Proceedings of the 1994 Winter Simulation Conference
ed. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila

THE CASM ENVIRONMENT REVISITED

Ray J. Paul
Vlatka Hlupic

Department of Computer Science
Brunel University
Uxbridge, Middlesex UB8 3PH, U.K

ABSTRACT

The CASM project started in 1982. Since that time a number of software developments have taken place which have been reported on in the literature. These developments reflect the research group's views on one of the ways in which simulation modelling could be conducted in practice. This paper re-examines these views with respect to the underlying methodology of simulation modelling; discusses the latest version of the simulation software developed by members of the CASM team; and mentions the method of organisation of the CASM research team. Some comments on the success or otherwise of these endeavours are made and on the future anticipated research endeavours of the group.

1 INTRODUCTION

The Computer Aided Simulation Modelling project (CASM) was initiated in 1982, and its fundamental approach to simulation annunciated by Balmer and Paul (1986). This research is continuing at Brunel University within the recently instituted Centre for Applied Simulation Modelling. Paul (1992b) reported on the work of CASM up until 1992, and this paper revises that work. The research group's objectives were based on the teaching and consultancy experience of the project's directors. Research work is undertaken by a continuous stream of bright would-be Ph.D students, who undertake specific parts of the research programme. Twelve students have already succeeded in obtaining their Ph.Ds (Doukidis 1985, Chew 1986, El Sheikh 1987, Knox 1988, Mashhour 1989, Au 1990, Domingo 1991, Angelides 1992, Barakat 1992, Mejia 1992, Mak 1993 and Hlupic 1993) and two students are likely to complete in 1994. There are ten research students currently working on the project. The research group also includes several academics from institutions in various countries such as the U.S.A, Croatia, Slovenia, Brasil, Greece and Hong Kong. Many papers have been published in the research literature and these and prospective papers are listed in a fairly complete listing of all the CASM papers and theses since Paul (1992b).

The CASM research group concentrates on the problems related to discrete event computer-based simulation modelling. This area of modelling is particularly popular amongst the operational research and information systems fraternities. Whilst continuous modelling, differential equations, systems/industrial dynamics and other temporal modelling systems are undoubtedly of interest, and are related to discrete event modelling, at the current time CASM is restricting its research interests in order to make progress in one of these dimensions.

In the next section, the process of simulation modelling is reiterated and the problems associated with simulation modelling as seen by the authors are outlined. Following this, the objectives and underlying methodology of the CASM research group are described. Some of the modelling environments that have been developed are described in the following section. Application areas are covered next. The paper concludes with the experiences and future anticipated research of members of the research group.

2 SIMULATION MODELLING PROCESS

Paul (1992b) described CASM's view of the simulation process. To summarize, the problems associated with using simulation modelling as a decision aiding technique are as follows. First of all, most problems to which one applies simulation are poorly defined. In fact one might go further, and claim that if the problem is not poorly defined, there are probably better and more reliable methods of solving the problem than the rather crude technique of simulation modelling. Secondly, any problem of any complexity which is important will probably involve conflicting interests and understanding. One must anticipate that if the modelling process is going to lead to change in the organisation, then it is unlikely that all decision makers will see these changes as favourable to them. The analyst must anticipate negative attitudes and spoiling tactics. As much as possible, the modelling process is used in a neutral way to help the participants in the decision making process understand their problem, and come to a resolution amongst themselves. The third problem associated with simulation modelling is that there never exists a static specification of the problem, it is always dynamic. Even if one succeeds in satisfying the conflicting views of the decision makers, it is probable that for complex problems the specification still undergoes change. The real world is dynamic and therefore the perceived problem will be dynamic as well. The fourth problem with simulation modelling is the question of `model confidence', which is better terminology than the commonly used description of verification and validation. No computer program of any size can possibly be verified. No model of any size can possibly be validated against the real world, especially given that the real world is not static. The model cannot be proved to be correct. The aim should be to use methods that demonstrate confidence in what the model is doing and the way it is doing it.

