Proceedings of the 1998 Winter Simulation Conference
D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

AN EXPERT SYSTEMS APPROACH TO SIMULATING THE HUMAN DECISION MAKER

Stewart Robinson

Operational Research and Systems Group
Warwick Business School
Warwick University
Coventry, CV4 7AL
UNITED KINGDOM

John S. Edwards

Operations and Information Management Group
Aston Business School
Aston University
Birmingham, B4 7ET
UNITED KINGDOM

Wu Yongfa

Department of Management Engineering
South East University in Nanjing
Jiangsu
P.R. CHINA

ABSTRACT

Many simulation models include elements of human decision making which present some difficulty to the simulation modeller. It is often difficult to determine how a human goes about making decisions, and even where this is possible, representing this within the constructs of a simulation package may be problematic. In this paper an expert systems approach to representing the human decision maker is proposed. An example of an expert system linked to a simulation model is given. Not only was it possible to train the expert system by programming a set of decision rules, but also by obtaining examples of decisions through running and interacting with the simulation model. The paper concludes by discussing the future directions of this research.

1 INTRODUCTION

Many simulation models include some elements of human decision making. Typically these involve scheduling and allocation rules. For instance, a production supervisor needs to determine a week's production schedule, or to consider which staff to allocate to particular tasks at the start of a shift. In rail yard operations, goods wagons are allocated to particular sidings by the yard supervisor. In warehouse and distribution operations the allocation of stock is often determined by a supervisor, as well as the allocation of lorries to loading/unloading bays. Customers determine which route to take around a supermarket based on a complex set of decision rules. Indeed, since most simulation models represent human activity systems, as opposed to purely automated systems, there is almost always some element of human decision making.

This paper discusses how expert systems could be used to represent these elements of human decision making in simulation models. First, some previous work that has linked simulations and expert systems is described. In the following section the difficulties of representing a human decision maker are discussed as well as the general approach that could be used in simulation modelling. Following this, an example of an expert system linked to a simulation model is described. The paper concludes by discussing the future of this research.

2 EXPERT SYSTEMS IN THE LIFE-CYCLE OF SIMULATION STUDIES

It has been proposed that expert systems could aid the development and use of simulations throughout the life-cycle of a simulation study (Doukidis and Angelides, 1994). Indeed, there are examples of expert systems being applied at every stage of a simulation study, from model conception to experimentation and the analysis of results. An early attempt at automating the development of conceptual models can be found in Doukidis and Paul (1985). Later, however, it is conceded that intelligent front ends probably provide a less rigid and, therefore, more useful approach (Doukidis and Angelides, 1994).

Input data modelling provides a role for expert systems in the simulation life-cycle. Hurrion (1993a) trains a neural network with an empirical distribution and proposes that the approach might be used to generate random variates for a simulation model.

In terms of model development, there have been attempts at using expert systems to automatically generate simulation program code, for instance, CASM (Balmer and Paul, 1986) and Mathewson (1989). Expert systems have also been used for model verification and validation. Doukidis (1987) uses an expert system, SIPDES, to help locate and resolve compilation errors in simulation programs. Deslanders and Pierreval (1991) develop a system with limited capability for aiding model validation.

As an aid to experimentation and results analysis, there is considerable scope for applying expert systems. For instance, Hurrion (1991) uses an expert system to aid the design of experiments. He also employs a neural network to analyse a simulation model's output (Hurrion, 1992; 1993b) and as a basis for simulation optimisation (Hurrion, 1997).

3 REPRESENTING HUMAN DECISION MAKING

The presence of human decision making within simulation models presents two problems to the simulation modeller. First, it is necessary to determine the way in which the decisions are made by the people involved and, second, it is important that the decision making process is modelled as accurately as possible. It is probably the first of these that presents the greatest problem. For instance, when one of the authors (Robinson) was investigating the modelling of an engine assembly facility, it became apparent that it was all but impossible to determine how different supervisors allocated staff to machines on different shifts. What was apparent though was that some supervisors were more effective than others in their allocation decisions.

The typical approach to representing human decision making in simulation models is to try to elicit the decision rules from the decision maker. In some cases this amounts to little more than a guess on the behalf of the modeller. Following this, the rules are included in the model using the constructs of the simulation language or simulator. This normally requires the use of a series of ‘if’, ‘then’, ‘else’ statements. This can result in large amounts of code that is difficult to interpret and even harder to change.

