Beer distribution game: A simulation using Java agents and MySQL

Roger J Jenkins
Lecturer, University of Technology, Sydney

Geoff Breach
Technical Officer, School of Management UTS

Abstract

The Beer Distribution Game is widely used in the teaching of Supply Chain Management and other complex interacting systems. The lessons illustrated by the game are accepted as having a useful level of validity, and interesting cases have been noted that support the notion that the game captures the dynamics of real systems. This paper is a preliminary description of a novel approach to the modelling of the game. Our model has been developed using conventional Java based agents, but persistent data has been stored in the Open Source relational data base management system (RDBMS), MySQL. This model forms the first stage of a project that will use this framework to examine a range of rules and system design features that can be used to define multi level supply chains. This approach will also enable the simulation component of the model to be interfaced to real systems that store dynamic information in RDBMS.

Introduction

The area of agent based modelling of supply chains has been an area of interest in some recent publications. Southwest Airlines is reported as achieving savings of more than $10million on a range of logistics based costs as an outcome of agent based modelling approaches to the scheduling of material transfers in its business (Bonabeau and Meyer, 2001). Successes are noted in a range of other applications by Bonabeau (2002, 2003). Agent based modelling may be an appropriate strategy for the analysis of systems characterized by chaotic complexity. Agent based modelling can be grounded in specific programming environment. A widely reported application is Swarm; a Java based system for modelling systems with many agents. Repast is a similar system, and there are many others. Many of these systems are not simple to master. Axelrod (1997) suggests that Swarm requires a substantial level of programming knowledge to master, and a standard programming language is often used to avoid the task of mastering a package such as Swarm. While Axelrod suggests a procedural language such as Basic or C, we have chosen Java. This language was chosen as it embodies the concept of an agent (as an object) within the paradigm of the programming language. It is expected that Java will then offer fewer barriers to model development than a language with a less developed object orientated paradigm.

The basic concept of this simulation approach is that the complex behaviour of systems containing many agents can be a result of the interaction of those agents, operating autonomously with simple rules. Axelrod (1997) refers to emergent behaviour as large scale effects arising from the local actions of agents in the system. Emergent behaviour has been applied to many fields including organizational studies (Axelrod, 1997; Anderson, 1999; MacIntosh and MacLean, 2001). In this paper we use agent based modelling to examine the behaviour of a simple, widely used system in the teaching of operations management; the Beer distribution game BDG. Key features of the BDG are reviewed in a later section.

A key goal in our development of an agent based simulation of the BDG was to use programming languages and software tools that are 'free'. The software development community distinguishes between different interpretations of 'free' using the metaphors '*free as in speech*' and '*free as in beer*'. Free software is generally accepted to be software that is provided with a license that grants the end user the *freedom* to run the software, the *freedom* to study and modify the software, the *freedom* to redistribute the software, and the *freedom* to make public any changes or improvements that they make. Free Software is, by definition, *free as in speech*. Most free software is also *free as in beer* – that is to say there is no fee or charge for its acquisition ("The Free Software Definition", Free Software Foundation, http://www.gnu.org/philosophy/free-sw.html) Most important in this context was an absence of financial and legal encumbrance by way of commercial software products, and an ability to directly translate the techniques dis cussed herein into computing environments in common use in business today. The agent based BDG is written in Java, and stores its operating data in a MySQL database. Some initial development was made with the Perl programming language and the graphs and diagrams were generated using Microsoft Excel, but OpenOffice suite of programs will be used in further work in this project. OpenOffice is a suite of software that includes a word processor, a spreadsheet and graphing program, and presentation tools. OpenOffice was released as free software by Sun Microsystems in 2000 and is a competitor to Microsoft's 'Office' package. OpenOffice is available for Windows, Linux and Macintosh at http://www.openoffice.org/.

Java is a programming language designed and developed by Sun Microsystems that saw its first commercial release in 1995. Central to Java's design is the concept of portability – the principle that software developed in Java will run equally well on many different computing platforms ("Java 2 Platform", Sun Microsystems, http://java.sun.com/java2/whatis/index.html). Java runs on a great many platforms including Microsoft's Windows, Apple Computer's MacOS and OSX, Sun Microsystems' Solaris and most versions of Linux.

