A FRAMEWORK FOR BUSINESS PROCESS SIMULATION: THE GRAB AND GLUE APPROACH

Tillal Eldabi
Man Wai Lee
Ray J. Paul
Centre for Applied Simulation Modelling
Department of Information Systems and Computing
Brunel University
Uxbridge, Middx UB8 3PH, U. K.

KEYWORDS
Discrete event simulation modelling framework, grab and glue simulation

ABSTRACT

Simulation modelling is a powerful tool for problem understanding and problem solving. Constructing simulation models following the classical simulation modelling framework has disadvantage of being time consuming, hence making it expensive. Users can sometimes be reluctant to use simulation due to these reasons or implement simulation results. This paper proposes a new simulation approach that tackles the problem of time. For this purpose, this paper will start by reviewing a number of existing simulation modelling frameworks. From this analysis, we attempt to develop a simulation framework that deals with the question of time. The proposed simulation framework is supposed to enhance simulation results and reduce disadvantages related to cost and time.

INTRODUCTION

Simulation modelling is defined as "an analysis and planning tool that captures real-world system variability and subsystem event interactions through time" (SDI 2001). Because of its ability to explore "What-If" questions, it is highly used as a predicting tool in forecasting the performance of the new systems under different sets of circumstances, or as a designing tool for analysing systems (Pidd 1998; Banks et al. 2001).

There is a number of simulation modelling frameworks which have already been developed by various researchers, for example Shannon (1998), Law and Kelton (2000), and Banks et al. (2001). The common problem when applying these frameworks is that they are time consuming. In this paper, we aim to find out why this problem exists and then attempt to tackle it.

This paper is structured as follows. In Section 2, we review the classical simulation modelling frameworks by Shannon (1998), Law and Kelton (2000), and Banks et al. (2001), shown in Figures 1, 2, and 3 respectively. Problems that may exist when applying these frameworks are discussed in Section 3. In Section 4, the requirements to tackle the problem are presented, an alternative proposed framework is presented in Section 5. Section 6 is a discussion about the envisaged advantages and disadvantages of the proposed framework. Finally, Section 7 reports the conclusions.

SIMULATION MODELLING FRAMEWORKS

Shannon (1998) stated that the process of constructing a simulation model should be problem definition; project planning; system definition; conceptual model formulation; preliminary experimental design; input data preparation; model translation; model verification and validation; final experimental design; experimentation; analysis and interpretation; and implementation and documentation (see Figure 1). Law and Kelton (2000) mentioned that a simulation model should be started by formulating the problem and planning the study; collecting data and defining the model. After that, the validity of the conceptual model will be checked. The data collection process will be restarted if there is any problem, and a new conceptual model will be defined again. If the conceptual model can pass the validity test, it will then be translated into a computer program. A pilot run will be executed, and the validity of the program model will be checked. If the testing of this model fails, the process of data collection will be restarted, otherwise a series of experiments will be designed and production runs will be executed. The output of the model will then be analysed, and finally, the model and the outputs will be presented (see Figure 2). Banks et al. (2001) mentioned that the steps followed in a simulation process are problem formulation; setting objectives and the overall project plan; constructing a conceptual model and checking the validity, and data collection at the same time. After that the model transformation and verification of that model can be tested. If the model fails at this step, the transformation of the model needs to be restarted, otherwise, the validation of the model will be checked. If it fails, the modeller needs to go back to the step for constructing the conceptual model. If the test is successful, the experimental design will be completed. Production runs and analysis can then be executed, documentation and reports can be written, and finally, the model can be implemented (see Figure 3).

Proceedings 15th European Simulation Symposium
Alexander Verbraeck, Vlatka Hlupic (Eds.)

