Gaines, B. R. (1986). From differential analyzers to knowledge-based systems: the changing technologies of simulation. In Proceedings of JSST Conference on Recent Advances in the Simulation of Complex Systems, pp. 271-276. Japan Society for Simulation Technology, Tokyo.

FROM DIFFERENTIAL ANALYZERS TO KNOWLEDGE-BASED SYSTEMS: THE CHANGING TECHNOLOGIES OF SIMULATION

Brian R. Gaines
Department of Computer Science, University of Calgary
Calgary, Alberta, Canada T2N 1N4

From the earliest days of electronic computing, techniques and applications of simulation have developed to exploit the increasing potential of computing technologies. Over the years the focus of attention in simulation research has changed from simulator hardware to simulation languages. In recent years the focus has begun to change again as expert system techniques have come to have a significant role in simulation. This paper analyzes the place of simulation technologies in the growth of the infrastructure of the computing industry, shows how the changing foci of attention correspond to major technological stages in computing, and uses this model to forecast future developments in simulation.

1 INTRODUCTION

Simulation was one of the earliest applications of computers in the 1940s and is a vigorous and growing area of computing science today. The technology used in simulation has changed dramatically over the past 45 years. The early wave of electronic analog computers, their counterparts in digital differential analysers, their extensions to hybrid computers—all these have played a significant role in their time but ultimately given way to the stored-program digital computer. The emphasis has changed from simulator hardware to simulation languages and the software engineering tools to develop accurate and efficient simulations rapidly. In recent years the focus of attention has changed again as expert system techniques have begun to play a major role in simulation and the development of knowledge-based systems has widened the scope of what may be simulated—from the numeric to the symbolic.

Are there any patterns to the development of simulation technologies and foci of attention that give meaning to the past and enable us to structure the future and predict further stages of development? What are the roles of the fifth [20] and sixth [9] generation computer development programs in the development of simulation, and what roles does simulation play in these programs?

This paper answers these questions from a number of viewpoints. It gives a model of the development of computing which accounts for the shifts in simulation technologies and the changing foci of attention in simulation research. It gives an overview of the current fifth and sixth generation development programs, analyses the impact of technological advances in these programs on simulation technology, and analyses the role of simulation in these programs.

2 THE INFRASTRUCTURE OF COMPUTING

In coming to understand computing technology, it is important to realise that the electronic computer itself is only one component in an elaborate and highly differentiated infrastructure. Like other basic technologies, the growth of computing is comprised of the envelope of learning curves in successive underlying technologies. This infrastructure has grown through a succession of *generations* of computers, each of which represents a major change in computing technology [6,7]. The role of simulation in fifth and sixth generation computing, and its implications for research and product development, can be analysed through analogy with other components of the infrastructure of computing. During the eight year span of each computing generation, revolutionary changes have taken place that correspond in magnitude to those taking place over some seventy years or more in the aircraft industry. The first generation commenced in 1948 and we are now in the fifth generation spanning 1980 through 1987.

The growth of the infrastructure of computing technology may be seen as the recursive application of the basic process underlying social and individual development, that of *learning* [13]. Marchetti has shown that treating society as a learning process leads to simple explanations of socio-economic processes such as business cycles and technology substitution [16]. The introduction of any new knowledge, technology or product follows a logistic *learning curve* as shown in Figure 1, in which growth takes off slowly, begins to climb rapidly and then slows down as all the information has been assimilated [28].

