Proceedings of the 1986 Winter Simulation Conference
J. Wilson, J. Henriksen, S. Roberts (eds.)

MULTIFACETTED, MULTIPARADIGM MODELLING PERSPECTIVES: TOOLS FOR THE 90'S

Bernard P. Zeigler
Department of Electrical
and Computer Engineering
The University of Arizona
Tucson, Arizona 85721
USA

Tuncer I. Oren
Computer Science Department
University of Ottawa
Ottawa, Ontario K1N 9B4
Canada

ABSTRACT

Multiplicities of perspective inherent in modelling and simulation methodology are enumerated and rationales given for their existence. Characteristics of futuristic simulation environments which support flexible adoption of multiple perspectives are outlined. Finally, we discuss the construction of models which simultaneously embody differing perspectives. Advances in modelling methodology along these lines will constitute a quantum leap in tool sophistication which can greatly extend the domain of simulation application.

1.0 INTRODUCTION

The complexity of man's environment seems to expand with every technological success. No longer threatened by cold and wild animals, he is now threatened by exploding complexity (Beer, 1975). In the past, adopting a single perspective was a useful way to manage complexity. Pretending that only certain factors count works well to a limited extent: it delays dealing with the rebounded consequences of those ignored. A model is a "macroscope", a conceptual tool to "observe" a complex system (Rosnay, 1975) from a particular point of view. Many such macroscopes are needed to counterbalance the blinding effects of a single one. A new level of simulation environment is needed to encourage and support the routine adoption of such multiple perspectives.

Some concepts already exist for developing such simulation environments. Multifacetted modelling methodology (Zeigler, 1984) signifies an approach to simulation modelling that recognizes the irreducible complexity of reality, while affirming that useful partial models can be constructed in the service of limited decision making objectives. Essential to the methodology is its support of an organized base of models whose partial perspectives can be integrated to achieve a coherent whole. Recently, the infusion of artificial intelligence (AI) paradigms into simulation has opened up a related source of multiple paradigms for dealing with complexity.

In this paper, we explore the implications of the new concepts of multifacetted, multiparadigm modelling and simulation. First we analyze the various connotations of the term "modelling perspective": one can adopt a single modelling objective, a single level of aggregation, a single level along the behavior-structure axis, a single modelling formalism, or a single programming paradigm. Or one can work in a simulation environment which encourages the adoption of multiple perspectives, hence supports the use of multiple modelling objectives, multiple levels of aggregation, multiple levels of structure and behavior, multiple modeling formalisms, and multiple programming paradigms. Thus, we outline the characteristics and utilities of much an environment. Finally, as models of cognizant systems (Oren and Zeigler, 1986) come increasing to the fore, they will faithfully represent the multiple perspectives needed for truly effective decision making. Hence, we examine the construction of models which embody multiplicities of goals, structural change, and formalism.

2.0 MULTIPLE PERSPECTIVES

2.1 Multiple Objectives

Modelling and simulation activities are carried out to achieve a multiplicity of objectives which arise from the goals of either gaining knowledge about a real system or of exerting control, management or design interventions on it. Although many objectives may have to be accomplished simultaneously in such interventions, often a simulation study is conducted only with a particular objective in mind -- the other objectives may be considered by employing other analytic or intuitive means. Multifacetted methodology is an antidote to this approach.

2.2 Multiple Levels of Aggregation

Models may be constructed at different levels of aggregation (resolution, abstraction). The level is jointly determined by the objectives at hand, the available knowledge, and the given resource/time constraints. The objectives, including the accuracy desired of the answers, dictate the minimal degree of disagregation needed in the model to be able to satisfy the objectives. Beyond this level further disaggregation may be futile or even counterproductive. The concepts of (generic) experimental frame and applicability of frames to models have been introduced to link the objectives with the models. Available knowledge places a lower limit on the aggregation limit -- in principle (from a reductionist standpoint), one could always sink to the level of basic physical particles in every model provided that relationships were available to express the desired behavior of interest. Likewise, since as resolution increases, the complexity of the model almost always rapidly increases, time and money determine how much disagregation can be considered.