ISBN 3-936150-28-1 (book) / 3-936150-29-X (CD)

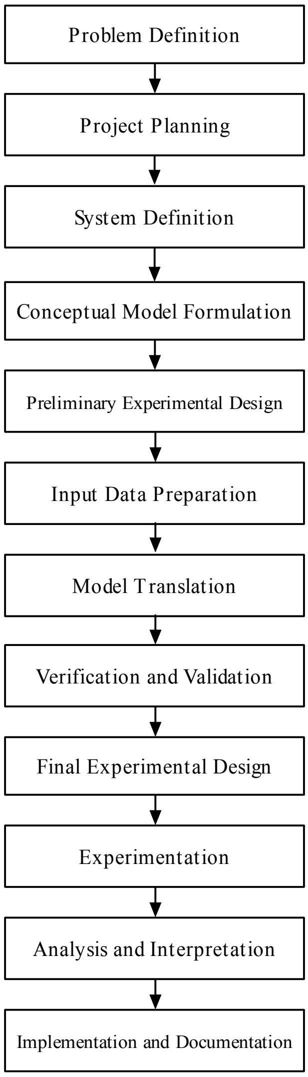

Figure 1: Simulation Modelling Framework (by Shannon, 1998)

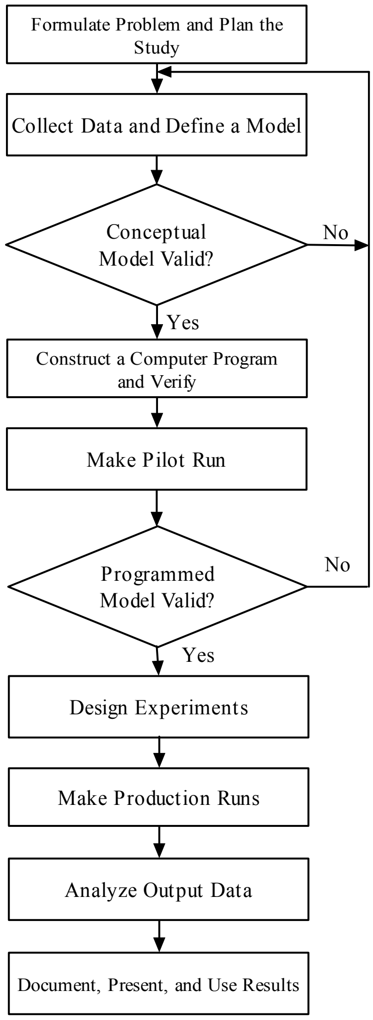

Figure 2: Simulation Modelling Framework (by Law and Kelton, 2000)

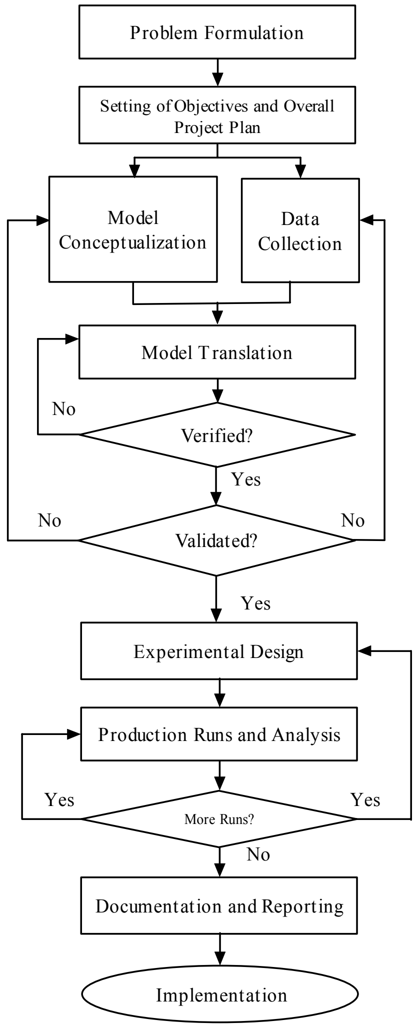

Figure 3: Simulation Modelling Framework (by Banks et al., 2001)