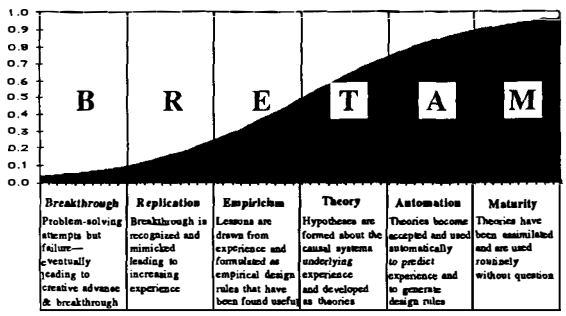


Fig.1 Phases of learning superimposed on logistic learning curve

It has been noted in many disciplines that the qualitative phenomena during the growth of the logistic curve vary from stage to stage [3]. If we take the amount of knowledge necessary for a mature technology to be 100%, then the 10%, 25%, 50%, 75% and 90% points are equally spaced in time along the logistic curve. These points mark significant phases in knowledge acquisition:–

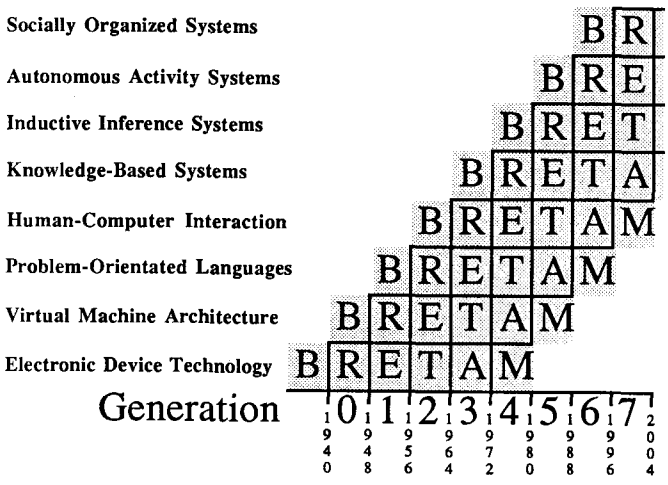
- When less than 10% of the knowledge is available, little can be achieved. We are in a pre-*breakthrough* period where most projects attempted will fail through lack of some critical information or skill.
- At around the 10% level some form of recognizable breakthrough becomes possible and we move into a phase of *replication* in which the breakthrough activity is noted and mimicked widely with little understanding.
- Replication leads to increasing experience and knowledge based upon it and when the 25% level is reached sufficient information is available for a phase of *empiricism* in which design rules are formulated based on experience.
- At the 50% level sufficient information has been accumulated for deeper patterns to be discerned, and a phase of *theory* commences in which causal rules are derived that allow further experience to be predicted and design rules to be derived.
- At the 75% level the theories have become accepted as the normative basis for further progress and an *automation* phase starts in which they are used for design and to search out the residual knowledge.
- Above the 90% level a phase of *maturity* is entered when the knowledge is expected to be available and used routinely.

This *learning curve model* may be applied to the substitution of different electronic device technologies (EDTs) as a basis for computing. It then distinguishes the generations of computers in terms of the technology in use:–

- The zeroth generation started in 1940 with the relay-based Bell Complex Number Computer followed in 1941 by the Zuse Z3.
- The first generation started in 1948 with the vacuum tube-based BINAC in the USA and EDSAC in the UK.
- The second generation started in 1956 with the transistor-based Bell Leprachaun followed in 1959 by the RCA 501.
- The third generation started in 1964 with the IBM 360 family using some integrated circuits.
- The fourth generation started in 1972 with the use of large-scale integration in the main memory of the IBM 370/158 and in the Intel 8008 8-bit microprocessor.
- The fifth generation started in 1980 with the use of very large-scale integration by IBM to put the 370 processor on a chip followed by the HP-9000 in 1981 with 450,000 transistors on a chip.
- Projecting the 8-year cycle of generations has the sixth generation starting in 1988 with ultra large-scale integration and some 10 million transistors on a chip.
- Projecting further has the seventh generation starting in 1996 with grand-scale integration and 1,000 million transistors on a chip.

