used to satisfy different simulation requirements.

In this paper, we will show the architecture of our Sense-and-Respond system. We tested our system using a variety of business scenarios; the results of these scenarios are also presented.

The rest of the paper is organized as follows: Related work is discussed in Section 2. Section 3 gives an overview on the Demand Conditioning process. Section 4 describes the high-level architecture of our Sense-and-Respond system, focusing on the decision support subsystem. Sections 5 and 6 present simulation results for two different simulation techniques: discrete event simulation and systems dynamics simulation. Section 7 compares these two techniques. Section 8 concludes the paper.

2 RELATED WORK

Stephan Haeckel (1999) of the IBM Advanced Business Institute described the concept of a Sense-and-Respond organization. Lin et al. (2004) applied Sense-and-Respond concepts to supply chain management.

Systems dynamics simulation has been widely used to study causality in real life scenarios. Angerhofer et al. (2000) review the use of systems dynamics modeling in supply chain management. They classify the research and development efforts into three main categories – Modeling for Theory Building, Modeling for Problem Solving and

Improving the Modeling Approach – and go on to review the related work and identify future research directions. Sterman (2000) discusses a variety of scenarios such that can be studied using systems dynamics simulation, not limited to supply chain management. An et al. (2004) propose a systems dynamics model with real time process control for Sense-and-Respond supply chain management.

3 DEMAND CONDITIONING

Demand Conditioning is a decision-making process within the IBM Personal Computing Division (PCD) that monitors mismatches between demand and supply for each IBM PC component and recommends corrective actions before a potential exception becomes a threat to customer service.

The Demand Conditioning process is triggered by an imbalance between supply and demand of PC components. To effectively detect existing and potential imbalances, a Sense-and-Respond monitoring system has been deployed in PCD to proactively monitor the supply and demand profile of every component. When an imbalance is detected, the monitoring system raises situations, which are then classified and presented to the decision makers through a business dashboard. To resolve an imbalance situation, the decision makers need to decide which corrective actions to take. These actions fall into the following three categories:

- *Procurement supply conditioning.* Focuses on working with suppliers to improve flexibility in supply to react to customer demand that is difficult to predict.
- *Demand conditioning.* Focuses on providing a dynamic sales plan in the sense that it can be changed in reaction to supply imbalances. It considers pricing actions and promotions to provide incentives to customers to choose alternatives.
- *Offering conditioning.* Focuses on identifying alternative products or substituting PC components that can be provided to customers in reaction to supply imbalances. It is supported by a proactive product definition phase that provides more flexibility to define product configurations.

It is possible for several of these actions to be executed simultaneously. The selected actions form a solution. The decision makers need to consider all potential solutions for a given situation and choose the most promising one. In this paper we describe a simulation capability to help evaluate different solutions.

4 SYSTEM ARCHITECTURE

As shown in Figure 1, our Sense-and-Respond system consists of a Business Dashboard, a Proactive Monitor, a Simulation Manager, a Sense-and-Respond Database,

Simulation Models, Service Mediation, and Simulation Engine plug-ins.

The Sense-and-Respond Database stores the current and historical supply and demand data of each PC component.

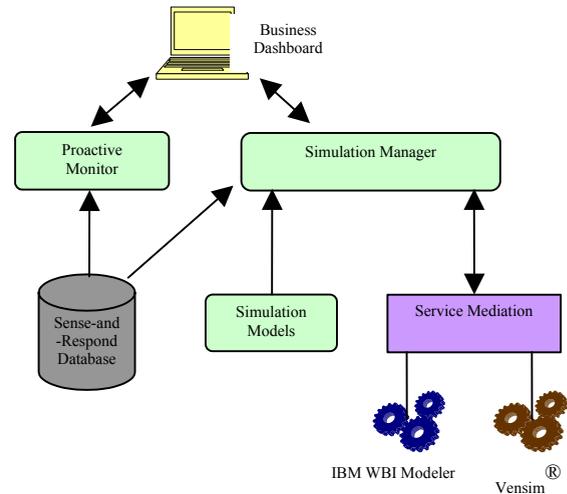


Figure 1: Architecture of a Sense-and-Respond System Using Different Simulation Engine Plug-ins to Evaluate Alternative Decisions

The Proactive Monitor identifies potential gaps between supply and demand by using information stored in the Sense-and-Respond Database and future indicators to forecast order trends.