By comparing these three frameworks, it is obvious that the first step to build a simulation model is to formulate the problem. Formulating the problem is the most important step in the simulation modelling process (Eldabi 2000). If the user wants to find out the solution of a problem, he/she must know what the problem is. Without an understanding of the problem, the constructed model may not have the ability to solve the real problem that the problem owner has. After formulating the problem, all three researchers agreed that it is necessary to develop a project plan to ensure that the resources and support are enough to construct the model. Moreover, Shannon (1998) mentioned that it is necessary to determine the boundaries and restrictions of the system or progress as well as to investigate how it works. After that, a conceptual model and to validate that model is required. A conceptual model can be constructed either by graphics or pseudo code. Shannon (1998) mentioned that data can be collected after deciding the required type and the required amount of data. However, Law and Kelton (2000) stated that data collection should be done immediately after the problem formulation process.

After collecting the data, all three frameworks agree that the simulation model can be built based on the conceptual model. Verification of the model is required after translating the model into computer codes or computer graphics. A pilot run of the model is necessary to validate the model. Finally, after validation, it is necessary to make the experiment design, product run and analysis, documentation and report, and the implementation of the model. Experimental design is used for making production runs. Depending on the system configuration of interest, experiment design can be the length of each run, the length of the time to warm up the model, or the number of independent simulation runs using different random numbers. The objective of analysing the output results is either to determine the absolute value of the system configuration or to compare different system configurations under the relative condition. Finally, documentation or reports need to be provided so that the user can understand more about the model (Law and Kelton 2000).

In this section, three different simulation modelling frameworks have been studied, and the similarity and differences were investigated. In the next section, the disadvantage of applying these frameworks to construct a simulation model will be discussed.

SOME SIMULATION PROBLEMS

A successful simulation model can help the problem owner to understand their problem. Applying the framework mentioned in the previous section can help to build the required model. However, constructing this model successfully is very time consuming and, hence, expensive, especially in the process of data collection and model analysis (Pidd 1998). One of the reasons is that data collected from a single person or document is insufficient for our complex world. Another reason is that people may provide inaccurate information. These two reasons show that data collection takes a long time and is hence high cost (Law and Kelton 2000).

Looking back to Figures 1, 2, and 3, Shannon (1998) stated that the process of data collection can be started after deciding the required type of data and the required amount of data, while Law and Kelton (2000) stated that this process should be done immediately after the problem formulation process. However, regardless of when we start the process of data collection, there is a possibility that the collected data will not be suitable for developing the model.

It could be argued that problem owners can discuss their problem with an expert before collecting the data. However, it is still difficult to guarantee that the recommendation provided by the expert is exactly what the problem owner wants. Thus, if the simulation modeller starts collecting the data without exactly understanding the problem, he/she may need to recollect the relevant data. Hence, the time and cost will increase significantly.

Because of this disadvantage, the usage of simulation is relatively low in the business area. According to a survey about the use of business process simulation (BPS) by different practitioners conducted by Melão and Pidd (2003), nearly 80% of respondents claimed that they did not use simulation in designing and improving business processes. There are several reasons why they refused to use simulation. In 82 replies, 71 said that they are not using simulation giving the following reasons: 34 think that is because of the nature of their current job; 17 think that it is the nature of the process/problem; 10 think that it is because of the limitation of BPS; 7 think that it is because of the context of the organisation; and 3 think that it is because of their lack of expertise/awareness. From the limitations of BPS, 4 of them think it is too time and resource-consuming; 2 failed to find suitable software; 2 felt that simulation is too complex; 1 felt that it is too difficult to justify investment and 1 felt that it is not always appropriate.

As we can see from this survey, 4 out of 10 mentioned that applying simulation in their project was too time consuming. However, BPS projects are normally short, and the funding of the project is relatively less compared with other projects (Melão and Pidd 2003). Therefore, it is necessary to discover a method which can help simulation to apply to a BPS project.

In this section, it is justified that being time-consuming, resource-consuming, and difficult to justify investment, are the main disadvantages of classical simulation. As a result, a method to find out the requirements to overcome these disadvantages are discussed in the next section.

WHAT DOES THIS PROBLEM NEED?