3 QUALITATIVE CHANGES IN COMPUTING

The definition of generations in terms of EDT captures some important aspects of computing technology such as cost decreases, size decreases, power increases, and so on. However, it fails to account for the *qualitative* changes that have given computing its distinct character in each generation. The change from mechanical to electronic devices made it possible to store programs as data and enabled the use of computers as a general-purpose tool and then the development of programming language compilers. The transistor made reliable operation possible and enabled routine electronic data processing and then interactive timesharing. Integrated circuits reduced costs to the level where computers became commonplace and made possible the personal computer dedicated to a single user. What is missing from the model of computing developing through successive generations of EDT is the infrastructure that has been built above this. A complete account of the development of the computing industry must be two-dimensional with breakthroughs in key system technologies also represented.



B • Breakthrough: creative advance made
R • Replication period: experience gained by mimicing breakthrough
E • Empirical period: design rules formulated from experience
T • Theoretical period: underlying theories formulated and tested
A • Automation period: theories predict experience & generate rules
M • Maturity: theories become assimilated and used routinely

Fig.2 The infrastructure of computing as an envelope of learning curves in a sequence of technologies

Figure 2 shows this infrastructure with the phases on the learning curve of EDT at the base and successive computing technologies above it as the infrastructure.

Now the generations may be defined in terms of major qualitative changes:–

- The initial breakthrough in EDT leading to the zeroth generation is placed at 1940 about the time of the Atanasoff and Berry experiments with tube-based digital calculations.
- The first generation breakthrough was the introduction by Mauchly and Von Neumann of stored program and subroutine concepts around 1947 which led to the transition from the ENIAC to the EDVAC designs. This detached computing as a separate discipline from electronics by substituting software for hardware in a *virtual machine architecture* (VMA).
- The second generation breakthrough was to bridge the gap between machine and task through the development of *problem-orientated languages* (POLs) such as FORTRAN around 1956.
- The third generation breakthrough was to bridge the gap between the computer and the person with the development of *interactive* time-shared computers in 1964 allowing close *human-computer interaction* (HCI).
- The fourth generation breakthrough was in the early 1970s with developments in *expert systems* based on *knowledge-processsing*. Such *knowledge-based systems* (KBSs) have the capability to store information through its inter-relations and make inferences about its consequence.
- The fifth generation breakthrough was in 1980 with developments in machine learning and *inductive inference systems* (IISs).
- The sixth generation breakthrough may be seen as deriving from the growth of robotics in which goal-directed, mobile computational systems will act autonomously to achieve their objectives. The breakthrough into the sixth generation era commencing in 1988 will be one of *automomous activity systems* (AASs).

• The seventh generation breakthrough may be projected as deriving from interaction between these systems which will become increasingly important in enabling them to cooperate to achieve goals. The breakthrough into the seventh generation era commencing in 1996 will be one of *socially organized systems* (SOSs).

4 INVENTION, RESEARCH AND PRODUCTS

The model of the development of computing described above is important in any evaluation of the impact of computing. Each generation represents a revolution in technology with a qualitatively different impact. Each generation subsumes the capabilities of that preceding it, providing better facilities at very much lower cost, and adding new capabilities not possessed by the previous generations. The availability of these new capabilities in a generation is linked to the stage of the learning curve for each technology in that generation (Figure 3):–

• *Invention* is focused at the **BR** interface where new breakthrough attempts are being made based on experience with the replicated breakthroughs of the technology below.
• *Research* is focused at the **RE** interface where new recognized breakthroughs are being investigated using the empirical design rules of the technology below.
• *Product Innovation* is focused at the **ET** interface where new products are being developed based on the empirical design rules of one technology and the theoretical foundations of the technology below.
• *Established Products* are focused at the **TA** interface where product lines can rest on the solid theoretical foundations of one technology and the automation of the technology below.