Discrete event simulation modelling is a quantitative technique. The outputs are numerical, and numerical values tend to indicate that one course of action might be better than another. However, such a numerical technique cannot represent all possible factors in the problem scenario. It can crudely represent most or some of them in a quantitative way, but it cannot represent subjective factors. It must be remembered that the simulation modelling process is not designed to find the answer or answers. It is there to help decision makers take decisions, or to help decision makers gain an understanding of their problem. The numerical output of the simulation model in itself may often be of no particular intrinsic value. Learning about the processes of the interactions that go on within a complex environment, the relationships between the variables, is probably the dominating characteristic of interest in simulation modelling.

3 CASM OBJECTIVES

CASM is researching into simulation modelling, with a view to producing computer systems that automate as much as possible, the simulation modelling process. The aim is to make simulation efficient as a modelling tool for helping decision makers understand their problems. It is impossible to produce an all purpose simulation modelling system that can handle any problem that one might wish to model. The analyst is restricted to what a simulation system can handle, or the simulation system must provide programming code that can be modified to do the task that has been set. In this latter context, CASM are dedicated to the production of transparent models (i.e. program code that can be read by somebody else). Gifted amateurs not only produce program code that cannot be read by other programmers, but after a short lapse of time, cannot even be read by themselves! It is therefore quite apparent that a highly stylised, highly structured method of writing computer simulation models is required, so that anyone familiar with this structuring and style is able to read and understand it.

CASM is a research group operating within a university environment, so the computer systems that are researched into must also help in the teaching of simulation modelling, as well as assisting in further research into simulation modelling. Other apparently relatively insignificant factors need to be taken into account. A variety of career paths for the research participants must be satisfied. If this were not so, then individuals would feel free to go in any direction that appeared to satisfy their goals. Lastly, but not least, in a research environment it is important that the individuals concerned enjoy what they are doing. If the researchers do not enjoy their work, a variety of reasons will be found for why things are not working, not being done, or not happening.

4 THE CASM APPROACH TO ENVIRONMENTS

Figure 1 illustrates the sort of simulation environment that CASM envisages would assist the analyst assist the decision maker. This environment is more extensively described by Balmer and Paul (1986). The analyst and customer, or decision maker, would use a system that assisted in problem formulation. This problem formulation system would essentially capture the model logic of the problem to which could be applied an interactive simulation program generator (ISPG). The ISPG would produce a simulation model which called on a library of software subsystems to actually run the simulation itself. Simulation model output would be analysed by an output analyser which would, again under analyst control, help determine experimental design for running and controlling the simulation model. It is anticipated that the problem formulator and output analyser would close the loop, so that the analyst and decision maker could collectively use the complete system.