If a potential mismatch is detected, the Proactive Monitor raises a business situation, which is then classified and presented to decision makers through the Business Dashboard. At this point, the decision makers invoke the Simulation Manager to evaluate alternative decisions.

The Simulation Manager receives simulation requests from the decision makers through the Business Dashboard. For each request, the Simulation Manager chooses a simulation model that is suitable for this request from the Simulation Models repository, initializes this model using relevant data from the Sense-and-Respond Database, and sends a service request along with the model to the Service Mediation, which will then find a proper Simulation Engine to execute the model and return the result to the decision makers through the Business Dashboard.

This design follows a Service Oriented Architecture (SOA). Different simulation engines are plugged into the Sense-and-Respond system by exposing them as a service and registering themselves to the Service Mediation. For the purpose of evaluating the Demand Conditioning decisions, we plugged in two types of simulation engines: WBI Modeler (IBM Corporation), a discrete event simulation engine, and Vensim[®] (Ventana Systems Inc.), a systems dynamics simulation engine.

Figure 2 shows the functional components within the Simulation Manager and their interactive pattern. When a

discrepancy between demand and supply is detected by the Proactive Monitor, the system generates a dataset that describes the alert situation. This alert dataset along with a simulation request is given as input to the Simulation Manager. The Simulation Manager does not need to simulate the entire business, but needs to focus only on the parts of the business that are affected by the particular alert situation. The Model Generator will pick up a proper model from the Simulation Models repository based on the alert situation and simulation request, initialize the model using the input alert dataset and other information in the Sense-and-Respond Database, including all affected finished products and components, relevant Bills of Materials (BOMs), demand forecasts, supply commits, and on-hand inventory of all affected materials. Depending on the current context, different response actions may be simulated by several different simulation models.

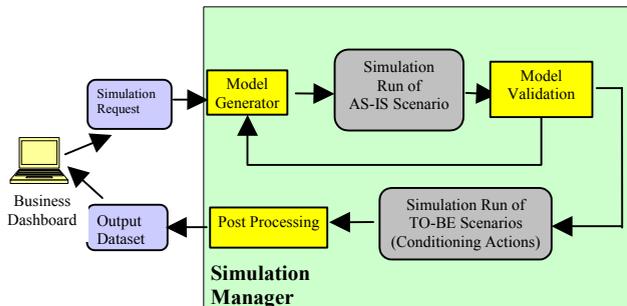


Figure 2: Architecture of Simulation Manager

When the input dataset is passed to the Simulation Manager, one or more simulation models are dynamically configured and executed using the simulation engine. In the Simulation Manager, there are several generic simulation models, each of which can simulate a class of response actions. For all these generic simulation models, simulation parameters are all externalized into a well-structured data format. The simulation parameters include a set of end products and technology components, Bills of Materials, demand forecasts, supply commits, initial inventories, inventory policies, allocation policies and the duration of the simulation. For each response action, an appropriate generic simulation model is selected and its simulation parameters are configured according to the situation and input dataset.

After the model is initialized, external simulation engines are called to execute the model. We first simulate the AS-IS scenario, which represents the current alert situation. Unless the AS-IS scenario is correctly simulated, the simulation results of the TO-BE scenarios (response actions) are not trustworthy. There is a utility called Model Validation, shown in Figure 3, which adjusts simulation parameters until the AS-IS scenario is correctly simulated. Once the simulation model is validated, response actions are simulated. This can be a simulation of one particular action or a sequence of several actions. For each action, simulation runs are replicated to enable statistical output analysis.

After simulation runs are completed, a Post Processing utility is triggered to summarize simulation runs. This utility generates an output dataset, which typically contains inventory profiles, back orders (or lost sales), lead times to shipment, etc. The output dataset is passed to and displayed on the Business Dashboard. Upon review, business decision makers can evaluate additional actions through the simulation module.

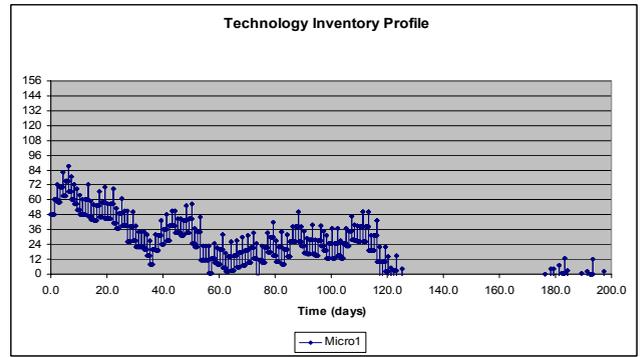


Figure 3: Inventory Profiles of Technology Components for an Alert Situation

5 DISCRETE EVENT SIMULATION

In this section, we present several scenarios that are simulated using the WBI Modeler. The details of the simulation model is described in another paper by Lee (2004).