One approach to overcoming these problems might be to use an expert system to represent the human decision maker, and link it with a simulation model. Indeed, some have already attempted to do this (Flitman and Hurrion, 1987; O’Keefe, 1989; Williams, 1996; Lyu and Gunasekaran, 1997). This could be implemented in two ways:

- elicit the decision rules from the expert and represent them within an expert system
- use the simulation model to prompt the expert to make decisions, building up a set of examples from which an expert system could learn

These correspond to the two fundamental approaches to knowledge acquisition for any expert system: elicitation by human knowledge engineer and machine learning from examples, respectively. The first approach would employ the constructs of an expert system and so make it easier to accurately represent the decision process. It should also be easier to interpret and easier to change since expert systems are specifically designed to facilitate this. In this way the approach should aid model development. What it does not provide, however, is a simple means for knowledge elicitation. This remains a well-known problem in expert systems generally (Waterman, 1986). It is in the second approach where the link to a simulation model could provide significant advantages. Most work on machine induction (e.g. Hart (1987)) treats the set of examples as somehow “given”, and devotes little or no discussion to the process of obtaining the examples. By getting the simulation model to present the human decision maker with realistic conditions and asking for a decision, a set of examples could be obtained at an accelerated speed (assuming the model runs faster than real-time!). In this way the approach acts as an aid to obtaining input data, that is, the process by which a human decision maker works.

With recent advances in computing technology, particularly Object Linking and Embedding (OLE), it should be relatively simple to run a simulation model and an expert system in parallel on the same PC. The next section describes an example of this approach, in which an expert system is first programmed to represent the human decision maker, and is then trained via examples obtained from the simulation.

4 EXAMPLE APPLICATION: ALLOCATING LORRIES AT A LOADING BAY

Using the Witness simulation package a model was developed of a fictional lorry loading bay (Figure 1). Lorries arrive at the lorry park at an average interval of

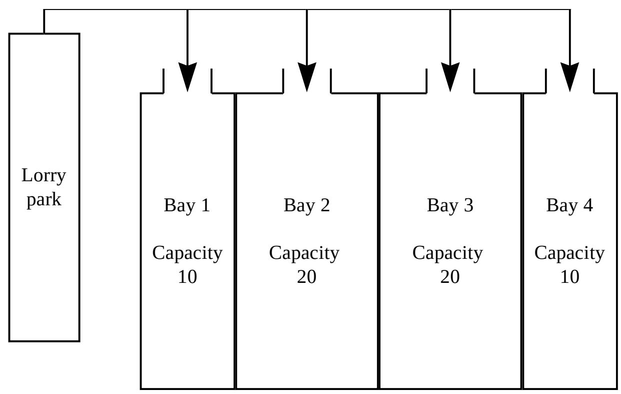

Figure 1: Lorry Loading Bay Example

10 minutes (based on a negative exponential distribution) and require loads of between 5 and 20 items (uniformly distributed). On arrival the lorries are allocated to a loading bay by the bay supervisor, should a suitable one be available. In making this decision the supervisor must take account of the restrictions on the bay capacities. Lorries requiring more than 10 items must be allocated to bay 2 or 3, since bays 1 and 4 only have capacity for up to 10 items. Should a bay not be available then the lorry waits in the park until a suitable bay becomes available. Once a lorry is allocated, it moves to the bay where it is loaded before departing from the system. Lorries take 1 minute to move to the loading bay where each item takes 1 minute to be loaded.

4.1 Linking an Expert System to the Simulation Model

XpertRule was used to develop the expert system that represents the supervisor's allocation decisions. This package was selected for two reasons. First, it adopts a rule induction approach. Consideration was given to both neural network and case based reasoning approaches, but while these may provide benefits in terms of their ability to learn from examples, neither is able to provide information on how decisions are taken. Since, as discussed in the conclusion, this could be an important benefit of using expert systems, a rule induction approach was considered most appropriate. Second, XpertRule is one of the few expert systems packages available that has a true Windows implementation and is OLE compliant.

Since Witness can only work as an OLE slave, it was necessary to develop a model controller (MC) in Visual Basic (Figure 2). The MC initiates the run of the simulation model. At a point where an allocation decision is required, the simulation model automatically stops and waits until the MC returns a decision and continues the run. Once the MC has detected that the model is not running, it extracts data from the model which it passes to the expert system for a decision. The decision is returned to the simulation model via the MC. Some effort was required to ensure that this sequence of events was adhered to. A particular difficulty was encountered in detecting whether the Witness model had stopped running before seeking a decision from XpertRule. If Witness could act as an OLE client it could call XpertRule directly, removing the need for the MC. This would have simplified the linking of the packages significantly.

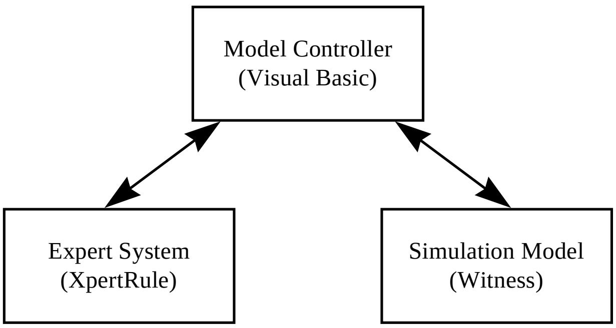

Figure 2: Linking Witness to XpertRule

Having developed the interface between the two packages, the model was used in two ways.