The Structured Query Language (SQL, formally known as SEQUEL2) is an English keyword based language for defining, manipulating and querying relational databases (Chamberlin et al., 1976). SQL, or some variant thereof, is the language used within the vast majority of modern commercial database systems. The agent based BDG uses SQL to communicate with the MySQL database server. The MySQL database server is 'free' software that is published under both commercial and free licenses. MySQL is designed with speed and stability in mind. The MySQL company "media kit"(http://www.mysql.com/press/index.html) claim that MySQL's success can be gauged by its acceptance by companies that includes the New York Stock Exchange, the University of California, the University of Texas and NASA. MySQL is available for Microsoft Windows, Apple OSX and many versions of the UNIX operating system at http://www.mysql.com The agent based BDG makes use of the MySQL database server, and the MySQL Java Connector.

Description of the modelling approach

Mihram (1972) described the process of developing a model as having five stages of development. The process used to develop the model reported in this paper will be described within the framework of these five stages.

1 Systems analysis

The BDG is well described in the literature (see for example: Levy et. al., 2003; Sterman, 1989; Forrester, 1992; Senge, 1992; Lee et al, 1997; and in particular Croson & Donahue, 2002) and so the description used for this paper will be very brief. The way in which the game is organized for play is illustrated in fig. 1.

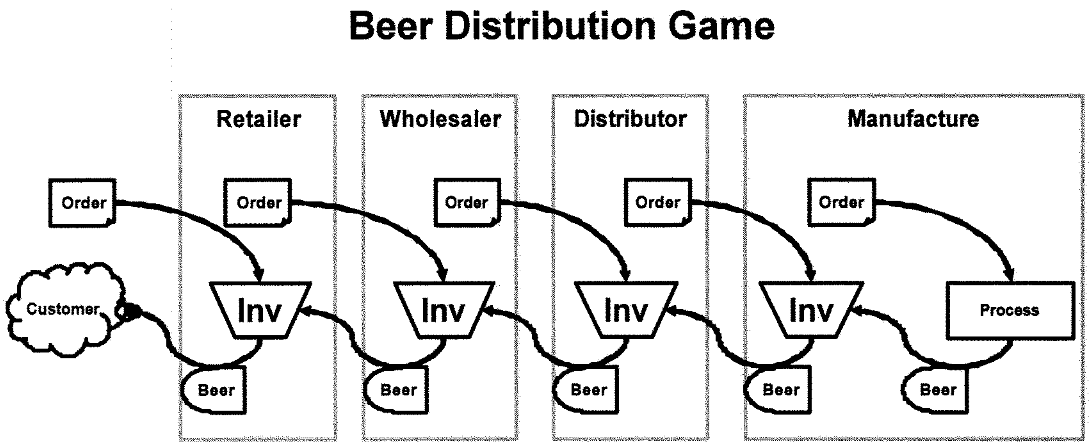


Figure 1 Beer distribution game layout

The game simulates a system of four independent agents within a simple supply chain. A customer, from outside the chain places orders on a *retailer*. The *retailer* supplies the customers with cases of beer, and attempts to maintain their capacity to supply the customer by ordering supplies of more beer from a *wholesaler*. The *wholesaler* operates similarly to the *retailer*, and orders their supply of beer from the *distributor*. The *distributor* is resupplied from a *manufacturer*. The *manufacturer* produces more cases of beer from within their own factory. When a case of beer is shipped by a supplier it will take two weeks to reach the next agent down the supply chain, similarly, an order will take two weeks to reach the next agent up the supply chain.

People playing the game have complete visibility of all products on the playing field, but no visibility of orders in the system. The critical role of the players is to develop a strategy that will enable them to maintain effective supply capability; they may only do this by managing the levels of inventory in their stage of the supply chain. The players may not modify the structure of the supply chain; problems related to process structure and agent mindset form the major grounds for the debrief of the game. System analysis for this project is a relatively simple matter. A project that is based in a real situation is never so simple. The advantage of selecting this strategy for this stage of the project is that it allows more attention to be directed at modeling issues; rather than on the scope of the problem and the purpose to which the model's output will be put. Given the highly defined nature of the reference system the next stage of the process is relatively straightforward.

2 System synthesis

This refers to the construction of a representation of the reference system that captures the essential logics and interactions that are of interest to the clients of the modeling process. This stage will typically include the collection of data required to establish the parameters of the model. The parameters for this project are set by the published parameters of the game.

The essential aspect of this project is in the selection of the appropriate modeling strategy. Logistics systems, and the BDG, have been modeled previously. Typically a process orientation is taken for the models. In the case of a global crude oil logistics system Jenkins (1995) chose to use *Siman,* a commercial discrete event simulation application, to build the model. Sterman (1989) has developed a series of models of dynamic systems in a programming environment typified by the software package *iThink*, and this also is a modeling environment that has a focus on the process through which entities such as cases of beer pass. In these software applications the focus is on the process workstations, and the entities that pass through these workstations. These modeling environments can become very difficult to use when the reference situation includes people who are influencing the nature of decisions taken in the reference situation (Jenkins, et al, 1998).