A fast and cheap way needs to be pursued to tackle the above problem. According to the frameworks by Shannon (1998) and Law and Kelton (2000), a model can start to be constructed after collection of the required data. Pidd (1998) mentioned that a model can be built by using computer programming code or a simulation software package. Using programming code has the advantage of high flexibility and low cost, but it is extremely time consuming whilst, construction of a model by using a package greatly reduces the time required due to the cost of packages being different, the amount of money required to be spent on package is variable.

According to a survey conducted by Hlupic (2000), over 60% of academic simulation users think that ease of modelling is necessary; while just 20% think that flexibility is important. Flexibility can be defined as easy to extend (Snowdon et al. 1998), coupled with the ability to link to external code (Hlupic 2000). When constructing models, modelling flexibility is the ability to model any system, no matter how much a system is complex or unique (Law and McComas 1997; Law and McComas 1998). Ease of use is the second criteria for selecting tools for problem solving. According to Snowdon et al. (1998), ease of use is defined as “user-friendly to multiple types of users: business process analysts, planners, and operations research experts throughout the organization”.

Criteria for Tackling the Problems

Based on the results of the survey by Melão and Pidd (2003) and Hlupic (2000), a set of criteria in terms of *time* and *cost*, *ease of use* and *flexibility*, need to be defined before finding a new way to construct a simulation model.

Time and cost are the first criteria that needs to be achieved. As discussed above, constructing simulation model is very time consuming and hence expensive. As a result, it is necessary to find a way which can construct a model in a faster time, hence, the amount of money required can be reduced.

The second criterion is that the tool used to construct a simulation model should be easy to use and high flexibility. Because most of the simulation packages are either too difficult to use or low flexibility, the problem owner may be reluctant to adopt the simulation model to help to solve their problem. As a result, it is also important that the introduced modelling method is easy to use and has high flexibility.

In this section, the disadvantages of classical simulation model have been discussed. Some criteria for constructing simulation which can reduce the problems of classical simulation model have been indicated. Based on these criteria, a new way to build simulation model will be introduced.

A PROPOSED SIMULATION MODELLING FRAMEWORK

One possible way to achieve the above-mentioned criteria is to assemble a model instead of constructing a model from scratch (Paul and Taylor 2002). The assembly of different existing objects together can save the time of building the objects from scratch, hence saving money. Moreover, if a tool can allow a simulation modeller to construct a model by putting different objects together, it will become easy to use and have high flexibility. When the user feels that something is not relevant, he/she can change it quickly and use some relevant objects instead. In addition, because the model is built by assembling different objects together the model can be used easily.

This paper proposes an assembly based framework based on the above idea. The proposed framework of assembly model is named the "Grab and Glue, Run, Reject and Retry" (G^2r^3) simulation modelling approach as shown in Figure 4 (Paul, 2002). The concept is based on grabbing different objects from the web and gluing them together to form a model.

According to Yücesan et al. (2001), the web has experienced tremendous growth since the 1990s. The required objects for assembling a model can be easily grabbed from the Internet. After that, they can be glued together. If the model is satisfactory, life moves on and problem owners can continue their work. If it is unsatisfactory, the undesired parts of the model will be rejected. The grabbing process will be repeated and glued to the relevant position. This process will be iterated until the model is satisfactory.

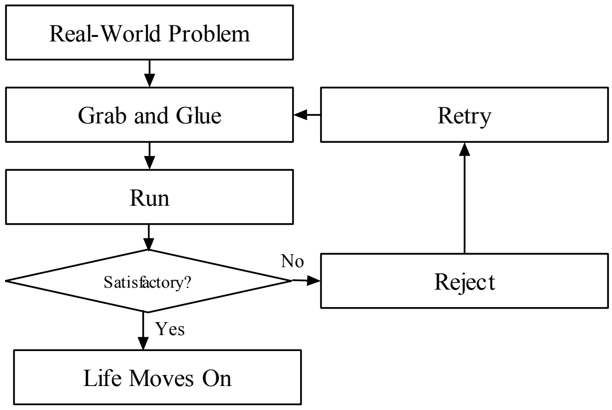

Figure 4: Framework for Grab and Glue Approach (Paul, 2002)

Can G^2r^3 Helps to Solve the Problems?