4.2 Mode 1: Developing Decision Rules Directly in XpertRule

One of the authors (Robinson) acted as the expert and was interviewed in order to elicit information on how he would make the allocation decisions. These were then represented in XpertRule as a decision-tree (Figure 3).

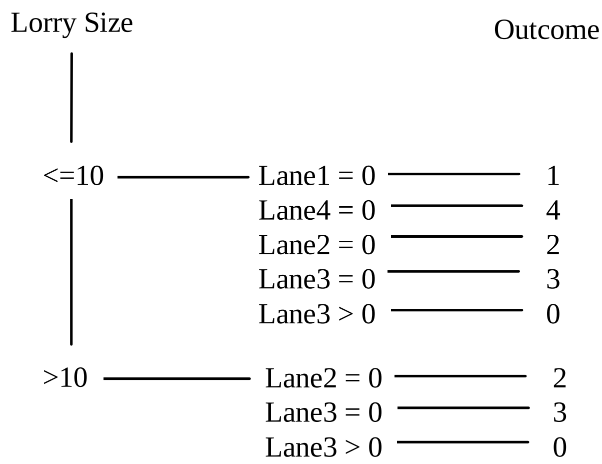

Figure 3: Decision-Tree Developed from Knowledge-Elicitation Exercise

The allocation decision rests primarily on the size of the lorry. If this is less than or equal to 10 items, then the lorry can be allocated to any one of the lanes. An attempt is made to allocate the lorry to the smaller lanes first (lanes 1 and 4). The variables Lane1 - Lane4 are set to 0 if the lane is not allocated, or to the number of the lorry (the first lorry to arrive is numbered 1 etc.) that is currently allocated to that lane. Lorries that require more than 10 items can only be allocated to lanes 2 and 3. The outcome is the number of the lane to which the lorry is to be allocated. If no lane is available, then the outcome is 0.

4.3 Mode 2: Learning Decision Rules from Examples Supplied via the Simulation

The simulation model was run and at a decision point the user was prompted for an allocation decision. These decisions were logged in a data file along with five state variables: the number of items to be loaded on the lorry (Lorry Size) and whether each of the four lanes are already allocated (Lane1 - Lane4). These were then used to train the expert system. With as few as 40 examples it was possible for the expert system to obtain approximately the same decision-tree as derived in mode 1 (Figure 4). A difference occurred because no instances of lorries requiring 10 or less items being allocated to bay 2 or 3, or indeed, not being allocated to a lane, were encountered in the examples; this was considered possible in mode 1.

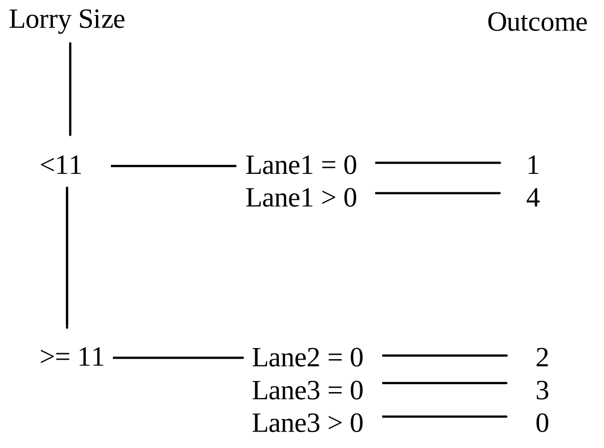

Figure 4: Decision Tree Induced from Examples

5 CONCLUSION

What this simple example demonstrates is that it is technically feasible to link an expert system with a simulation model, on the same PC, to represent human decision making. The approach is particularly likely to reap benefits when used in the second mode described above. Experts may not always be able to clearly define how they go about making complex decisions. However, by presenting them with a set of examples via a simulation, and recording their decisions and some relevant state variables, it may be possible to elicit their decision making process by training an expert system. Indeed, this was possible in the loading bay example.

Having trained the expert system it could be used in three ways:

- to run the simulation model without the need for intervention from the human decision maker
- to train decision makers: either novice decision makers or potentially established decision makers by comparing the approach of different experts
- to operate the real-life facility

For the first of these applications the aim of the expert system is to represent the human decision maker as accurately as possible. This contrasts with the usual aim of an expert system, which is to make the best decisions possible, not to match the standard of the expert precisely.