3 Verification

System synthesis may produce some form of representation of the reference system. This might be diagrammatic, discursive, or it might be in some structured form such as UML. Mapping this representation into the models programming environment, such as specific Java code, used in the modeling framework is a crucial stage. The ease with which the human mind can cope with ambiguity and exception is only realized when a model needs to be written in some form of computer based modeling environment. Verification is the stage where the operation of the code developed is checked until the modeler is satisfied that the model is an accurate expression of the system representation. This of course does not imply that the model is a valid representation of the reference system.

```
In TICK 99 results index#4 as agent#4
 Delivery received by agent #4 of 8 units
 Inv of 16 units for agent 4 before shipment received
 Inv after shipment of 8 units rec for agent 4 is 24
Previous demand 8
 current backorder = 0
 so despatch = 8
 new backorder = 0
 Closing inventory is = 16
 Shipped out to agent #5 an amount of 8
 Allowing for the amount onOrder = 16
 Using rule #1
 Order calculated by unresponsive agent #4 against a
targetInv of 40 is 8 units
 Order put by agent #4 on agent #3 is 8 units
```

Exhibit A

Two main strategies were followed as the BDG model was developed for this paper. The first strategy was the use of an object orientated language and the use of object orientated agents. Entities in the model were developed as agents, and the agents contained methods and properties. This approach enabled verification to be achieved on small sections of the overall model as the model was developed. The second strategy was the development of an extensive log reporting of properties as the code was executed. A section of the logging output is shown in Exhibit A.

4 Validation

Validation compares the output of the model with data collected from the reference system. Validation is grounded in the work of stage 2; system synthesis. Good levels of interaction between modelers and clients at the system synthesis stage will facilitate effective validation, and if this is not effective then the scope of the model may be quite different to that envisaged at the early stages of the project; validation then becomes very difficult. This is a particularly difficult problem if the reference system contains social agents that are empowered to take autonomous decisions in the process being modeled (Jenkins et al, 1998).

In this paper we argue that our model has captured the non social dimensions of the game. The output of the model reflects, at a face level, the appearance of output from typical students groups that play the game. This paper is however a preliminary paper that establishes the basic model, and its basic validity.

Ultimately the model will be used to explore aspects related to the reflexive social agents included in the reference system; we do not claim validity for that aspect of the model at this stage.

5 Model use and experimentation

This stage sees the model used to explore some aspect of the reference system. In this paper we report on the use of the model to examine aspect of the BDG. Although other researchers have argued that the BDG reflects aspects of real supply chains (Lee et al, 1997); we do not. Our argument is that the model reflects aspects of the game, and we leave till later work the extension of the model to real systems. We argue in particular, that the inclusion of SQL functionality in the model will facilitate the extension of the model in further work to the area of real supply chains. Typically, simulation models will operate within a self contained system, where data is sourced from dedicated files, and is reported to dedicated files (this is the case, for example, for ProModel, Witness, and iThink; at the time of writing of this paper). This will restrain the model to its own system, making an interface to operating systems difficult. We envisage that at a later stage we will be able to connect our simulation modules to operational RDBMS, enabling real time operation of the model from within the supply chain system. The benefits of this strategy will flow from the incorporation of a wider range of rules for order calculation, and the addition of standard optimization routines such as simulated annealing and genetic algorithms and function (as described for example by Downsland, 1993 and Ghanea-Hercock, 2003).

Results

Typical student performance

The plot of inventory for each of the four levels of the supply game, when played by university students, will typically have the appearance of that shown in fig. 2 The retailer (Agent 4 for our model) has an inventory that varies from about 18 to -22. The wholesaler (Agent 3) has a wider variance, from about 30 to -38, and the distributor (Agent 2) has a further increase variance from about 70 to -20. The factory (Agent 1) has a lower variance of about 38 to -18.