G^2r^3 attempts to reduce the time for model development. It does so by rethinking the concept of model development rather than merely reinventing a new tool that follows the procedures in Figures 1, 2, and 3. Since the aim of modelling is to understand the problem, data collection is deemed unnecessary, because the collected data is not usually the right data and the problem is not well understood. G^2r^3 concentrates on enhancing the debate between the problem owners rather than producing mathematically precise models that bear no relation to the unknown problem.

G^2r^3 could also provide assistance on the second criteria, flexibility. The main principle of G^2r^3 is to construct a model by assembling different objects. If the problem owners are not satisfied with the result of the model, irrelevant objects will be rejected and relevant ones will be glued easily. This process will be iterated until the problem owners are satisfied with the results from the constructed model. Models constructed by assembling techniques are easier to build and higher flexibility.

Object Reuse

G^2r^3 heavily depends on object reuse. The idea of object reuse has already been adopted for a few years in software engineering and simulation. Object reuse has the benefits of reducing software development time and costs, increasing software productivity, improving software system interoperability, reducing the number of people required to develop software, reducing the maintenance costs and producing better quality software (McClure, 1999). According to McClure (1995), from the organizational perspective, reuse can shorten development time, reduce costs and increase competitiveness; from a personnel perspective, the productivity can be increased; while from the customer perspective, a greater user satisfaction through the production of more flexible products can be achieved.

Pierre and Nouisser (2000) show that reusing graph theory algorithms, in terms of components reuse, can increase the reliability of the software, and increase the maintainability by applying on a telecommunications network design. Bellettini et al. (2001) agree that reuse can increase product quality and decrease time-to-market, adding to the competitive edge of software development enterprises. Etzkorn et al. (2001) agree that software reuse can increase productivity, reduce costs, and improve quality. Ewing (2001) agrees that reuse can have significant effects on the cost and worth to use on simulation.

The idea of the grab and glue principle is not new, although its use in this domain is new. Mackulak et al. (1998) stated that reuse of existing generic models like simulators or software packages that contain pre-programmed models, can reduce model building time as well as increasing simulation accuracy. An automated material handling systems (AMHS) design project was used to investigate the effectiveness of simulation modelling reuse, and finally, to discover that both the model building and analysis time have been reduced from over six weeks to less than one week (Mackulak et al. 1998). Although most of the simulation packages such as Simul8 are now using the idea of grab and glue, objects are only reused within the same simulation package. However, in G^2r^3, what we are looking for is to find objects from anywhere on the Web instead of a specific simulator.

A CRITIQUE AND LIMITATIONS OF G^2r^3

Because G^2r^3 allows simulation modellers to build a model by assembling different objects together, it can greatly reduce the time required and, hence, the cost. Moreover, assembling a model is much easier than building a model using programming code. Thus, it is believed that this modelling method can help to reduce the disadvantages of classical simulation modelling methods. However, it could be argued that G^2r^3 framework might lose the accuracy of the model. The accuracy of the model can be influenced by two factors: the accuracy of the selected objects or the accuracy of the result of the model. For the first factor, it is important to make sure that the collected objects are accurate enough before applying them to the assembly of the model. For the second problem, it is necessary to know that the purpose of the simulation model is to help the decision maker to make decisions, or to help the problem owner to gain an understanding of their problem (Paul and Taylor 2002). The grab and glue approach is only fit for the purpose of simulation modelling rather than being an elegant calculating machine. The underlying principle of G^2r^3 is to enable modellers and problem owners to collaboratively develop better understanding through continuous modelling process using the G^2r^3 approach. Because the design of a simulation model is not for finding out an exact solution, the interest of the numerical output becomes insignificant.