5.1 Alert Situation

In this sample alert situation (Figure 3), a demand surge is expected for microprocessor #1 (Micro1). As a result, shortage of Micro1 is expected to appear on around 120th day and last until 200th day. The shortage of Micro1 is shown in the Figure 4, the total expected shortage (back order or lost sales) is 3,583 units.

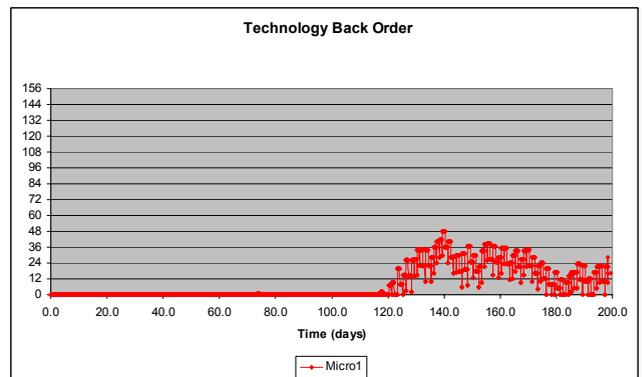


Figure 4: Back Orders of Technology Components for an Alert Situation

5.2 Conditioning Action 1: Shifting Demand to Alternative Technology Components

In this scenario, the conditioning team decides to evaluate a demand conditioning action, which is shifting demand to alternate technology components, e.g., from Micro1 to Micro2 and Micro3 for the duration of the simulation time, which is 200 days. From the SaR system, a relevant information is gathered as an input dataset and simulation module is called to evaluate this action. After simulating this action, the result (Figure 5) indicates that with this particular action, the majority of the shortage for Micro1 would be eliminated, but the shortage, although much smaller at only 152 units, would still occur at later time, between 170th day and 200th day. (The shortage can be seen in Figure 5 as points below zero on the y-axis.) The simulation result also indicates that there are no new constraints which occur for the Micro2 and Micro3 as a result of this action. The simulation result is displayed in the dashboard, and the response team considers an additional action to resolve the remaining shortage of Micro1.

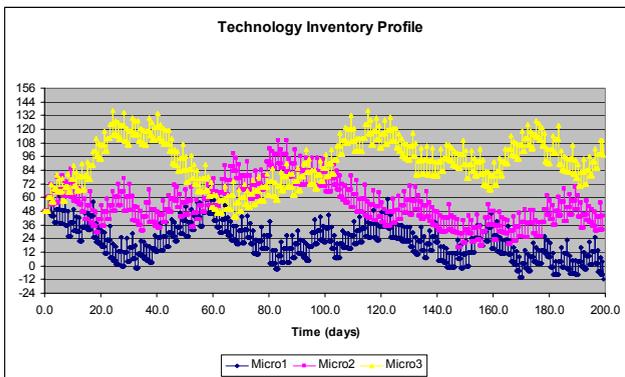


Figure 5: Inventory Profiles of Technology Components after Conditioning Action 1 (Demand Shifting)

5.3 Conditioning Action 2: Secure Additional Supply

In this scenario, the conditioning team concludes that it is not possible to further condition the demand, and that a proper next step is to secure additional supply from a new supplier for Micro1 to remove the expected constraint of Micro1. Also, the conditioning duration is decided to be only the first two weeks.

The model simulates the action and returns the result shown in Figure 6 which indicates that the shortage for the Micro1 would be completely eliminated as a result of this action. The conditioning team is now confident that the sequence of the two actions resolve the alert situation, and makes plan to implement the actions.

6 VENSIM[®] MODELS

In this section, the simulation engine used is Vensim[®].