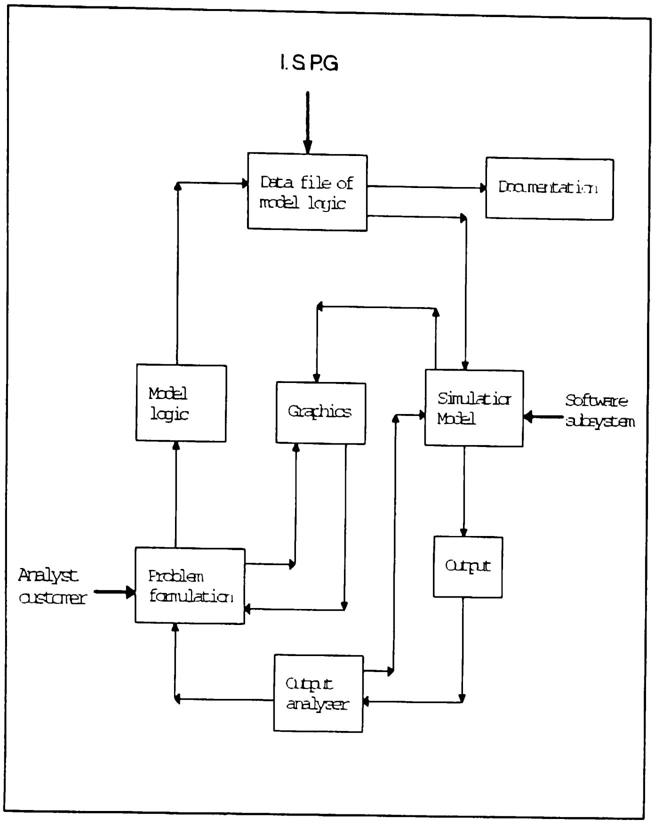

Figure 1: CASM Modelling Environments

Attempts have been made by CASM to develop a problem formulator. These attempts are described by Doukidis (1985,1987), Doukidis and Paul (1985,1986,1987b), Paul & Doukidis (1986), and Paul (1987). The final system, which works, is a natural language understanding system. Using an activity cycle diagram paradigm, the natural language understanding system asks questions of the analyst at a computer terminal. This system has been demonstrated to work, in that the model logic for a problem can be determined in this way. Curiously enough, however, the system is not used in practice. The reason is that the purpose of the system, to help the analyst formulate a problem in conjunction with the decision makers, is not practically feasible in this way. One cannot expect a decision maker, or decision makers, to sit in front of a screen, talking sideways to an analyst, who is being controlled, in a textual sense, by the computer. This is a completely unnatural way for humans to hold a discourse. The terminology that the natural language understanding system uses is also unnatural for the decision maker, albeit well understood by the analyst.

The latest attempts at applying intelligent systems to problem formulation are described by Abdurahimin and Paul (1994). A system is being developed using inductive learning where the problem is hypothesised using positive and negative examples of aspects of the problem. The resultant formulation is logically correct, although there is no guarantee concerning its validity.

The interactive simulation program generator (ISPG) part of the environment has been researched into throughout the duration of the CASM research project. First attempts emulated the work of Clementson's CAPS/ECSL package (1982). The second version, called AUTOSIM, made some minor improvements (Paul & Chew 1987). A later version, VS6, is described by Knox (1988). All these interactive simulation program generators generate program code in a high level programming language, in this case Pascal. A high level programming language was deliberately selected because of the availability of expertise and assistance on a broad level. Many simulation systems develop simulation code in their own purpose built language. These languages have undoubtedly been developed to a high degree of sophistication suitable for simulation modelling purposes. However, they require participants in the modelling process to learn the language in order to use it. CASM has concentrated on using a high level programming language to avoid the problem of scarcity of experts.

In order to handle the problem of simulation specific code, CASM have produced well written modifiable libraries of simulation routines. These routines enable the commonplace parts of any simulation model to be easily accessed. The generated program code for any particular problem is written in a three phase structure, as described by Crooks et al (1986) and Paul and Balmer (1993). This structure has the virtue of describing the control of flow in any model accurately and is easy to modify. Bearing in mind the earlier points made about the need for dynamic model development, it is clear that this ease of modification is an essential characteristic of any generated simulation program. Apart from the virtue of writing simulation models in a highly stylised structured way, an interactive simulation program generator also has the advantage that it produces models in which one may have a high degree of confidence. As the ISPG is applied to more and more problems, the errors in the system itself are slowly removed, so that the generated code is more likely to be correct. Another virtue of using an ISPG is the ability to undertake rapid prototyping. This means that if the specification of the problem changes, as it almost invariably does, then the ISPG can be reapplied to write a new program each time.

In theory, if the description of the model logic is adequate enough for an ISPG to produce the simulation code, then it must be adequate enough for documentation of the simulation model to also be automatically produced. Whilst this is almost self evident, it seems that the task of producing a system that automatically documents programs is not quite as exciting for prospective researchers as many of the other tasks available within the research project!

The last part of Figure 1, the output analyser, presupposes that the large body of published statistical knowledge describing how to analyse output from simulation models can be encoded into some sort of intelligent system. Regrettably, however, it turns out that the statistical knowledge available, largely tested on simple simulation models, does not appear to work so readily on the complex sort of models that one generally applies simulation to. Therefore, research in this area is into simple ways of analysing the output from simulation models. Some early results that reinforce this approach are emerging in work undertaken by Mejia (1992).