2.3 Multiple Levels of Behavior and Structure

Considering a real system as a black box, there is hierarchy of levels at which its models may be constructed ranging from the purely behavioral -- in which the model claims to represent only the observed input/output behavior of the system, up the strongly structural -- in which much is claimed about the structure of the system. Simulation models are usually placed at the higher levels of structure and they embody many supposed mechanisms to generate the behavior of interest. In contrast, behavior descriptions obtained by curve fitting represent lowest level models. High structural detail usually implies a high degree of disagregation in model variables needed to express the relationships involved. However, the converse is not necessarily true: one can employ a high degree of resolution in defining variables and corresponding measurement and still use only curve fitted relationships to express the values of variables over time.

2.4 Multiple Formalisms

Simulation models can be specified in a number of formalisms and simulated (i.e., have their behavior generated) in a variety of corresponding simulation media. Formalisms are set-theoretic short-hands for specifying mathematical dynamic systems. Often, formalisms are associated with particular simulation languages, so that for example, the break down of discrete event formalisms into event-based, activity scanning, and process interaction subclasses, is mirrored by simulation languages which support the corresponding world views. Nevertheless, formalisms have an independent conceptual existence. Indeed, as in the case of modular, hierarchical DEVS (discrete event system specifications), the formalism was defined before any simulation languages had been developed to express it. No one formalism is best to represent the variety of behaviors in real systems of interest; some formalisms are more natural (correspond more directly with the preceived operation) and (usually at the same time) lead to more computationally efficient simulation than others in certain domains. The applicability of a formalism does not depend so much on disciplinary area (physical, biological, industrial, etc.) as on the level of aggregation indeed, the formalism may alternate with level, going from continuous change to discrete change and back as successive layers of description are unfolded.

2.5 Multiple Programming Paradigms

Descending a level of abstraction from models to their programmed implementations, there are emerging distinct paradigms for conveying instructions to a computing system. In conventional procedural languages, the programmer lays down an explicit sequence of instructions. In logic programming of the PROLOG variety one provides sets of clauses specifying goals and the preconditions necessary for their satisfaction -- the language interpreter itself searches through all possible sequences of clauses for those leading to satisfaction of the main goal. Expert system shells provide media for rule-based programming (rules, i.e., condition-action pairs, are executed under the control of an inference engine). Yet other programming styles are being developed for parallel processors. Finally, in contrast to conventional procedure-oriented programming, modern object-oriented programming encourages association of procedures and data structures with the objects they relate to.

3.0 ENVIRONMENT SUPPORT OF MULTIPLE PERSPECTIVES

3.1 Varied Objectives Within Environment

Although, when approached with differing objectives, a real system may yield various distinct models, there is nevertheless an underlying unity that binds these models together -- namely, their common origin. Rather than can consider each model as a distinct entity, an environment can support the integration of models so that a coherent whole emerges. With this support, the construction of models to meet new objectives may be fostered, since components already existing in the model base may be exploited. To gain full advantage of the knowledge in the model base, there must be provided a strong capability of representing the components of models, their variations and their interconnection. To facilitate synthesis, models must be readily dissembled into components, and these must be able to be easily assembled into new combinations, i.e., the environment should support modular, hierarchical model construction.

3.2 Varied Levels of Aggregation Within Environment

As indicated above, models oriented to fundamentally the same objectives may be constructed at different aggregation levels due to tradeoffs in accuracy achievable versus complexity costs incurred. An environment can support construction of such aggregation related models by facilitating elaboration of models (constructing a new model related to an existing one by adding new variables or refining their ranges) and simplification of models (constructing a new model by dropping variables, coarsening their ranges, or grouping several together to form aggregated variables). Moreover, relationships among such collections of models should form part of the knowledge base of the environment and be available for use in model validation (against the real system) and cross-validation (against each other).

3.3 Varied Levels of Behavior and Structure Within Evironment

While simulation models are, by nature, formulated at high levels of structure, an environment may support the development of other kinds of models at lower levels.

Behavioral descriptions obtained by curve fitting, statistical correlations, or inductive systems modelling may complement the simulation models by providing summaries of real system behavior. Such summaries may replace the original extensive records and therefore may be more economical to use in such activities as validation of simulation models of the same phenomena. The same kinds of techniques can be applied to simulation generated behavior. For example, statistical metamodels summarize the dependence of a performance index on model parameters. Introspective simulation employs artifical intelligence to discover causal relationships in simulation records to form symbolic cause-effect models (Reddy et. al., 1986).