For example, in the fourth generation (1972-79):–

• **BR**: recognition of the knowledge acquisition possibilities of KBS led to the breakthrough to inductive-infererence systems;
• **RE**: research focused on the natural representation of knowledge through the development of HCI, e.g. the Xerox Star direct manipulation of objects;
• **ET**: experience with the HCI using the POL BASIC led to the innovative product of the Apple II personal computer;
• **TA**: the simplicity of the POL RPG II led to the design of the IBM System/3 product line of small business computers.

In the current fifth generation (1980-87):–

• **BR**: recognition of the goal-seeking possibilities of IIS is leading to the breakthrough to automomous-activity systems in robotics;
• **RE**: research is focused on learning in KBS;
• **ET**: the advantages of the non-procedural representation of knowledge for HCI led to the innovative designs of the Visicalc spread-sheet business product and the LISP machine scientific product;
• **TA**: the ease of HCI through a direct manipulation POL led to the Apple Lisa/Macintosh product line of personal computers.

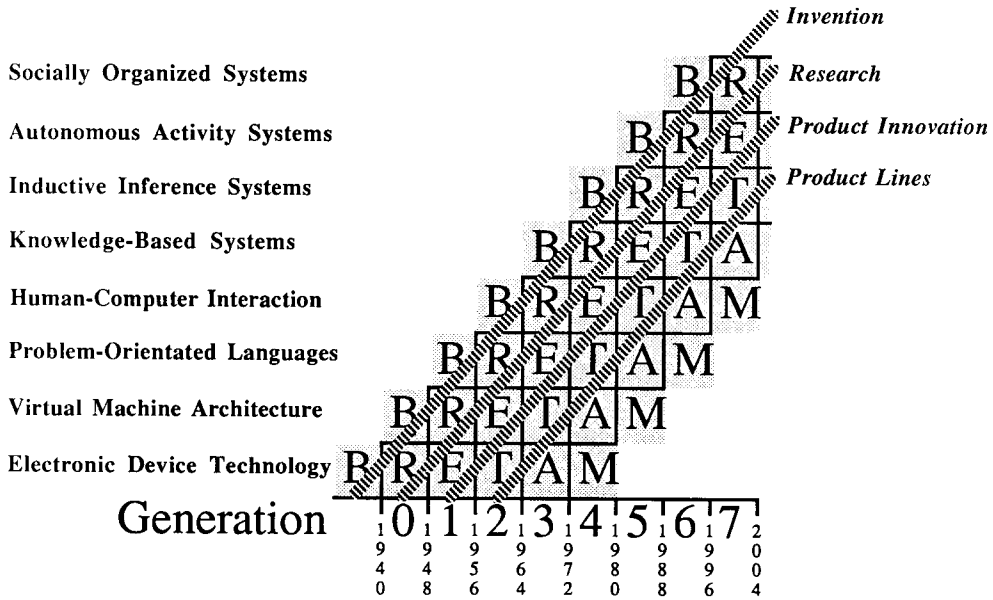


Fig.3 Invention, research, innovation and product lines through generations of computers

6 SIMULATION IN THE INFRASTRUCTURE

The model of the development of the infrastructure of the computing industry developed above gives a basis for analysing simulation studies and their changing technological priorities. Product innovation for a particular application such as simulation takes place when technologies have passed through the breakthrough and replication stages and are into the empirical and theoretical phases where design rules are available and theory is becoming available. This corresponds to the diagonal line marked *product innovation* in Figure 3—its relation to simulation technologies is shown in Figure 4.

The Past Four Generations: 1948-1980

In the first generation product innovation for simulation was concerned with the potential of electronic devices to simulate the behavior of differential equations. This led to the development of the *operational amplifier* in which a precision passive feedback network around an imprecise high-gain invertor is used to simulate the linear differential equations corresponding to the network [26]. It also led to the development of the *digital differential analyzer* in which the differential equations are solved by incrementing digital registers [15,17].