5 MODELLING ENVIRONMENTS

5.1 Specification Methods

Surprisingly, given the relative length of time that simulation modelling has been undertaken on computers, there is no fixed method for specifying simulation problems. There are a large number of diagrammatic techniques such as activity cycle diagrams and Petri nets, and semi formal methods as exemplified by Zeigler's work (1984). The basic problem in specification appears to be as follows. If specification is going to be used as a vehicle for communication, it must have a simple structure. However, many simulation models inherently model complex situations, and the combination of objects or entities in an activity requires some complex conditions to be stated. If these conditions are described explicitly in the specification method, then the specification becomes very difficult to follow.

At one extreme are diagramming methods, which give a very simple representation of the basic simulation model structure. At the other extreme are formal methods or mathematical approaches, which make everything explicit but suffer from a heavy use of mathematics, and which is therefore not understandable by very many people. Ceric and Paul (1989) describe a brief survey of available diagrammatic methods that are commonly used in simulation modelling. In a later paper the principle of Comprehensive Harmony is expounded for the requirements of a specification method. This principle quite simply requires that the specification method must be reasonably comprehensive. However, this comprehensiveness must be balanced by a harmony in the method of specification that makes it intelligible to the active participants in the simulation modelling process. It is anticipated that such comprehensive harmony might be provided by a mixture of diagrammatic methods with a hierarchy of descriptions leading to formal methods at the lowest level.

Mak (1993) has have shown that the activity cycle diagram specification of a discrete event simulation model can be translated into a Systems Dynamics diagram. There are certain extra pieces of information required, and the translation can require intelligent transformation. Curiously, the reverse translation proved to be very difficult, since the Systems Dynamics model needs much less structure than the discrete event simulation.

5.2 Graphics Driven Environments

Since the inception of CASM, the creation of a simulation environment has been one of the main objectives of the research group. Chew (1986) produced the first of CASM's interactive simulation program generators (ISPGs) which form the basis for a three phase simulation system written in Pascal. Later work on graphics, adding a picture to the simulation modelling process, is described by Knox (1988).

The latest development in these environments is described by Au (1990) and Au and Paul (1993). This graphics driven environment allows the users, the analyst and the customer, to specify the problem using iconic representations for the objects in the system. The icons are laid out on the screen in a logical fashion, intelligible to the user as well as to the analyst. No particular formulism is used for this, in terms of diagrams or methods, although underpinning the method is the activity cycle diagram concept. This system was developed on the Macintosh microcomputer, which is an ideal environment for mixing graphics display with text. The system provides the user with assistance in the construction of the logic of the problem, and in the addition of quantitative and conditional information to the model logic.

Future work in this area is intended to remedy some of the possible deficiencies in complexity of problem that can be handled by this system. This might be achieved using a mixture of graphics and artificial intelligence techniques. Further enhancements might in any case be provided by producing a richer mixture of interrelated screens for the analyst to specify the problem with, plus some better help facilities for reminding the user of what is required for a complete specification. Hopefully, in the not too distant future, one might build such an environment and incorporate the benefits of the research in formal methods described above.

6 APPLICATION AREAS

The CASM research group are constantly aware that their research endeavour into simulation modelling needs to be related to the real world if the research is not to become esoteric. There are a number of application areas that we are looking at.

Kuljis et al (1990) are examining an out-patients clinic model with respect to commercial application packages. This out-patients clinic has been built so that OR analysts in the health service can go to the administrators and doctors who operate, or who are responsible for, such clinics, to demonstrate the feasibility of different clinic practices. A common problem with clinics is the waiting time of the patients at various stages throughout the process, and it is hoped by showing a visual simulation representation of these clinics for different clinic operating practices, that the people concerned might be persuaded to operate practices that are more beneficial to the patients and at no loss to them. The system is currently undergoing trials.

Lehaney and Paul (1994) describe work that is being conducted on an outpatient clinic in the U.K. Here the purpose of the exercise is to use Soft Systems Methodology to understand the problem sufficiently to build a simulation model. The soft modelling approach added to the hard simulation language is proving both effective and acceptable to the 'customer'.