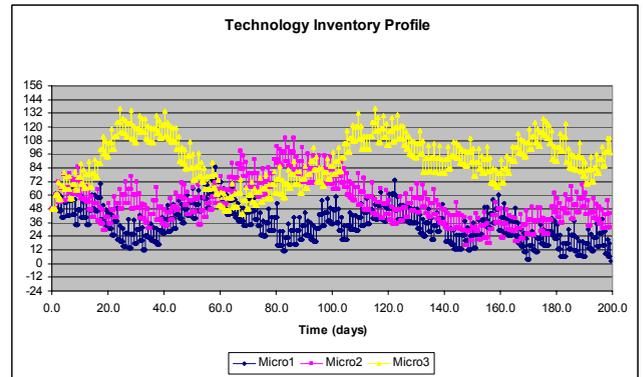


Figure 6: Inventory Profiles of Technology Components after Conditioning Action 2 (Secure Additional Supply)

6.1 Conditioning Action 3: Price Changes and Additional Supply

In this section, we implement price change actions on demand to match supply and demand. In parallel to the price change actions, we also allow additional supply to be obtained within a limited range. As in the previous section, such actions are taken when alerts are produced regarding supply-demand gaps. Though Vensim[®] is traditionally used for performing continuous time systems dynamics simulations, we implement discrete time simulation using Vensim[®] by adjusting the time step accordingly.

We adopt a simple price-demand relation among several related technologies and associate the price change actions and additional supply procurement with some objective, defined through a payoff function. The objective of the model is to maximize the system profit (defined as revenue – cost). We set up the problem so that the price of each technology can be changed within a certain range. A one time additional supply can be obtained within a given amount at an additional cost. We use the Vensim[®] optimization mode to pick the prices and the amount of additional supply to be procured so as to maximize the value of the payoff function. In order to minimize the supply-demand gap, we look at the difference between excess inventory and backlogs and try to get their difference as close to zero as possible.

6.2 System Dynamics Formulation

In Figure 7, we show the model setup as a stock-flow diagram using Vensim[®]. The stocks, denoted by rectangles, represent variables that can accumulate. We have *Backlog*, *Supply* and *Cum Profit* as stocks in our model. The valves along with arrowed double-lines represent flows into or out of stocks. We have *Demand Rate*, *Fulfillment Rate*, *Replenish Rate*, *Shipment Rate* and *Profit Rate* as flow rate in or out of stocks in our model. By connecting variables with arrows, the causal relationships are entered. After entering the formula for each variable using the Equation Editor, we can simulate the system and demonstrate its be-

havior. The input to the model is the Original Demand, Original Price and the Original Supply. The decision variables are the Change in Price for each of the related technologies (components) and the Additional Supply.

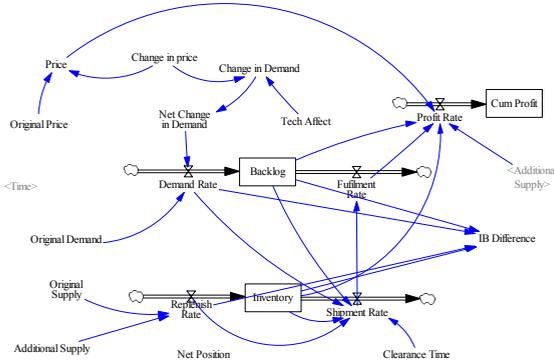


Figure 7: System Dynamics Model for Price Change Action

Note that we use subscript capabilities of the software to represent multiple technologies in the model. Most of the variables are arrays, except for profit, which is aggregated from the others by summing over the various technologies.

6.3 Price Demand Model

We use price change to influence the change of demand rate. The *Demand Rate (DR)* is formulated as:

$$DR[i] = OD[i](1 + NCD[i]), \quad i = 1, \dots, n$$

$$NCD[i] = \sum_{j=1}^n DemandFunc(CP[j] * TA[i, j]), \quad i = 1, \dots, n$$

where i and j refer to the technologies under consideration, OD is for the *Original Demand*, NCD for *Net Change in Demand*, CP for *Change in Price*, and TA is for *Tech Affect*. The TA is defined as:

$$TA[i, j] = \begin{cases} 1, & i=j \\ -\frac{\alpha}{n-1}, & i \neq j, 0 < \alpha \leq 1 \end{cases} \quad (1)$$

The *DemandFunc* is defined as in Figure 8. This is just an initial assumption to begin modeling. From the definition of TA and the *DemandFunc*, it follows that the demand for certain technologies will decrease and the demand for related technologies will increase as the price of the technology arises.