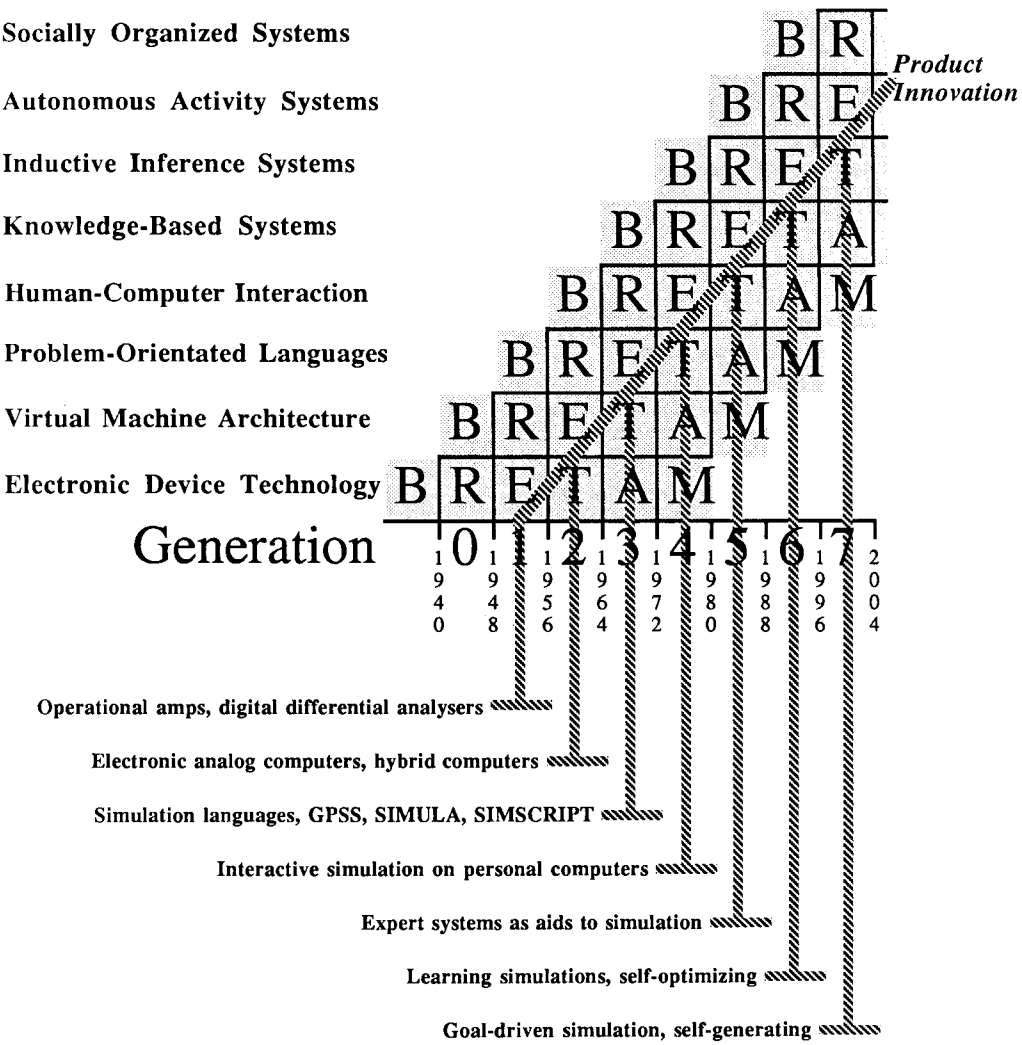


Fig.4 The changing foci of attention in simulation research related to the infrastructure of computing

In the second generation architectural considerations led to the combination of analog switches and operational amplifiers to give programmable *analog computers*. These could be combined with digital computers to give *hybrid computers* in which the analog sub-system is used for the rapid solution of differential equations and the digital sub-system is used to control decision processes, such as optimization, associated with the application of these equations [14]. Digital differential analyzers were similarly combined with switching logic [11,5] and general-purpose digital computers [24].