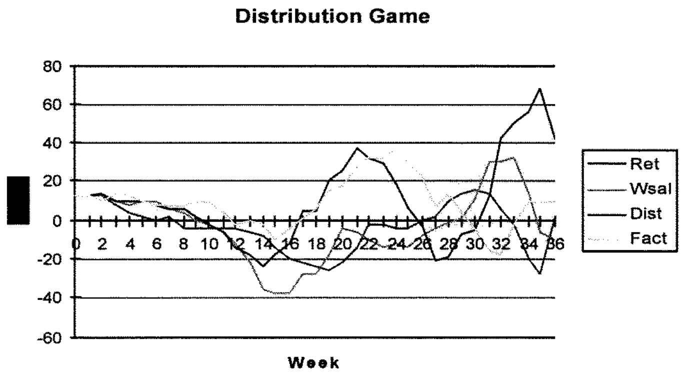


Fig. 2 Student performance

This effect is generally referred to as the bullwhip effect (see for example Lee et al, 1997), and it is an effect that has been observed in real logistics systems (Lee et al, 1997). Lee et al (1997) outlined four major causes of this effect:

- Demand forecast updating

- Order batching.
- Price fluctuation leading to order batching.
- Rationing and shortage gaming leading to order batching in anticipation.

Approaches that could reduce the bullwhip effect include a less frequent updating of demand forecasting and increased weight to the demand of the final consumer rather than to agents within the supply chain. Any strategy that reduces batching in the supply chain; batching can be influenced by transportation policies, lot pricing, returns policies and shortage management. We have developed a model that demonstrates the first two effects, and in principle we argue that it can be developed in such a way as to illustrate further effects in supply chains that arise from the impact of reflexive social agents.

We report the result for seven runs of the model in this paper, and then discuss the significance of these results and the importance of the integration of the Java model and the storage of model data in the MySQL relational database management system (RDBMS). Each run was carried out with, wherever possible the same conditions. Specific details of the model and its code can be obtained from the first author. The run was initialized to exactly the same opening position, and run for a period of 100 ticks. The term 'tick' is often used in the area of agent based modeling, and in this case a single tick is equivalent to a single cycle of operations in the BDG. Agents in the supply chain were referred to by an index, where the higher the index, the closer the agent to the consumer. The Manufacturer was index 1, and the Retailer was index 4. The customer was index 5. Plots are presented only of the inventory for each agent as this represents the dynamics of the system quite effectively. Unless specified, the demand for each run is 4 units per tick for the first two ticks, and then 8 units per tick for the remainder of the run; this is identical to the normal form of the BDG.

Run 1 Unresponsive agents

The simplest scenario to model is that where the agents in the supply chain simply set out to control their inventory at a given, fixed level. This run set this target inventory at 40 units, thus enabling the system to operate with positive stocks for most ticks (where a tick represents one week in the game). This represents a system where the agents do not batch, and do not update demand forecasts.

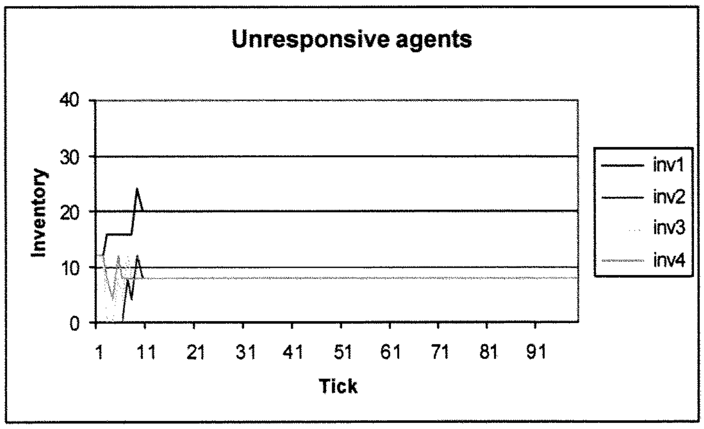


Fig 3 Unresponsive agents

Inventories vary in the first ten weeks of operation and then are completely stable for all agents. Agent 1 has a different level of inventory as it has a shorter delivery loop for resupply than the other agents.

Run 2 Responsive agents

In this run the agents 2 and 3 update demand forecasts after each tick, attempting to control inventory as a multiple of the previous demand. Agents 1 and 4 are unresponsive. There is no attempt to smooth the forecast.