Hlupic and Paul (1992, 1994b) describe work being carried out in the area of Flexible Manufacturing Systems, which includes an extensive case study with a manufacturing company (Hlupic and Paul, 1993c-e, 1994a).

Our early CASM systems, described by Crookes et al (1986), have been used in a number of military applications. These are described by Holder and Gittins (1989), Williams et al (1989), and Stapley and Holder (1992). The interesting characteristic of the use of the simulation systems by these groups is that they partially replaced previous systems quite successfully and very effectively. The models described in the first two papers were eventually joined together in a reasonably short space of time. It is pleasing that the claimed flexibility and effectiveness of these systems has actually been demonstrated in a real application, and the systems are continuing to be of use (Ceric and Hlupic, 1993), (Stapley and Holder, 1992).

7 OTHER CASM RESEARCH AREAS

There are several other research areas covered by the members of the CASM research group. For example, Hlupic (1993) researches into simulation modelling software approaches to manufacturing problems. The major part of this work relates to simulation software evaluation criteria and software selection methodology (Hlupic and Paul, 1993a-b).

Angelides and Paul (1993) research into combining simulation games and intelligent tutoring systems. Hirata and Paul (1994) examine an object-oriented programming architectures for simulation modelling. Mladenic et al (1993, 1994) research into using machine learning techniques to interpret results from discrete event simulation. Paul and Chanev (1994) have investigated the potential of genetic algorithms in simulation model optimisation.

Some members of CASM are researching into object-oriented program generators for simulation modelling. As a part of this research, a Model Description Language is proposed for the specification of models based on the process interaction approach, and is used inside a program generator currently under development (Kienbaum and Paul, 1994).

Future CASM research will be into the areas of graphical problem formulation to drive the software systems that automate the simulation process. Research into expert system development for simulation software evaluation has been initiated. Some work has started on determining the relationships between discrete event simulation modelling and more general forms of modelling of systems over time, such as systems dynamics (Mak, 1993), control theory, differential equations and queueing theory.

8 EXPERIENCES AND CONCLUSIONS

The CASM simulation systems have been tested on many groups of students over the last twelve years. It is good experience for students who are going to work in Operational Research to use systems that are not fully tested. It teaches them to be more than a little wary of software! One of the features of the systems developed is the concentration on activity cycle diagrams. However, activity cycle diagrams are not all-embracing. It is very easy to construct examples of problems where the logic of the problem is not captured in the activity cycle diagram. For example, in the port problem described by

El Sheikh et al (1987), the activity cycle diagram is very simple. It has two small cycles and only two activities, but the logic in the model is very complex. The rules for engagement of ships and berths require a matching between the ship cargo, the handling facilities of the berth, the priorities that various ships have on different berths, and so on. These priority rules cannot be visually displayed on an activity cycle diagram, but they are an essential component of this particular simulation problem.

In conclusion, CASM believe that their research approach will lead to a concentration on the more difficult tasks of simulation modelling. These are problem definition and understanding, improving model confidence, experimental design and 'implementation'. These are the intellectual tasks facing analysts in helping the decision maker. They are often not given the effort they require because of the time taken in the more mundane programming elements of the simulation model. If the analyst can concentrate, with the assistance of efficient low cost software support, on these more difficult intellectual tasks then the analyst will then be able to work more closely with the decision maker. There is no doubt that collaboration between analyst and decision makers in decision aiding is synonymous with success, however one defines success.

REFERENCES

Ceric V. and V. Hlupic. 1993. Modelling a Solid-Waste Processing System by Discrete Event Simulation. *Journal of of the Operational Research Society*. 44(2): 107-114.

Clementson, A.T. 1982. *Extended Control and Simulation Language*. Birmingham: Cle.Com Ltd.

Holder, R.D. and R.P. Gittins. 1989. The Effects of Warship and Replenishment Ship Attrition on War Arsenal Requirements. *Journal of the Operational Research Society*. 40: 167-175.

Stapley N.R. and Holder R.D. 1992. The Development of an Amphibious Landing Model. *J.N.S.* 18(3): 193-202.