In the third generation problem-orientated languages for simulation began to promote the standard digital computer as a simulation tool. The development of such languages as GPSS, SIMULA and SIMSCRIPT made available techniques for simulating systems combining linear and nonlinear elements in an environment where further statistical processing of data could also be simply programmed [30]. In this third generation era the flexibility and increasingly widespread availability of the general-purpose digital computer began to win out against the speed and cost-effectiveness of special-purpose analog computers and digital differential analyzers. It is also of interest to note that the *blocks* specifying asynchronous, interacting objects to be simulated in these languages were early precursors of the *object-orientated* programming languages of today.

In the fourth generation the increased speed and reliability of computers shifted emphasis to interactive simulation and on-line decision-making. This again undermined the advantages of analog computers as interactive machines and these went out of manufacture during the fourth generation era. The human-computer interface to simulation languages, and ongoing simulations, became of major importance. In particular, the availability of low-cost color graphic interfaces made it possible to present both simulated structures and behaviors in pictorial and graphic forms more readily assimilated by people. The declining costs of personal computers with interactive graphics also opened up a wide range of applications of simulation in training, education and business management. A major example of this is the Xerox Star based on the simulation of the office desk-top and working environment [25].

The Present Fifth Generation: 1980-1988

The previous section outlines how the focus of attention in simulation technology has developed through hardware, architecture, software and interaction to bring us to the present era. In the current fifth generation of computing technology we routinely use interactive graphic simulation systems, programmed in high-level simulation languages, and based on efficient high-speed processing architectures making effective use of vlsi devices. The focus of attention at the frontier of simulation technology has now switched to expert systems in response to development of knowledge-based systems technologies in the computer industry [12,23].

At first sight expert systems technology seems quite remote from that of simulation. Expert systems encode the processes of the human mind, knowledge representation and inference. Most simulation applications are concerned with physical variables that seem far more concrete, and certainly better understood, that the concepts of "knowledge" and "reasoning". There also appears to be a confrontation of paradigms involved:

(a) the classical approach to designing a system solving some problem is to identify the problem in detail, model it mathematically, and then develop an algorithm for its solution either analytically or through simulation;
(b) the expert system approach is to assume that the problem is not identifiable in detail but that heuristic solutions to it exist and are exemplified in the skills of experts who are known to be able to solve it, and to model the behavior of the expert in order to develop an algorithm.

This is a substantial shift in problem solving techniques that represents a fundamental change in scientific paradigms. The success of expert system technology in solving previously intractable problems in industrial control, medicine, geology, and so on, has created widespread interest in the technology and acceptance of the new paradigm [22]. It has many significant immediate applications. However, there are many links between the technologies of simulation and those of expert systems, and possibilities for even more significant applications if the two can be blended effectively together [12]. In particular we need to combine the *deep* knowledge of scientific principles used in simulation with the superficial *what-to-do* knowledge used in expert systems [8]. The linguistic simulation of imprecise systems is now feasible [29]. The production rule languages used in expert systems can be applied directly in symbolic and numeric simulation [2]. Simulation and expert system techniques can be combined in integrated advisory systems [8,10,21,31].

The Future Sixth and Seventh Generation: 1988-2004

The breakthrough that distinguishes the coming sixth generation of computing technology has been advances in inductive inference such as Lenat's AM [4] and Michalski's INDUCE [19]. In the current fifth generation era machine learning has become a highly active research area still in its replication phase [18]. In the coming sixth generation era we may expect related product innovation affecting simulation, such as the use of inductive inference to allow real-time learning simulations to be developed as major new modeling tools. Seventh generation projections are more speculative but it is reasonable to suppose that such learning systems will form the foundation for autonomous robot technologies, in which knowledge acquisition and simulation will be combined to support goal-driven systems. However, to achieve significant results with such systems we need to add perceptual acts to the planning structures of a language such as SIPE [32] and develop logic programming languages that cope with the resulting temporal logic [1]. In such developments the sixth generation breakthrough will come to be recognized, and in all of them the knowledge of how to handle modeling, optimization and temporality in simulation studies has a major role to play [27].