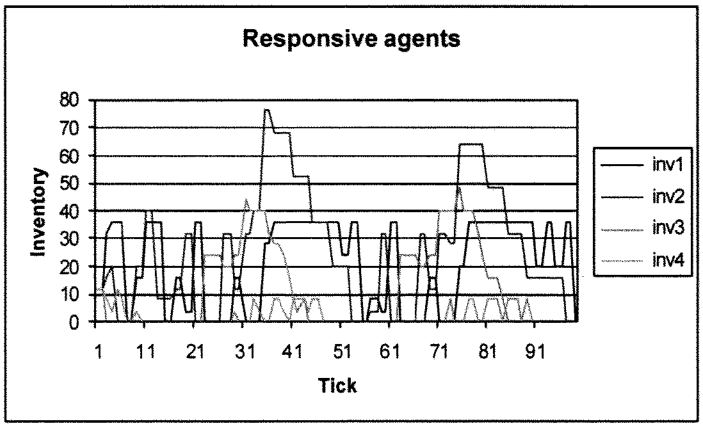


Fig 4 Responsive agents
In this run the supply chain exhibits the classic bullwhip effect. Agent 4 varies between 0 and about 10; agent 3, the retailer, varies between 0 and about 45; agent 2, the wholesaler, varies between 0 and about 70; and agent 4, with the capacity to control its own supply side loop varies between 0 and about 35. Note that in this chart the Y Axis has a different scale to that used in the other figures due to the higher levels of inventory used by the agents in this run.

Run 3 Forgetful agents

Students playing the game will often forget how much material they have on order. This run was used to examine the impact of this on Unresponsive agents. Agents 2 and 3 did not allow for material on order when they calculated their next order.

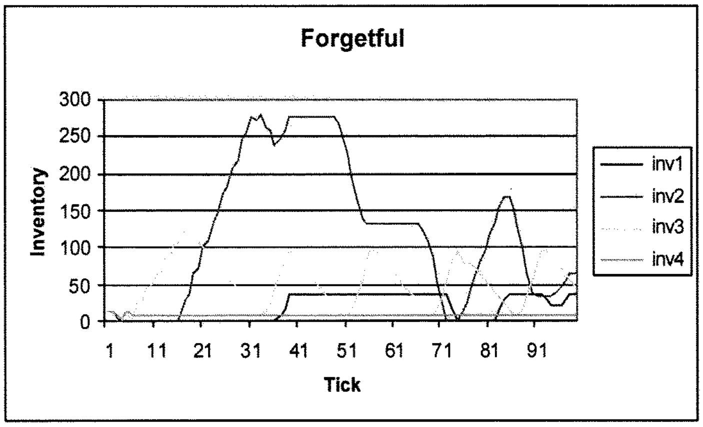


Fig 5 Forgetful agents

Agents that forget how much material they have on order demonstrate the presence of the bullwhip effect, and very high levels of peak inventory.

Run 4 EOQ agents

Agents in this run are unresponsive but agents 2 and 3 do not place an order unless they require at least 15 units; just less than two weeks demand of 16 units.

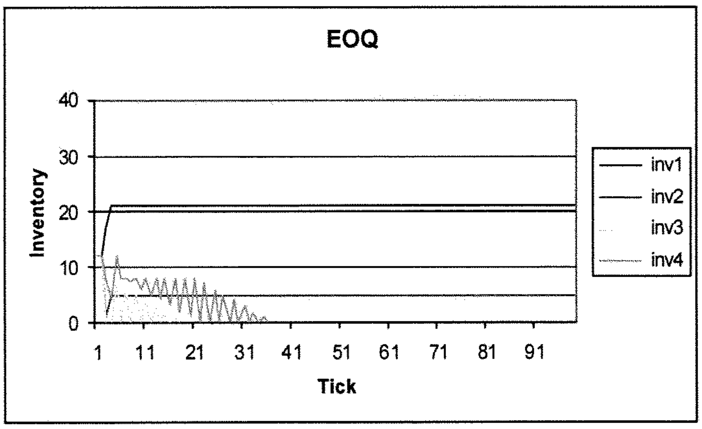


Fig. 6 EOQ agents

In this run; agents 1 and 2 appear to have stabilized rapidly, agents 2 (Wholesaler) and 3 (Distributor) oscillate to a backorder state from which they do not recover.

Run 5 Global responsive

Agents in this run are responsive, but use customer demand to update demand forecast. This can be compared to run 2, but the run is limited by the fixed demand profile of this scenario.

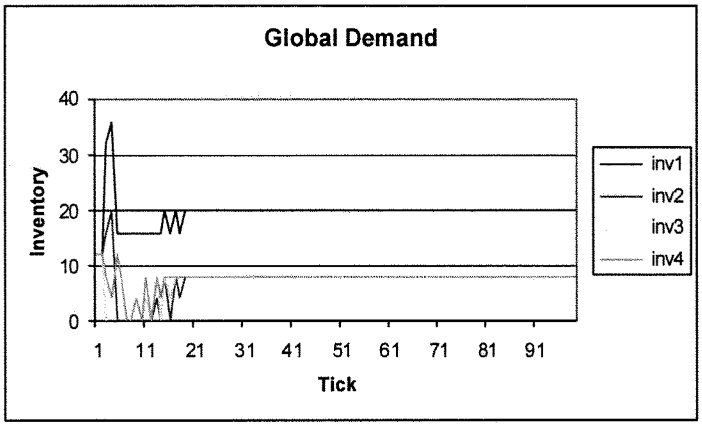


Fig. 7 Agents using customer demand driven forecasts

There is no evidence of a bullwhip effect, and levels of inventory are about 20 for agent 4, and about 8 for all other agents. This is similar to run 1, the unresponsive agent. The chart shows that this has allowed the system to avoid the problems that occurred with the responsive agents using local demand as the driver of demand forecasting systems.