6.4 Test Instance Discussion

We use an instance in which there are four related technologies – P0, P1, P2 and P3. The initial data on Forecasted Demand and Supply suggests that the technologies P0 and P1 are going to be in shortage and P2 and P3 are going

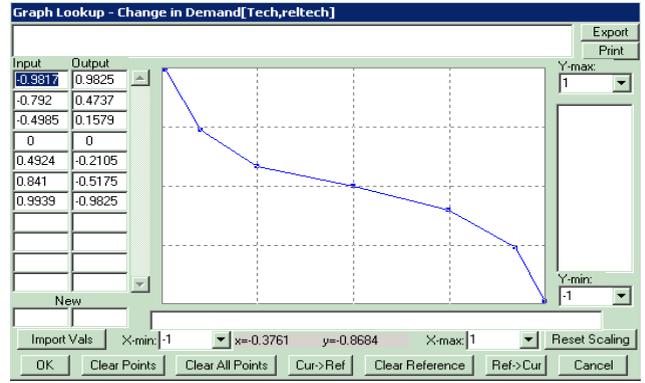


Figure 8: Demand Function – Percentage Change in Price vs. Percentage Change in Demand

to have a lot of coverage in the next 16 weeks. This produces an alert that is captured and sent to the demand conditioning team. The team decides to use price changes to match supply to demand. So, they use the Vensim® model developed to find out the required changes to the price to match the supply to the demand in a better way. They use a payoff function that puts a charge of \$150 on Backlogs, and \$150 on Excess Inventory per unit. The additional procurement cost is \$50 per unit. The initial prices for P0, P1, P2 and P3 are \$184, \$157, \$190 and \$145 respectively. α was set to be 0.4. We allow a one time procurement of a maximum of 100 units for any technology in period 8. Also, the price can be changed only within 50% of the current price (increase or decrease). Figure 9 below shows the graph of (Inventory-Backlog) for the current situation. We ran the optimization using Vensim® and found that by increasing the price of P1 by 33%, decreasing the price of P3 by 31% and obtaining 79 additional units of P0 in period 8, we are able to maximize our profit and generate a better (Inventory-Backlog) profile for the technologies as shown below in Figure 10. Figure 11 shows the comparison between the original demand and the modified demand.

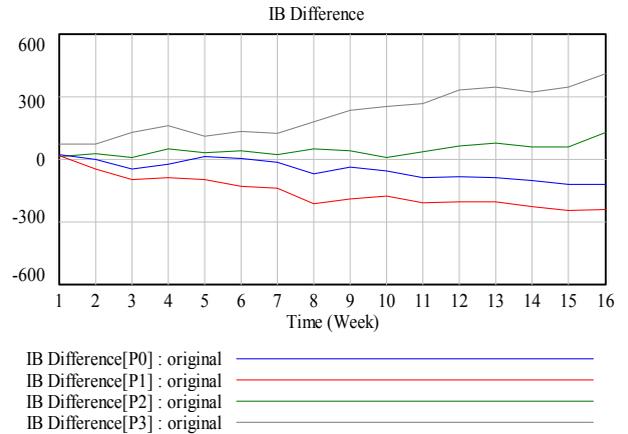


Figure 9: (Inventory – Backlog) before Price Change Action

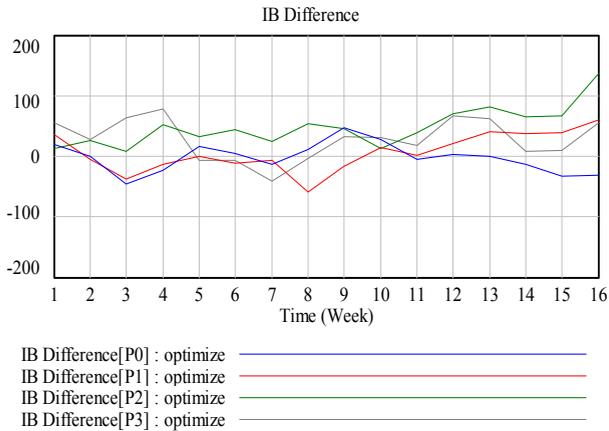


Figure 10: (Inventory – Backlog) after Price Change and Additional Supply Procurement Actions