7 CONCLUSIONS

Underlying all developments in computing is a succession of learning curves which make up the infrastructure of the computer industry. The base curve is that for electronic device technology and above it in succession are those for: virtual machine architectures; problem-orientated languages; human-computer interaction; knowledge-based systems; inductive inference systems; autonomous activity systems; and socially organized systems. The phases of the learning curves at each level in the infrastructure occur in eight year generations corresponding to the business cycle of capital equipment purchasing. In each generation the types of breakthrough, research focus, product innovation, and established product lines, may be predicted from the levels of technologies in particular phases. This gives a satisfactory account of past surprises in invention, fashions in research, innovations in products and launches of product lines. It also gives a reasonable basis for forecasting future trends in the fifth, sixth and seventh generations.

The focus of attention of simulation studies has shifted to each new technological innovation in the infrastructure of the computing industry through hardware, architecture, software, interaction and, in the present fifth generation era, expert systems. Attention is focused on expert systems currently because they are a recent development involving a change in paradigm that has proved effective in industrial applications.

There are many benefits to be gained from the application of expert systems to problems where the classical approach of modeling the problem and computing an optimum solution is not possible and where, instead, it is possible to generate a solution by modeling a person solving the problem. This allows the automation of tasks that were previously beyond computer implementation because of our lack of in-depth understanding of the processes involved. However, there are even greater benefits to be gained through combination of simulation and expert system techniques in integrated systems. As we move into the sixth and seventh generation technologies of automatic knowledge acquisition and autonomy, the techniques of modeling, optimization and temporal representation developed in simulation studies will have a major role to play

8 ACKNOWLEDGEMENTS

Financial assistance for this work has been made available by the National Sciences and Engineering Research Council of Canada.

9 REFERENCES

[1] Allen, J.F. (1985). Towards a general theory of action and time. **Artificial Intelligence, 23**(2), 123-154 (July).

[2] Bruno, G., Elia, A. & Laface, P. (1985). A rule-based system for production scheduling. **Proceedings COMPINT'85, IEEE 85CH2136-O**, 35-39 (September). Montreal.

[3] Crane, D. (1972). **Invisible Colleges: Diffusion of Knowledge in Scientific Communities**. Chicago, Illinois: University of Chicago Press.

[4] Davis, R. & Lenat, D.B. (1982). **Knowledge-Based Systems in Artificial Intelligence**. New York: McGraw-Hill.

[5] Gaines, B.R. (1968). Varieties of computers—their applications and inter-relationships. **Proceedings of IFAC Symposium on Pulse-Rate and Pulse Number Signals in Automatic Control**, 1-16.

[6] Gaines, B.R. (1984a). A framework for the fifth generation. **Proceedings of National Computer Conference, 53**, 453-459. Arlington, Virginia: AFIPS Press.

[7] Gaines, B.R. (1984b). Perspectives on fifth generation computing. **Oxford Surveys in Information Technology, 1**, 1-53.

[8] Gaines, B.R. (1986a). Expert systems and simulation in industrial applications. Luker, P.A. & Adelsberger, H.H., Eds. **Intelligent Simulation Environments**, pp. 144-149 (January). La Jolla, California: Society for Computer Simulation.

[9] Gaines, B.R. (1986b). Sixth generation computing: a conspectus of the Japanese proposals. **ACM SIGART Newsletter**, No.95, 39-44 (January).

[10] Gaines, B.R. (1986c). The design of expert systems for planning flexible manufacturing. **Proceedings Japan-USA Symposium on Flexible Automation**, Osaka (July).

[11] Gaines, B.R. & Joyce, P.L. (1967). Phase computers. **Proceedings of the 5th Congress of the International Association for Analog Computation**, 48-57.