The framework of G^2r^3 is much simpler than the framework in Figures 1, 2, and 3. This is because the whole concept is each problem has its own criteria. It is not possible to develop an overarching methodology that is so detailed. It may also be argued that it is difficult to motivate the simulation experts to public their simulation object or the Web. However, they will do so because they will benefit from each other by using the Web.

At the moment, G^2r^3 is still at its infancy. It is difficult to conclude that which software tools or programming languages should be used for model construction. However, our aim is to use whatever is available to produce the model, object or web.

CONCLUSIONS AND FUTURE WORKS

In this paper, different simulation modelling frameworks have been reviewed and analysed. The reasons for lack of usage of simulation models has been identified as time consuming and hence high costs are associated with the modelling process. It is always the case that models are either too complex to construct or there is not enough flexibility associated with modelling tools. In order to solve the problem, G^2r^3 is introduced. However, it is not possible to conclude as yet whether constructing simulation following this approach can be successful or not, and this is due to the novelty of the approach. Different simulation software as well as programming tools will be studied to find out whether there are any existing tools suitable for assembling model by the idea of G^2r^3. After that, some realistic case studies will be conducted to analyse the performance of G^2r^3. If constructing models by assembling different objects is successful, it is believed that the disadvantages of classical simulation modelling could be significantly reduced.

REFERENCES

Banks, J.; J. S. Carson II; B. L. Nelson; and D. M. Nicol. 2001. *Discrete-Event System Simulation.* Prentice Hall International Series, London, 3rd ed.

Bellettini, C.; E. Damiani; and M. G. Fugini. 2001. "Software Reuse in-the-small: Automating Group Rewarding." *Information and Software Technology* 43, 651-660

Eldabi, T. 2000. Simulation Modelling: Problem Understanding in Healthcare Management. Unpublished PhD Thesis. Brunel University, London.

Etzkorn, L. H.; W. E. Hughes Jr.; and C. G. Davis. 2001. "Automated Reusability Quality Analysis of OO Legacy Software." *Information and Software Technology* 43, 295-308

Ewing, M. 2001. "The Economic Effects of Reusability on Distributed Simulations." In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters; J. S. Smith; D. J. Medeiros; and M. W. Rohrer. Association for Computing Machinery, New York, 812-817.

Hlupic, V. 2000. "Simulation Software: An Operational Research Society Survey of Academic and Industrial Users." In *Proceedings of the 2000 Winter Simulation*

Conference, ed. J. A. Joines; R. R. Barton; K. Kang; and P. A. Fishwick. Association for Computing Machinery, New York, 1676-1683

Law, A. M. and W. D. Kelton. 2000. *Simulation Modelling and Analysis*. McGraw-Hill International Series, Singapore, 3rd ed

Law, A. M. and M. G. McComas. 1998. "Simulation of Manufacturing Systems." In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros; E. F. Watson; J. S. Carson; and M. S. Manivannan. Association for Computing Machinery, New York, 49-52

Law, A. M. and M. G. McComas. 1997. "Simulation of Manufacturing Systems." In *Proceedings of the 1997 Winter Simulation Conference*, ed. S. Andradóttir; K. J. Healy; D. H. Withers; and B. L. Nelson. Association for Computing Machinery, New York, 86-89

Mackulak, G. T.; F. P. Lawrence; and T. Colvin. 1998. "Effective Simulation Model Reuse: A Case Study for AMHS Modeling." In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros; E. F. Watson; J. S. Carson; and M. S. Manivannan. Association for Computing Machinery, New York, 979-984

McClure, C. 1999. Reuse services --- Extending Software Development Methodologies to Support Reuse. Extended Intelligence, Inc. Available online via < http://www. reusability.com/serv2.html> [accessed April 20, 2002].

McClure, C. 1995. Model-Driven Software Reuse: Practing Reuse Information Engineering Style. Extended Intelligence, Inc. Available online via <http://www.reusability. com/papers2.html> [accessed April 20, 2002].

Melão, N. and M. Pidd. 2003. "Use of Business Process Simulation: A Survey of Practitioners." *Journal of the Operational Research Society* 54, 2-10.