Williams, T.M.; R.P. Gittins and D.M. Burke. 1989. Replenishment at Sea. *Journal of the Operational Research Society*. 40: 881-887.

Zeigler, B.P. 1984. *Multifacetted Modelling and Discrete Event Modelling*. Academic Press.

CASM REFERENCES AND SOME NEW CASM REFERENCES SINCE WSC'92 PAPER

Abdurahiman V. and R. J.Paul. 1994. Machine Learning and Simulation Model Specification. Accepted by *Journal of Simulation Practice and Theory*.

Angelides M. 1992. Developing the Didactic Operations for Intelligent Tutoring Systems: A Synthesis of Artificial Intelligence and Hypertext. Unpublished Ph.D thesis. University of London, England.

Angelides M. C.and R. J.Paul. 1993. Developing an Intelligent Tutoring System for a Business Simulation Game. Journal of Simulation Practice and Theory, 1(3):109-135.

Au, G. 1990. A Graphics Driven Approach To Discrete Event Simulation. Unpublished Ph.D thesis. University of London, England.

Au G. and R. J. Paul. 1993. Computer Simulation Modelling Using Hypercard. Journal of Computing and Information Technology, 1(1): 1-13.

Balmer, D. W. and R. J. Paul. 1986. CASM- The Right Environment for Simulation. Journal of the Operational Research Society. 37(5): 443-452.

Barakat, M. 1992. Semantic Modelling for Discrete Event Simulation. Unpublished Ph.D thesis. University of London, England.

Ceric V. and R. J. Paul. 1989. Preliminary Investigations into Simulation Model Representation, in the Proceedings of the 11th International Symposium on "Computer at the University" (Cavtat, Yugoslavia, June 1989).

Ceric V. and R. J. Paul. 1992. Diagrammatic Representations of the Conceptual Simulation Model for Discrete Event Systems. Mathematics and Computers in Simulation, 34(3-4): 317-324.

Chew, S. T. 1986. Program Generators For Discrete Event Digital Simulation Modelling. Unpublished Ph.D thesis. University of London, England.

Crookes, J. G.; D. W. Balmer; S. T. Chew; and R. J. Paul. 1986. A Three-phase Simulation System Written in Pascal. Journal of the Operational Research Society. 37(6): 603-618.

Domingo, L. T. 1991. Formal Methods in Specifying Discrete Event Simulation Models. Unpublished Ph.D thesis. University of London, England.

Doukidis G. I. 1985. Discrete Event Simulation Model Formulation Using Natural Language Understanding Systems. Unpublished Ph.D thesis. University of London, England.

Doukidis G. I. 1987. An anthology on the homology of simulation with artificial intelligence. Journal of the Operational Research Society 38(8): 701-712.

Doukidis G. I. and R. J. Paul. 1985. Research into Expert Systems to Aid Simulation Model Formulation. Journal of the Operational Research Society. 36(4): 319-325.

Doukidis G. I.; and R. J. Paul. 1986. Experiences in Automating the Formulation of Discrete Event Simulation Models. In AI Applied to Simulation, (E. J. H. Kerckhoffs, G. C. Vansteenkiste and B. P. Zeigler (Eds.)) Simulation Series. 18(1): 79-90. The Society for Computer Simulation, San Diego, USA.

Doukidis G. I.; and R. J. Paul. 1987a. ASPES: A Skeletal Pascal Expert System. in Expert Systems and Artificial Intelligence in Decision Support Systems (H. G. Sol et al., Eds.) D. Reidel, Dordrecht, Holland. pp.227-246.

Doukidis G. I.; and R. J. Paul. 1987b. Artificial Intelligence Aids in Discrete Event Digital Simulation Modelling. IEE Proceedings, 134(4): 278-286.

Doukidis G. I. and R. J. Paul. 1991. SIPDES: A Simulation Program Debugger Using an Expert System. Expert Systems With Applications, 2(2/3):.153-165.