3.4 Varied Formalisms Within Environment

An environment can support the use of various model expression formalisms to lessor or greater extent. At the least, it can make available various simulation languages associated with the different formalisms. More satisfactory would be to provide a uniform simulation language in which all other methodological components remain the same, while the formalism can be chosen at will. In this way, incompatibilities arising due to differing conventions (unrelated to formalism differences) are obviated. For example, the methodologically desirable decomposition into model and experimental frame segments is independent of model formalism and may be supported by a uniform simulation language. Still greater support can be rendered by providing tools to transform model specifications from one formalism to another. For example, models expressed in non modular formalisms (associated with conventional languages, can be algorithmically transformed into modular equivalents. Likewise, continuous time models can be transformed into discrete time and discrete event versions which may be computationally more efficient. Another alternative is to provide a universal formalism for initial model expression, and tools to find the optimal special formalism for a model, once specified in the general terms. However, this latter approach may encounter the problem that modellers may not as readily be able to formulate models in the universal formalism as in the special formalisms, which may enjoy conceptual advantages over it.

3.5 Varied Programming Paradigms Within Environment

Environments which support the construction of knowledge-based simulations should provide a choice of programming paradigms to implement various functional elements. Object-oriented programming may be best to develop the knowledge representation data structures (system entity structures) for model organization. Logic programming may be most natural for model synthesis since such programs can readily generate the hierarchical compositions spanned by an entity structure, filtering out those that do not satify the coupling constraints. Rule-based programming may be provided to conveniently develop expert systems to work in conjunction with the simulation models.

4.0 MULTIPLE PERSPECTIVES WITHIN MODELS

4.1 Varied Objectives Within Model

To achieve realism, models of intelligent agents must be able to represent goal-directed behavior. Humans can generate new goals to strive for, prioritize goals, take actions to achieve highest priority goals, and guage progress in reaching them. Likewise, models of humans or other intelligent agents may need to possess these features. When intelligent agents employ models to help achieve goals, they each set up what we referred to above as modelling objectives. Thus, models of intelligent agents must to some degree to able to represent not only multiple and variable goal manipulation, but also the generation of modelling objectives and the synthesis of models to meet these objectives.

4.2 Varied Levels of Aggregation Within Model

Intelligent systems ought to be able to change the level of aggregation in their observation of the system they are interacting with. For example, intelligent control systems should be able to focus their attention on a critical behavior of the controlled object and to allocate resources to this behavior so long as it remains critical. Likewise, simulation models of intelligent systems ought to be able faithfully model such systems, hence should have the capability of controlling the level of aggregation autonomously.

4.3 Varied Levels of Behavior and Structure Within Model

Conventional simulation systems adequately support only a single level at which change occurs in the model, that of changes in the model descriptive variables, viz. its behavior. Although changes in the model structure may be introduced, and to some extent, tracked in such systems, specific and powerful suuport is not provided for such activities. Biological, and other adaptive, systems are most readily perceived as exhibitting changes simultaneously at structural and behavioral levels. A new paradigm, structural simulation (as opposed to conventional "trajectory" simulation) is needed to avoid having to force structural changes down to the same level as behavioral ones. Simulation technology must be advanced to deal with the problems of i) specifying the intelligence within a model to determine structure state transitions, ii) handling behavior during the intervals in which such structural changes occur, iii) providing experimental frames and measurements sensitive to structural, as well as behavioral change, and iv) enabling such measurements to be employed within the model itself.

4.3 Varied Formalisms Within Model

Since components of a model may be most naturally expressed in distinct formalisms, one should not be forced into a Procrustean bed of a single, or small number of, formalisms by a model specficiation environment.

Certainly, differential equation components (continuous change) should be interfaceable with discrete event and discrete time (discrete change) components. Moreover, symbolic model components, representing intelligent agents for example, should be readily coupled with conventional simulation model components.

CONCLUSIONS

The tendency among some simulation system makers has been to "grope in the light" (Oren and Zeigler, 1986), preferring to perfect the tools laid down decades ago, rather than step out with bold new ones into the unexplored. The success of Artificial Intelligence, an alternative in knowledge representation and utilization, in capturing the public imagination and full scale commercial interest shows the limitations in this conservative approach. The lesson is plain: take the next step!