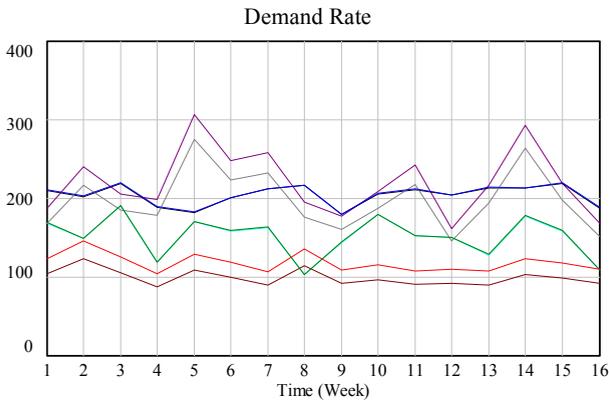


Figure 11: Comparison of Original and Changed Demands (Parallel Lines Indicate Demand Shifting)

No price action was taken on P0 and P2 but still their demand is affected due to the actions taken on the related technologies. Further, it should be noted that the unit costs on inventory and backorder play an important part in bridging the supply-demand gap and should be set accordingly. Also, in order to further improve the supply demand gap, we could take more actions. For example, in the case of P0, further action would help remove the backlog that can be seen at the end. Also, shifting supply to earlier periods if possible would help improve the backlog situation for P1 and P3. The effect of such actions along with the cost can also be estimated using simulations.

6.5 Future Extensions to the Vensim® Model

At this stage, this is a toy problem that was constructed to illustrate the use of Vensim® models in examining the effect of demand conditioning actions. We have made some naïve assumptions about the behavior of the system to simplify the model, which can be replaced with more realistic assumptions later. The first extension to the above model would be to incorporate realistic system behavior to

price changes. This can be done by modifying the change in demand relation according to the result of a learning component. Another issue with the model is that demand is specified at the component level. A more realistic model would capture the demand at the product level and the supply at the component level. The price changes would happen at the product level which would be translated into demand changes at the component level.

7 COMPARISON OF VENSIM® AND DISCRETE EVENT SIMULATION

In this section we will discuss the use of Vensim® and discrete event simulation in developing models for testing demand conditioning actions.

The traditional use of Vensim® in business is to model strategic behavior using continuous time simulation. Vensim® does numerical approximations in its calculations when performing the continuous time simulations. By adjusting the time step accordingly, we can make Vensim® perform discrete time simulations as well. Though the actual model may not look like a discrete simulation model, the results of the simulation can sometimes be made to match those of discrete simulation models. Some advantages of using Vensim® for modeling the effect of demand conditioning actions are given below.

- The effects of various feedbacks and other relations can be represented and studied with ease using Vensim®. Hence control activities can be modeled with ease using Vensim®. This is an important factor contributing towards the usefulness of Vensim® in making strategic decisions.
- Quick comparisons on the effect of various strategic scenarios/parameter settings can be performed using Vensim® by utilizing the SyntheSim mode. This enables the user to quickly get the effect of changing various parameters instead of restarting simulations.
- The ability to perform quick sensitivity analysis and display the results in a clear graphical pattern is another strongpoint of Vensim® in terms of using it to study the effect of various actions on system performance.

Having discussed the advantages of Vensim® it is also appropriate to discuss the advantages of discrete event simulation.

- Modeling complex business policies for capturing operational level details is easier using discrete event simulation.
- Simulation is executed only when events occur and not at each and every time step as the event calendar drives the simulation. This may result

in significant savings in run time for large simulation models.

- Tracking of individual entities and their attributes can be done easily using discrete event simulation. This might be a requirement under certain conditions, for example, when we want to track certain orders and obtain their start and end times and so on.

Given these points, it is appropriate to use discrete event simulation models for certain conditioning actions and Vensim[®] for certain other actions. The choice of any tool would depend on what the action entails in terms of both modeling and in terms of the results.

8 CONCLUSION

Simulation is an effective way to evaluate alternative decisions in a Sense-and-Respond system. We developed a Sense-and-Respond system that adopts a SOA to accommodate different simulation techniques for business decision support. The general architecture of the decision support system can be applied to integrate other decision support services such as optimization, rule-based analysis and business intelligence.