Run 6 and 7 Variable demand

Variation in demand was introduced for these two runs. Responsive agents using global demand information were compared to unresponsive agents.

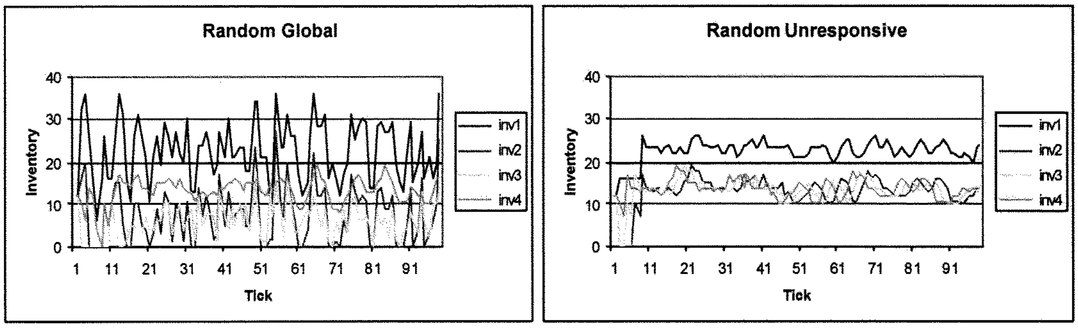


Fig. 8 The effect of variable demand

Both strategies have given reasonable inventory traces for the run, and there is no evidence of the bullwhip effect. These results are an elaboration of runs 2, and 5. The chart on the left indicates that if the system requires responsiveness to cope with variable demand, then if the excessive inventories found in run 2 are to be avoided than the demand forecasts will need to be driven by customer demand; not supply chain agent demands.

Discussion

The behaviour of this model, under the range of different rules used to calculate orders, indicates that the model is a valid model of the BDG system. The bullwhip effect is readily observed under rules that are used by students. The dominant behaviour of students in the game is that of responsiveness and forgetfulness. The combination of these two strategies is sufficient to cause high inventories, variable inventories, and a bullwhip effect up the supply chain. Particular aspects of the results are interesting. It was expected that the base system would not have achieved such stable state in the short time of 10 weeks. The results of run 2, responsive agents compares well to typical results. The indication is that reflexive behaviour, within a rational context, explains much of the erratic behaviour of the system. The extreme impact of a forgetful strategy, if it were to be overlaid on a responsive strategy is also consistent with some of the extreme inventory levels experienced by some student groups.
The relatively stable behaviour of the EOQ run is a surprising result. Upstream agents have managed to maintain inventories, whereas the downstream agents have gone into permanent backorder status. While further analysis of this result is required it appears that some level of synchronization of ordering is indicated, as suggested by texts in the area of supply chain management (such as Chopra and Meidle, 2004). The run of the globally responsive agents is not surprising. In most debriefs of the game, participants identify the crucial importance of linking local decisions to customer demand. This strategy, in the absence of disruptive batching strategies will enable stable performance up and down the supply chain. Runs 6 and 7 were carried out in order to examine the impact of variations in demand on the capability of the global strategy. The results illustrate the performance of the strategy, suggesting that it is similar to the unresponsive strategy. Within the context of the model this would not appear to be of great value. In the actual performance of the game however, and probably in real supply chains, this might be of more importance. It is likely that a strategy based on simply targeting a fixed inventory level will be difficult to be defended in most businesses, whereas a responsive strategy might appear to be more valuable. If this is the case; then the responsive strategy can be effectively executed, but only on customer demand data and not on supply chain agent data.

Finally, the model demonstrates the capability of Java to develop valid models of complex systems, and further demonstrates the capability of developing simulation models in Java which use SQL and RMDBS as the means of making decisions and storing persistent data. Both of these outcomes are important if the models ultimately are to be embedded within actual operating systems.

References

Anderson, P. 1999, 'Complexity theory and organizational science', *Organization Science,* vol. 10, no. 3, pp. 216-232.