Paul, R. J. 2002. "The Internet: An End to Classical Decision Modelling?" In *Internet Management Issues: A Global Perspective*, J. D. Haynes (Ed.). Idea Group Publishing and Information Science Publishing, Hershey, 209-219

Paul, R. J. and S. J. E. Taylor. 2002. "What Use is Model Reuse: Is There a Crook at the End of the Rainbow?" In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan; C. H. Chen; J. L. Snowdon; and J. M. Charnes. Association for Computing Machinery, New York, 648-652. Available online via < http://www.in forms-cs.org/wsc02papers/083.pdf> [assessed January 26, 2003].

Pidd, M. 1998. *Computer Simulation in Management Science.* John Wiley & Sons, Chichester, 4th ed.

Pierre, S. and N. Nouisser. 2000. "Reusing Software Components in Telecommunications Network Engineering." *Advances in Engineering Software* 31, 159-172.

Shannon, R. E. 1998. "Introduction to the Art and Science of Simulation." In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros; E. F. Watson; J. S. Carson; and M. S. Manivannan. Association for Computing Machinery, New York, 7-14

Simulation Dynamics, Inc., SDI 2001. Simulation dynamics. Available online via < http://www.simulationdynamics .com/Simulation/SimulationDefined.htm> [accessed January 17, 2002].

Snowdon, J. L.; S. El-Taji; M. Montevecchi; E. MacNair; C. A. Callery; and S. Miller. 1998. "Avoiding the Blues for Airline Travelers." In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros; E. F. Watson; J. S. Carson; and M. S. Manivannan. Association for Computing Machinery, New York, 1105-1112

Yücesan, E.; Y. C. Luo; C. H. Chen; and I. Lee. 2001. "Distributed Web-Based Simulation Experiments for Optimization." *Simulation Practice and Theory* 9, 73-90.

AUTHOR BIOGRAPHIES

TILLAL ELDABI is a lecturer at the Department of Information Systems and Computing at Brunel University, UK. He received a BSc in Econometrics and Social Statistics from the University of Khartoum. He received his MSc in Simulation Modelling and his PhD from Brunel University. His research is in aspects of healthcare management and the intervention of simulation and his main research also concentrates on the economy of healthcare delivery. He is looking to exploit the means of simulation on the wider information systems management area to assist in problem understanding. Dr. Eldabi's email and web addresses are <tillal.eldabi@brunel.ac.uk> and <www.brunel.ac.uk/~cssrtte>, respectively.

MAN WAI LEE is a PhD student in the Centre for Applied Simulation Modelling and the VIVID Research Centre at the Department of Information Systems and Computing, Brunel University, U.K. He is now under the supervision of Professor Ray J. Paul and Dr. Tillal Eldabi. He received a B.Eng in Department of Mechanical Engineering in The University of Hong Kong and a M.Sc. (with distinction) in Building Services Engineering from Brunel University. His main research concentrates on fast simulation process, the new G^2r^3 modelling technique. His email address is <manwai.lee@brunel.ac.uk>.

RAY J. PAUL is a Professor of Simulation Modelling, Director of the Centre for Applied Simulation Modelling, creator of the Centre for Living Information Systems Thinking, and Dean of the Faculty of Technology and Information Systems, all at Brunel University, UK. He received a B.Sc. in Mathematics, and an M.Sc. and a Ph.D. in Operational Research from Hull University. He has published widely, in books, journals and conference papers, many in the area of simulation modelling and software development. He has acted as a consultant for a variety of United Kingdom government departments, software companies, and commercial companies in the tobacco and oil industries. He is the editor of the Springer-Verlag Practitioner book series. His research interests are in methods of automating the process of modelling, and the general applicability of such methods and their extensions to the wider arena of information systems. He is currently working on widely aspects of simulation, in particular in Web-Based Simulation and the new G^2r^3 modelling technique. His email and web addresses are <ray.paul@brunel.ac.uk> and <www.brunel.ac.uk/~csstrjp>.