[12] Gaines, B.R. & Shaw, M.L.G. (1985a). Expert systems and simulation. Birtwistle, G., Ed. **AI, Graphics and Simulation**, 95-101 (January). La Jolla, California: Society for Computer Simulation.

[13] Gaines, B.R. & Shaw, M.L.G. (1985b). The infrastructure of fifth generation computing. **Proceedings of COMPINT'85**, IEEE 85CH2136-0, 747-751 (September).

[14] Korn, G.A. & Korn, T.M. (1964). **Electronic Analog and Hybrid Computers**. New York: McGraw-Hill.

[15] Lebedev, S.A. (Ed.) (1960). **Computer Engineering**. Oxford: Pergamon Press.

[16] Marchetti, C. (1981). Society as a learning system: discovery, invention and innovation cycles revisited. **Report RR-81-29** (November). Laxenburg, Austria: International Institute for Applied Systems Analysis.

[17] Mayorov, F.V. & Chu, Y. (1964). **Electronic Digital Integrating Computers—Digital Differential Analyzers**. London: Iliffe.

[18] Michalski, R.S. & Carbonell, J.G., Eds. (1983). **Machine Learning: An Artificial Intelligence Approach**. Palo Alto, California: Tioga.

[19] Michalski, R.S. & Chilausky, R.L. (1980). Knowledge acquisition by encoding expert rules versus computer induction from examples—A case study involving soyabean pathology. **International Journal of Man-Machine Studies, 12**, 63-87.

[20] Moto-oka, T., Ed. (1982). **Fifth Generation Computer Systems**. Amsterdam: North-Holland.

[21] Oren, T.I. (1986). Knowledge bases for an advanced simulation environment. Luker, P.A. & Adelsberger, H.H., Eds. **Intelligent Simulation Environments**, pp. 16-22 (January). La Jolla, California: Society for Computer Simulation.

[22] Reitman, W., Ed. (1984). **Artificial Intelligence Applications for Business**. Norwood, New Jersey: Ablex.

[23] Shannon, R.E., Mayer, R. & Adelsberger, H.H. (1985). Expert systems and simulation. **Simulation, 44**(6), 275-284 (June).

[24] Sizer, T.R.H. (Ed.) (1968). **The Digital Differential Analyser**. London: Chapman & Hall.

[25] Smith, D.C., Irby, C., Kimball, R., Verplank, B. & Harslem, E. (1983). Designing the Star user interface. Degano, P. & Sandewall, E., Eds. **Integrated Interactive Computing Systems**. pp. 297-313. Amsterdam: North-Holland.

[26] Smith, G.W & Wood, R.C. (1959). **Principles of Analog Computation**. New York: McGraw-Hill.

[27] Unger, B, Dewar, A., Cleary, J. & Birtwistle, G. (1986). A distributed software prototyping and simulation environment: JADE. Luker, P.A. & Adelsberger, H.H., Eds. **Intelligent Simulation Environments**, pp. 63-71 (January). La Jolla, California: Society for Computer Simulation.

[28] Van Dujin, J.J. (1983). **The Long Wave in Economic Life**. London: George Allen & Unwin.

[29] Wenstop, F. (1981). Deductive verbal models of organizations. Mamdani, E.H. & Gaines, B.R., Eds. **Fuzzy Reasoning and its Applications**. pp. 149-167. London: Academic Press.

[30] Wexelblat, R.L. (Ed.) (1981). **History of Programming Languages**. New York: Academic Press.

[31] Wichmann, K.E. & Lenz, J. (1986). Integration of a Fortran-based simulation program and a Prolog-based expert system for intelligent manufacturing systems simulation simulation. **Conference on Intelligent Simulation Environments** (January). La Jolla, California: Society for Computer Simulation.

[32] Wilkins, D.E.. (1984). Domain-independent planning: representation and plan generation. **Artificial Intelligence**, 269-301.