Axelrod, R. 1997, *The Complexity of Cooperation*, Princeton University Press, Princeton.

Bonabeau, E. 2003, 'Don't trust your gut', *Harvard Business Review,* no. May, pp. 116-123.

Bonabeau, E. 2002, 'Predicting the unpredictable', *Harvard Business Review,* no. Mar, pp. 109-116.

Bonabeau, E. and Meyer, C. 2001, 'Swarm intelligence: A whole new way to think about business', *Harvard Business Review,* no. May, pp. 106-114.

Chopra, S. and Meindle, P. 2004, *Supply Chain Management, 2nd ed*, Pearson Education International, New Jersey.

Croson, R. and Donahue, K. 2002, 'Experimental economics and Supply-Chain management', *Interfaces,* vol. 32, no. 5, pp. 74-82.

Dowsland, K.A. 1993, 'Some experiments with simulated annealing techniques for packing problems.', *European Journal of Operational Research.* vol. 68, no. 3, pp. 389-399.

Forrester, J.W. 1992, 'Policies, decisions and information sources for modelling', *European Journal of Operational Research,* vol. 59, pp. 42-63.

Ghanea-Hercock, R. 2003, *Applied evolutionary algorithms in Java*, Springer-Verlag, New York.

Jenkins, Roger. 1995, Short term crude oil pricing: Simulation of a global energy system in the discrete modelling environment, Symposium on the modelling and control of national and regional economies, Gold Coast, 8 pages,

Jenkins, Roger, Deshpande, Y., and Davison, G. 1998, Verification and Validation and Complex Environments: A study in Service Sector., Proceedings of the 1998 Winter simulation conference, 1433-1440.

Lee, H.L., Padmanabhan, V. and Whang, S. 1997, 'The bullwhip effect in supply chains', *Sloan Management Review,* no. Spring, pp. 93-102.

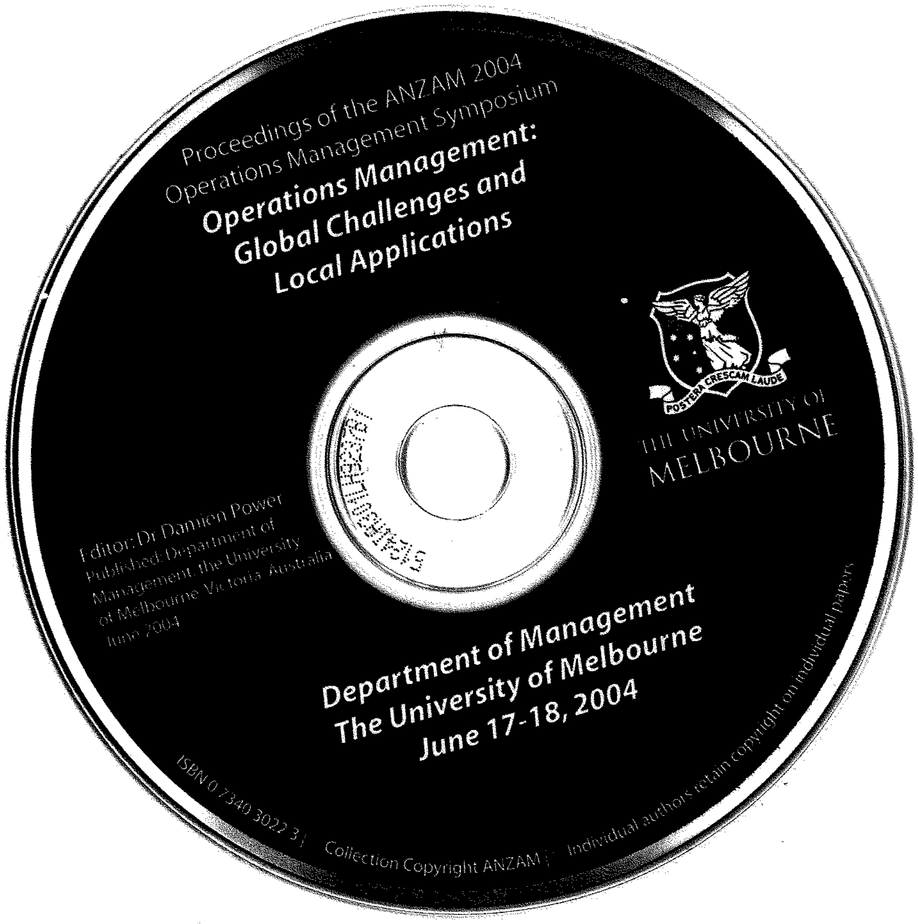
MacIntosh, R. and MacLean, D. 2001, 'Conditioned emergence: researching change and changing research', *International Journal of Operations & Production Management,* vol. 21, no. 10, pp. 1343-1357.