BIBLIOGRAPHY

Beer, S. (1975). Platform for Change. John Wiely, New York.

Birtwistle, G. (1985). Artificial Intelligence, Graphics, and Simulation. Society for Computer Simulation, San Diego, CA.

Bobrow, D.G. (Ed.) (1985). Qualitative Reasoning about Physical Systems. MIT Press.

Clema, J.K., Werling, R., Chande, A: (1986). Knowledge Bases and Machine Learning. In: Proceedings of Application of Artificial Intelligence to Engineering Problems, Southampton, England, April 1986.

Davis, P.K. (1986). Applying Artificial Intelligence Techniques to Strategic-Level Gaming and Simulation. In: Modelling and Simulation Methodology in the Artificial Intelligence Era, M.S. Elzas, T.I. Oren. B.P. Zeigler (Ed.s). North Holland, Amsterdam.

Elzas, M.S., Oren, T.I., Zeigler, B.P. (Eds.) (1968). Modelling and Simulation Methodology in the Artificial Intelligence Era, North Holland, Amsterdam.

Forbus, K.D. (1981). Qualitative Reasoning about Physical Processes. Proceedings of 7th IJCAI, International Joint Conf. on Artificial Intelligence, Vancouver, Canada, pp. 326-330.

Hayes, P. (1979). The Naive Physiscs Manifesto. In: D.E. Mitchie (Ed.), Expert Systems in the Microelectronics. Edinburg University Press.

Hogeweg, P., Hesper, B. (1979). Heterarchical, Selfstructuring Simulation Systems: Concepts and Applications in Biology. In: Methodology in Systems Modelling and Simulation, B.P. Zeigler, M.S. Elzas, G.J. Klir, T.I. Oren (Eds.). North Holland, Amsterdam, pp. 221-232.

Holmes, W.M. (Ed.) (1985). Artificial Intelligence and Simulation. Society for Computer Simulation, San Diego, CA.

Kerckhoffs, E.J.H., Vansteenkiste,/G., Zeigler, B.P. (Eds.) (1986). Artificial Intelligence Applied to Simulation. Society for Computer Simulation, San Diego, CA.

Luker, D.E. (1969). The Art of Computer Programming, Vol. 1, Fundamental Algorithms. Addison-Wesley, Reading, MA.

Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (Eds.) (1984). Machine Learning - An Artificial Intelligence Approach, Springer Verglag, Berlin.

Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (Eds.) (1986). Machine Learning - An Artificial Intelligence Approach, Vol. 2, Morgan Kaufmann Publishers, Los Altos, CA.

Oren, T.I. (1974). Deductive General System Theories and Simulation of Large Scale Systems. In: Proceedings of Summer Computer Simulation Conference, Houston, TX, July 9-11, 1974, pp. 13-16.

Oren, T.I. (1975). Simulation of Time-Varying Systems. In: Advances in Cybernetics and Systems, J. Rose (Ed). Gordon and Breach Science Publishers, England, pp. 1229-1238.

Oren, T.I. (1977). Cybernetics and Modelling And Simualtion of Large Scale Systems. International Association for Cybernetics, Namur, Belgium.

Oren, T.I. (1978). A Personal View on the Future of Simulation Languages (Keynote Paper). In: Proceedings of the 1978 UKSC Conference on Computer Simulation, IPC Science and Technology Press, Chester, England, April 4-6, 1978, pp. 294-306.

Oren, T.I. (1980). Design of SEMA: A Software System for Computer-Aided Modelling and Simulation of Sequential Machines. In: Proceedings of 1980 Winter Simulation Conference, T.I. Oren, C.M. Shub, P.F. Roth (Eds.). IEEE, New York, pp. 113-123.

Oren, T.I. (1981). Concepts and Criteria to Assess Acceptability of Simulation Studies: A frame of Reference. CACM, Vol. 24, No. 4, pp. 180-184.

Oren, T.I. (1984a). Direction for the Future of Modelling and Simulation: A Normative View. International Journal of Systems Analysis, Modelling, Simulation, Vol. 1, No. 4, pp. 293-306.