Doukidis G. I. and R. J. Paul (eds.) 1992. Artificial Intelligence in Operational Research. Macmillan. ISBN 0-333-55117-6

El Sheikh A. A. R. 1987. Simulation Modelling Using A Relational Database Package. Unpublished Ph.D thesis. University of London, England.

El Sheikh, A. A. R, R. J. Paul, A. S. Harding, and D. W. Balmer. 1987. A Microcomputer Based Simulation Study of a Port. Journal of the Operational Research Society 37(8): 673- 681.

Hlupic V. 1993. Simulation Modelling Software Approaches to Manufacturing Problems. Unpublished PhD Thesis, University of London, London, England.

Hlupic, V. and R. J. Paul. 1992. FMS Scheduling Strategies using the Simulation Package SIMFACTORY II.5. In the Proceedings of the 8th International Conference on CAD/CAM, Robotics, and Factories of the Future. (Place Edouard Branly, Metz, France, 17-19 August)

Hlupic V. and R. J. Paul. 1993a. Simulation Software in Manufacturing Environments: A Users' Survey. Journal of Computing and Information Technology. 1(3): 205-212.

Hlupic V. and R. J. Paul. 1993b. Selecting Software for Manufacturing Simulation. In the Proceedings of the XV International Conference on Information Technology Interfaces (15-18 June, Pula, Croatia). University of Zagreb Computer Centre, Zagreb, Croatia. pp. 387-394.

Hlupic V. and R. J. Paul. 1993c. A Critical Evaluation of Modelling an Automated Manufacturing Systems using the Simulation Package WITNESS. In the Proceedings of the 9th International Conference on CAD/CAM, Robotics, and Factories of the Future. (18-20 August, Newark N.J.)

Hlupic V. and R. J. Paul. 1993d. Simulation Modelling of an Automated System for Electrostatic Powder Coating of Metal Components. In the Proceedings of the 1993 Winter Simulation Conference (G. W. Evans, M. Mollaghasemi, E. C. Russell and W. E Biles, Eds.) (12-15 December 1993, Los Angeles). Association for Computing Machinery, New York, pp 1324-1329.

Hlupic V. and R. J. Paul. 1994a. Simulating an Automated Paint Shop in the Electronics Industry. Journal of Simulation Practice and Theory. 1(4):195-205.

Hlupic V. and R. J. Paul. 1994b. Simulation Modelling of Flexible Manufacturing Systems using Activity Cycle Diagrams. Journal of the Operational Research Society. 45(9): 1011-1023

Hirata C. M. and R. J. Paul. 1994. An Object-Oriented Programming Architecture for Simulation Modelling. accepted by International Journal of Computer Simulation.

Kienbaum G. and Paul R. J. 1994. H-ACD: Hierarchical Activity Cycle Diagrams for Object-Oriented Simulation Modelling. Accepted by the 1994 Winter Simulation Conference (Orlando, Florida).

Knox, P. M. 1988. Automated Graphically-Based Discrete-Event Simulation Systems. Unpublished Ph.D thesis. University of London, England.

Kuljis, J. and R. J. Paul. 1991. Human-Computer Interfaces for Modelling Systems. In the Proceedings of the XIII International Conference on Information Technology Interfaces (Cavtat, Yugoslavia, June).

Kuljis, J.; R. J. Paul, H. Malin and S. Thakar. 1990. Designing an Out-patient Clinic Modelling Package. In the Proceedings of the 12th International Symposium on "Computer at the University" (Cavtat, Yugoslavia, June).

Lehaney B. and Paul R. J. 1994. Using Soft Systems Methodology to Develop a Simulation of Out-patient Services. *Journal of the Royal Society for Health.* August, 197-200.

Mak H-Y. 1993. Systems Dynamics and Discrete Event Simulation Modelling. Unpublished PhD Thesis, University of London, London, England.

Mashhour, A. 1989. Automated Simulation Program Generation using a Relational Database Simulation System. Unpublished Ph.D thesis. University of London, England.

Mejia, A. 1992. Output Analysis in Discrete Event Modelling. Unpublished Ph.D thesis. University of London, England.