REFERENCES

- An, Lianjun, J.J. Jeng, Markus Ettl, and Jen-Yao Chung. 2004. A System Dynamics Framework for Sense-and-Respond System. To appear in *the proceeding of IEEE Conference on e-Commerce Technology for Dynamic E-Business (CEC'04 EAST)*.
- Angerhofer, Bernard J. and Marios C. Angelides 2000. "Systems Dynamics Modeling in Supply Chain Management: Research Review", *Proceedings of the 2000 Winter Simulation Conference*, 342-351. Ed. J.A. Joines, R.R. Barton, K. Kang and P.A. Fishwick. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Haeckel, S. 1999. *Adaptive Enterprise: Creating and Leading Sense-and-Respond Organization*. Harvard Business School Press, Cambridge, MA.
- Lee, Y. M. 2004. "Simulating Availability Outlook for e-Commerce Business of Personal Computer Sales", *the Proceedings of the 2004 Winter Simulation Conference*, Ed. R.G. Ingalls, M.D. Rossetti, J.S. Smith, and B.A Peters. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Lin, G., Jeng, J.J., Wang, K. 2004. Enabling Value Net Collaboration. In: *Evaluation of Supply Chain Management*, Chang, 417-430, Y.S., Makatsoris, H.C., and Richards, H.S. (eds.), Kluwer Academic Publishers.
- Sterman, John D. 2000. *Business Dynamics. Systems Thinking and Modeling for a Complex World*. Boston: McGraw-Hill.
- Vensim: Ventana Systems Inc. Available online via <www.vensim.com> (Accessed on July 20, 2004).
- WBI Modeler: IBM Corporation. Available online via <www-306.ibm.com/software/integration/wbimodeler> (Accessed on July 20, 2004).

AUTHOR BIOGRAPHIES

PAUL HUANG joined IBM as a Research Staff Member at the Watson Research Center in 2003 after receiving his Ph.D. degree from Carnegie Mellon University. His research interests cover Supply Chain Management, Stochastic Optimization, Games and Machine Learning. He is currently working on analytics models and technologies for Business Process Management. Dr. Huang's email is <puhuang@us.ibm.com>.

YOUNG M. LEE has been working in the mathematical science department of IBM's T.J. Watson Research Center since 2002 in the areas of supply chain simulation and optimization. Prior to joining IBM, he had worked for BASF Corporation for 14 years, where he had founded and managed the Mathematical Modeling Group, and led development of numerous optimization and simulation models for various logistics and manufacturing processes. He has a B.S., a M.S., and a Ph.D. degree in Chemical Engineering from Columbia University in the City of New York. His research interest includes simulation and optimization of supply chain and manufacturing processes. His email address is <ymllee@us.ibm.com>.

LIANJUN AN is currently working on IT Infrastructure Design for Business Performance Management and System Dynamics Modeling of Supply Chain in the Analytic Models and Architecture department at IBM's Thomas J. Watson Research Center. He received Ph.D. degree in Applied Mathematics from Duke University in 1991. He worked on Analysis of Granular Flow and Plastic Deformation in McMaster University of Canada (1992), and Scientific Simulation of Oil Reservoir on Parallel Computer in the State University of New York at Stony Brook (1994). He subsequently joined IBM and worked on Network Configuration Management System, Websphere Commerce Suite and Grid Computing Projects since 1998. He is a member of the Society of Industry and Applied Mathematics. His email address is <alianjun@us.ibm.com>.

MARKUS Ettl is a Research Staff Member at IBM's T.J. Watson Research Center. He joined IBM Research in 1995 after receiving his doctoral degree in Computer Science in 1995 from Friedrich-Alexander University in Erlangen, Germany. Since joining IBM, Dr. Ettl focused on advanced research in supply chain management. His research interests include operations research, simulation and decision support for manufacturing logistics, and sense-

and-respond business management for adaptive organizations. Dr. Ettl's email is <msettl@us.ibm.com>

KARTHIK SOURIRAJAN is a summer intern in the Mathematical Sciences department of IBM's T.J. Watson Research Center and has been working on developing models for analyzing Sense-and-Respond supply chain systems. He is currently doing his PhD in Industrial Engineering at Purdue University, West Lafayette, working on optimization models for supply chain network design with lead time and safety stock considerations. His research interests include optimization and analysis of dynamic supply chain systems. His email address is <souriraj@ecn.purdue.edu>.

STEVE BUCKLEY has been a Research Staff Member at the IBM Thomas J. Watson Research Center in Yorktown Heights, NY since 1987, and a manager at that facility since 1995. He currently manages the Analytic Models & Architecture department in the Mathematical Sciences organization. His most recent interest is in Sense-and-Respond systems. His team has implemented Sense-and-Respond for several IBM lines of business. Dr. Buckley received the Ph.D. degree in Computer Science from MIT in 1987. He also received the M.S. degree in Computer Science from Penn State in 1978, and the B.S. degree in Applied Mathematics and Computer Science from Florida State in 1977. His e-mail address is <sbuckley@us.ibm.com>.