Oren, T.I. (1984b). Model-Based Activities: A Paradigm Shift. In: Simulation and Model-Based Methodologies: An Integrative View. T.I. Oren, B.P. Zeigler, M.S. Elzas (Ed.s). North-Holland, Amsterdam, pp. 3-40.

Oren, T.I. (1984c). GEST - A Modelling and Simulation Language Based on System Theoretic Concepts. In: Simulation and Model-Based Methodologies: An Integrative View. (Eds.). North-Holland, Amsterdam, pp. 281-335.

Oren, T.I. (1984d). Model-Based Information Technology: Computer and System Theoretic Foundations. Behavioral Science, July 1984, Vol. 29, No. 3, pp. 179-185.

Oren, T.I. (1986a). Implications of Machine Learning in Simulation. In: Modelling And Intelligence Era. M.S. Elzas, T.I. Oren, B.P. Zeigler (Eds.), North-Holland.

Oren, T.I. (1986b). Artificial Intelligence in Quality Assurance of Simulation Studies. In: Modelling And Simulation Methodology in the Artificial Intelligence Era. M.S. Elzas, T.I. Oren, B.P. Zeigler (Eds.), North-Holland, Amsterdam.

Oren, T.I. (1986c - In Press). Model Behavior: A Taxonomy of Types and Generation and Processing Techniques. In: Systems and Control Encyclopedia: Theory, Technology, and Applications, M. Singh (Ed.), Pergamon Press, Oxford, England.

Oren, T.I. (1986d). Artificial Intelligence and Simulation. In: Artifical Intelligence Applied to Simulation. E.J.H. Kerckhoffs, G.C. Vansteenkiste, B.P. Zeigler (Eds.). Society for Computer Simulation, San Diego, CA, pp. 3-8.

Oren, T.I., Elzas, M.S., Sheng, O. (1985a). Model Reliability and Software Quality Assurance in Simulation of Nuclear Fuel Waste Management Systems. In: Waste Management '85, Vol. 1, R.G. Post (Ed.), pp. 381-396.

Oren, T.I., Ziegler, B.P. (1979). Concepts for Advanced Simulation Methodologies. Simulation, Vol. 32, No. 3, pp. 69-82.

Oren, T.I., Ziegler, B.P. (1986 - Submitted). Artificial Intelligence in Modelling and Simulation: Some Directions to Explore. Simulation.

Oren, T.I., Zeigler, B.P., Elzas, M.S. (1985b). Simulation and Model-Based Methodologies: An Integrative View. Springer-Verlag, Berlin.

Reddy, Y.V.R., Fox, M.S., Husain, N., McRoberts, M. (1986). The Knowledge-Based Simulation System, IEEE Software, March, 1986.

Zeigler, B.P. (1976). Theory of Modelling and Simulation. John Wiley, New York.

Zeigler, B.P. (1979). Modelling and Simulation Methodology: State-of-the-Art and Promising Direction. In: Proceedings of the IMACS Simulation of Systems Conference, North-Holland, Amsterdam.

Zeigler, B.P. (1984). Multifacetted Modelling and Discrete Event Simulation. Academic Press, London.

Zeigler, B.P. (Ed.) (1985). Expert Systems and Simulation Models. Seminar Notes, Tucson, AZ.

Zeigler, B.P. (1986a). Simulation Methodology for Variable-Structure Modelling. In: Modelling And Simulation Methodology in the Artificial Intelligence Era. M.S. Elzas, T.I. Oren, B.P. Zeigler (Ed.s), North-Holland, Amsterdam.

Zeigler, B.P. (1986b - to appear). Knowledge Representation from Newton to Minsky and Beyond. Applied Artificial Intelligence Journal.

Zeigler, B.P., De Wael, L. (1986). Towards a Knowledge-Based Implementation of Multifacetted Methodology. In: Artificial Intelligence Applied to Simulation. E.J.H. Kerckhoffs, G.C. Vansteenkiste, B.P. Zeigler (Eds.). Society for Computer Simulation, San Diego, CA, pp. 42-51.

Zeigler, B.P., Elzas, M.S., Klir, G.J., Oren, T.I. (Eds.) (1979). Methodology in System Modelling And Simulation. North-Holland Amsterdam.

Zeigler, B.P., Oren, T.I., Reddy, R. (In preparation). Knowledge Representation in Simulation Environments: Expert Systems and Simulation Models.