Mladenic D., I. Bratko, R. J. Paul and M. Grobelnik. 1993. Using Machine Learning Techniques to Interpret Results from Discrete Event Simulation. In the Proceedings of the XV International Conference on Information Technology Interfaces (15-18 June, Pula, Croatia). University of Zagreb Computer Centre, Zagreb, Croatia. pp. 401-406.

Mladenic D., I. Bratko, R. J. Paul and M. Grobelnik. 1994. Using Machine Learning Techniques to Interpret Results from Discrete Event Simulation. In the Proceedings of the European Conference on Machine Learning. (April 1994, Catania, Sicily).

Paul, R. J. 1987. A.I. and stochastic process simulation. In Interactions in Artificial Intelligence and Statistical Methods, (B. Phelps Ed.) Gower Technical Press: 85-98.

Paul, R. J. 1989a. Artificial Intelligence and Simulation Modelling in Computer Modelling for Discrete Simulation (Pidd, M. Ed), Wiley, London.

Paul, R. J. 1989b. Combining Artificial Intelligence and Simulation in Computer Modelling for Discrete Simulation (Pidd, M. Ed), Wiley, London.

Paul, R. J. 1989c. Use of Simulation to Investigate Stock Control Policies. In the Proceedings of the Technical Overview Symposium on Management Control Systems and Information Technology Implementation Issues of Inventory and Stock Control Systems. (London, June).

Paul, R. J. 1991. Recent Developments in Simulation Modelling. Journal of the Operational Research Society. 42(3):.217-226.

Paul R. J. 1992a. Outpatient Clinic Waiting Times: A Visual Simulation Approach. In the Proceedings of the Second Hong Kong (Asia Pacific) Medical Informatics Conference. (Hong Kong Polytechnic, 27-29 November).

Paul R. J. 1992b. The Computer Aided Simulation Modeling Environment : An Overview. In the Proceedings of the 1992 Winter Simulation Conference (J.J. Swain and D. Gainsman, Eds.). (13-16 December 1992, Arlington, Virginia). Association for Computing Machinery, New York.. pp. 737-746.

Paul R. J. 1993a. AI and Simulation. In AI and Computer Power : The Impact on Statistics (D. Hand, Ed.) Chapman Hall, London.

Paul R. J. 1993b. Activity Cycle Diagrams and The Three Phase Method. In the Proceedings of the 1993 Winter Simulation Conference (G.W. Evans, M. Mollaghasemi, E.C. Russell and W.E Biles, Eds.) (12-15 December 1993, Los Angeles). Association for Computing Machinery, New York. pp 123-131.

Paul, R. J. and D. W. Balmer. 1993. Simulation Modelling. Lund, Sweden: Chartwell-Bratt Student-Text Series.

Paul, R. J. and V. Ceric. 1990. Conceptual Modelling in Discrete Event Simulation using Diagrammatic Representations. In the Proceedings of the IMACS European Simulation Meeting on Problem Solving by Simulation (Esztergom, Hungary, August).

Paul R. J. and T. S Chanev. 1994. Optimising Simulation Models using a Genetic Algorithm. Under revision for the *Journal of Simulation Practice and Theory*.

Paul, R. J. and S. T. Chew. 1987. Simulation Modelling using an Interactive Simulation Program Generator. Journal of the Operational Research Society 38(8): 735-752.

Paul, R. J. and G. I. Doukidis. 1986. Further Developments in the Use of Artificial Intelligence Techniques which Formulate Simulation Problems. Journal of the Operational Research Society 37(8): 787-810.

Paul R. J. and V. Hlupic 1994. Designing and Managing a Masters Degree Course in Simulation Modeling. Accepted for the 1994 Winter Simulation Conference (December, Orlando).

Saliby E. and R. J. Paul. 1993. Implementing Descriptive Sampling in Three Phase Discrete Event Simulation Models. Journal of the Operational Research Society. 44(2):.147-160.

AUTHORS BIOGRAPHIES

See the biographies in Paul R. J. and V. Hlupic 1994. Designing and Managing a Masters Degree Course in Simulation Modeling in this conference proceedings.