Mihram, G.A. 1972, 'Some Practical Aspects of the Verification and Validation of Simulation Models', *Operations Research Quarterly,* vol. 23, pp. 17-29.

Senge, P.M. 1992, *The Fifth Discipline: the art and practice of the learning organization*, Random House, Sydney.

Simchi-Levi, D., Kaminski, P. and Simchi-Levi, E. 2003, *Designing and managing the supply chain 2ed*, McGraw Hill, Sydney.

Sterman, J. 1989, 'Modeling managerial behaviour: Misperceptions of feedback in a dynamic decision making experiment', *Management Science,* vol. 35, no. 3, pp. 321-339.


Proceedings of the ANZAM 2004
Operations Management Symposium
Operations Management:
Global Challenges and
Local Applications
POSTERA CRESCAM LAUDE
THE UNIVERSITY OF
MELBOURNE
Editor: Dr Damien Power
Published: Department of
Management, the University
of Melbourne Victoria Australia
June 2004
Department of Management
The University of Melbourne
June 17-18, 2004
ISBN 0 7340 3022 3 | Collection Copyright ANZAM | Individual authors retain copyright on individual papers

THE UNIVERSITY OF MELBOURNE

Proceedings of the ANZAM 2004 Operations Management Symposium

- Index of Papers

The Symposium

The symposium was held at the University of Melbourne, Australia on the 17th and 18th of June 2004. Dr Damien Power convened the symposium with the support of Professor Danny Samson and colleagues from the Department of Management at the University of Melbourne.

Members of the Organising Committee

Professor Danny Samson, The University of Melbourne
Associate Professor Mile Terziovski, The University of Melbourne
Dr Prakash Singh, The University of Melbourne
Dr Sharafali Moosa, The University of Melbourne
Dr Damien Power, The University of Melbourne
Professor Amrik Sohal, Monash University
Professor Ross Chapman, University of Western Sydney
Dr Douglas Davis, University of Technology, Sydney
Associate Professor Paul Hyland, Central Queensland University
Dr Roger Jenkins, University of Technology, Sydney
Dr Bishnu Sharma, University of the Sunshine Coast
Professor Nevan Wright, Auckland University of Technology
Dr Ross Milne, Auckland University of Technology

Conference Theme - Operations Management: Global Challenges and Local Applications

The pressures of global competition, technological, social and political change, and expanding markets have created significant challenges for the management of operations in all sectors. Managers are confronted with the need to be both locally responsive and globally competitive. As such, the effective management of operations becomes a strategic imperative and potential source of competitive advantage. The recognition of this fact has led many organisations to look not only at operations as an internal function, but as a set of interacting and interrelated processes. This view has also led to a focus on processes not just within the firm, but between firms. In this context this symposium aims to act as a focus for operations management research by providing a venue to present current research, as well as providing a meeting place to explore collaborative research opportunities.

Refereeing Process

Papers included in these proceedings were subjected to a double-blind refereeing process. Full versions of papers were submitted for the refereeing process. Each paper, after first having the identity of its author(s) removed, was forwarded to two appropriate reviewers for evaluation. The organising committee wish to express their sincere thanks to the many academics who reviewed papers for this symposium.

Model Citation

The following model citation is based on the first refereed paper in these proceedings.

> Beckett, R.C., (2004) Collaborative innovation: The Views of Some Industry R&D Managers, *Proceedings of the ANZAM 2004 Operations Management Symposium*, ed. D. Power, Faculty of Economics and Commerce, The University of Melbourne, Australia.

Additional Copies of CD-ROM

Additional copies of this CD can be obtained from Peggy Hui at the ANZAM Secretariat for $25 (includes GST) email: peggy.hui@uts.edu.au

Publication Details

ISBN: 0 7340 3022 3

Editor: Dr Damien Power
Published By: The Department of Management, Faculty of Economics and Commerce, The University of Melbourne, Victoria, Australia.
Date: June 2004

This CD-ROM contains those papers that were both accepted for the refereed stream and presented at the conference.

www.management.unimelb.edu.au

Proceedings of the ANZAM 2004 OPERATIONS MANAGEMENT SYMPOSIUM

Contents (CD will automatically open to Table of Contents)

Publisher: Department of Management, the University of Melbourne, Victoria 3010, Australia
Publication Date: June 2004
ISBN: 0 7340 3022 3