Following on from this initial work, a number of research questions arise. For a more complex environment, how many examples would be required to train an expert system to a satisfactory level? To what extent should outlier decisions be identified and included in the simulation/expert system? To what extent should the induced rules be edited before attempting to use them in the simulation? Having trained an expert system, how can it be prevented from giving spurious decisions if situations occur in a simulation run, or indeed the real world, that have not been previously encountered? Although a rule induction approach has been adopted here, what advantages and disadvantages are there in this approach over case based reasoning and neural network approaches? The aim of future research will be to investigate these questions, the next stage being to develop a model of a real system.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the financial support given by the British Council's Sino-British Friendship Scholarship Scheme and the Lanner Group in respect of this work.

REFERENCES

Balmer, D. and R .J. Paul. 1986. CASM - The Right Environment for Simulation. *Journal of the Operational Research Society* 37 (No. 5): 443-452.

Deslanders, V. and H. Pierreval. 1991. An Expert System Prototype Assisting the Statistical Validation of Simulation Models. *Simulation* 56 (No. 2): 79-89.

Doukidis, G. I. 1987. An Anthology on the Homology of Simulation with Artificial Intelligence. *Journal of the Operational Research Society* 38 (No. 8): 701-712.

Doukidis, G. I. and M. C. Angelides. 1994. A Framework for Integrating Artificial Intelligence and Simulation. *Artificial Intelligence Review* 8 (No. 1): 55-85.

Doukidis, G. and R. Paul. 1985. Research into Expert Systems to Aid Simulation Model Formulation. *Journal of the Operational Research Society* 36 (No. 4): 319-325.

Flitman, A. M. and R. D. Hurrion. 1987. Linking Discrete-Event Simulation Models with Expert Systems. *Journal of the Operational Research Society* 38 (No. 8): 723-734.

Hart, A. 1987. Role of Induction in Knowledge Elicitation. In *Knowledge Acquisition for Expert Systems: a Practical Handbook*, ed. A. Kidd, 165-189. New York: Plenum.

Hurrion, R D. 1991. Intelligent Visual Interactive Modeling. *European Journal of Operational Research* 54 (No. 3): 349-356.

Hurrion, R. D. 1992. Using a Neural Network to Enhance the Decision Making Quality of a Visual Interactive Simulation Model. *Journal of the Operational Research Society* 43 (No. 4): 333-342.

Hurrion, R. D. 1993a. Representing and Learning Distributions with the Aid of a Neural Network.

Journal of the Operational Research Society 44 (No. 10): 1013-1024.

Hurrion, R. D. 1993b. Using 3D Animation Techniques to Help with the Experimental Design and Analysis Phase of a Visual Interactive Simulation Project. *Journal of the Operational Research Society* 44 (No. 7): 693-700.

Hurrion, R. D. 1997. An Example of Simulation Optimisation Using a Neural Network Metamodel: Finding the Optimum Number of Kanbans in a Manufacturing System. *Journal of the Operational Research Society* 48 (No. 11): 1105-1112.

Lyu, J. and A. Gunasekaran. 1997. An Intelligent Simulation Model to Evaluate Scheduling Strategies in a Steel Company. *International Journal of Systems Science* 28 (No. 6): 611-616.

Mathewson, S. 1989. Simulation Support Environments. In *Computer Modelling for Discrete Simulation*, ed. M. Pidd, 57-100. Chichester, UK: Wiley.

O'Keefe, R. M. 1989. The Role of Artificial Intelligence in Discrete-Event Simulation. In *Artificial Intelligence, Simulation and Modeling* ed. L. E. Widman, K. A. Loparo and N. R. Neilsen, 359-379. New York: Wiley.

Waterman, D. A. 1986. *A Guide to Expert Systems.* Reading, Mass: Addison-Wesley.

Williams, T. 1996. Simulating the Man-in-the-Loop. *OR Insight* 9 (No. 4): 17-21.

AUTHOR BIOGRAPHIES

STEWART ROBINSON lectures in Operational Research and Systems at the Warwick Business School in the UK. He holds a BSc and PhD in Management Science from Lancaster University. Previously employed in simulation consultancy, he supported the use of simulation in companies throughout Europe and the rest of the world. His research interests are in finding ways to improve the use of simulation within industry and he is author of the book *Successful Simulation* (McGraw-Hill), a practical guide to simulation projects. Current work involves an investigation into the use of expert systems to represent a human decision maker in simulation models and developing an understanding of quality in relation to simulation studies.

JOHN EDWARDS, PhD is Senior Lecturer in Operational Research and Systems at Aston Business School. His principal research interests are knowledge-based systems and decision support systems, especially methods for their development. He has written more than 30 research papers on these topics, and two books, Building Knowledge-based Systems and Decision Making with Computers.

WU YONGFA is a lecturer in Management Engineering at the South-East University in Nanjing, P.R. China. His research interests are in decision support systems, systems engineering and computer-based systems in management in general; past research projects have included urban traffic management, and the textile industry. He has written several papers published in Chinese journals, and also a textbook on operational research/management engineering.