SOME REMINISCENCES ON THE HISTORY OF HARDWARE AND SOFTWARE FOR SIMULATION 1963-2003

RICHARD N ZOBEL

Department of Computer Science
The University of Manchester
Oxford Road, Manchester M13 9PL, United Kingdom
rzobel@cs.man.ac.uk, r.zobel@ntlworld.com

Abstract: During the past fifty years the world has changed and simulation has advanced out of all recognition. The author has progressed from very large analogue computers in the early 1960s, through hybrid analogue/digital systems in the late 60s, to design and construction of digital signal processing computers in the 70s, onto early microcomputers in the 80s, used for simulation, and more recently to distributed simulation in the 90s and 00s, always up with the leading edge. This paper summarises some of these developments and considers the current situation and future prospects.

Keywords: History, Hardware, Software, Languages, Applications, Parallel and Distributed, Virtual Environments.

INTRODUCTION

I still have some vacuum tubes (valves) and some early metal can transistors with serial numbers! How times have moved on. My introduction to simulation began with a simulation of the Mark II Seaslug surface to air missile which of course required the parallel solution of many simultaneous non-linear, time-varying, ordinary differential equations. My PhD concerned the design of a new hybrid analogue multiplier and some fast (for those days) analogue to digital converters. At this point, I had my first experience of simulation societies (UKSC) and organised a conference at Manchester in 1970. A successful excursion into digital signal processing and design of signal processing computers was terminated by the arrival of the microprocessor. Fortunately, John Stephenson (Bradford) steered me back to simulation and in 1985 I attended my first SCS conference in Reno, Nevada, USA [1]. Since then simulation has enabled me to take part in many organised events in over 30 countries, making in the process, many friends both nationally and internationally, and in the process enjoying myself immensely.

THE EARLY DAYS OF SIMULATION USING ANALOG COMPUTING

Lord Kelvin first outlined the idea of a differential analyser, using mechanical ball and disc integrators, in 1876. However, the lack of contemporary technology prevented the idea from being realised. The earliest implementation was by Prof. Vannevar Bush of M.I.T. in 1930. Prof. Douglas Hartree, FRS, of St Johns College, Cambridge, with his assistant Arthur Porter, a Manchester undergraduate, produced a Meccano machine in 1934. This was followed up in 1935 by a full scale machine by Hartree and Porter with engineering support by Metropolitan Vickers and funding by Sir Robert McDougall. It was for many years recognised as the best and most used machine for gunnery prediction and other applications [2,3].

By the end of the Second World War, vacuum tube based operational amplifiers were beginning to be used to construct an electronic equivalent for solving differential equations. That technology led to the development of the general purpose analogue computer, used mostly at that stage for research, development and performance assessment of military systems.

Fundamentals

Systems of ordinary differential equations, representing the mathematical models of real dynamic systems, may be solved using a small subset of mathematical operations directly or by means of combinations of these with some additional electronic circuits.

Figure 1 illustrates the basic uses of operational amplifiers for addition and subtraction of voltages and of the integration of the sum of voltages. For the examples given, the adder has an output voltage given by:

$$-V_0 = V1.Rf/R1 + V2.Rf/R2$$
$$= k1V1 + k2V2$$

Proceedings 17th European Simulation Multiconference
(c) SCS Europe BVBA, 2003 ISBN 3-936150-25-7

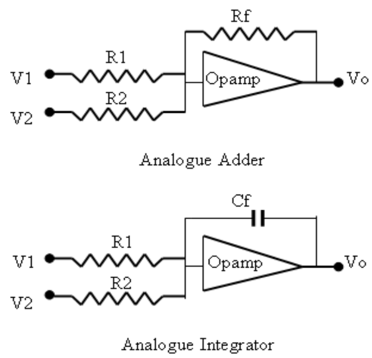

Fig. 1 Basic analogue computing elements

The integrator has an output voltage given by:

$$-V_0 = \int_o^t (V1/CfR1 + V1/CfR2)dt$$

$$-V_0 = \int_o^t (k1V_1 + k2V_2)dt$$

Voltages represent scaled equation variables. Variables may be negated using an inverter (an adder with only one input and Rf = R1).

Other required mathematical operation are:

$V_0 = kV1.V2$ (Multiplication)

$V_0 = f(V1)$ (Function Generation)

Multiplication may be achieved using a precision (10 turn) potentiometer (for fixed coefficients) or a time division multiplier, or quarter squares multiplier ($A.B = 0.25\{(A+B)^2 - (A-B)^2\}$, the squaring being achieved by a fixed diode/resister network. The latter was also used for non-linear function generation, but with adjustable segments.

Figure 2 is a reproduction of a photograph of the Solartron 247 system at Sperry Gyroscope Company, Bracknell, U.K., circa 1964.

There are 15 racks of mainly valve based operational amplifiers and other functional units. Clearly shown are 5 patch panels, where the units were interconnected to form the analogue network for solving the current problem. Each rack consumed around 1kW of power. The system cooling can clearly be seen above the units. The addition of a second machine of 12 racks yielded a total capability for solving problems requiring the resulting 1000 operational amplifiers. The operator (the author) is sitting in front of one of the control desks, setting coefficients on one of the potentiometers.

Analogue computers are parallel computers, solving sets of simultaneous differential algebraic equations (DAEs) in real time (or scaled faster/slower than real time). However, unlike modern digital computers, analogue hardware cannot be used to quasi-simultaneously compute more than one function as with a digital processor, because it is not possible to share variables on a single anlogue hardware unit due to lack of an equivalent storage mechanism and the near impossibility of remembering and then restoring the correct charges on the integrator capacitors with such large values of the order of .001μF to 1 μF, required for precision integration with low drift.

Accuracy, Repeatability and Correctness

There are limits to the accuracy of analogue systems, due to the physical and economic costs of producing precision resistors and capacitors. There is also the problem of variation with temperature and time. Further, as the resolution is increased, noise arising from many sources limits further increases in accuracy. The repeatability of results for a particular simulation run is an important issue and is determined by the combined accuracy and stability of the parallel analogue computing units. It was difficult to achieve better than around 0.1% in practice, due to the combination of errors due to the 0.01% tolerance on many components.

Test runs, duplicated on a KDF9 digital computer, exhibited excellent resolution and repeatability. However, the correctness claimed was somewhat misguided. The problems associated with integration algorithms, sample rates and sparse matrices had yet to be appreciated. The latter appeared as a very large ratio between long time constants associated with the physical dimensions of the vehicle and the short time constants associated with the required accelerations. Over a period of time the analogue and digital test runs got closer and we all learned a lot about our respective disciplines and to respect each others professional expertise and experience!

Valves, Transistors, Hybrids and Logic

The amplifiers were constructed using valves running on supplies of ± 300 volts, with reference voltages of ± 100 volts used for constants and initial conditions. Mechanical choppers, used for d.c. offset stabilisation, were replaced by semiconductor devices with greater reliability. The time division multipliers employed germanium transistors for switching, which limited the ambient temperature for accurate operation (germanium melts at around 55 Celsius, hence the air conditioning requirement for both machine and operators. A few voltage comparators and simple logic gates provided for some useful logical operations. Examples of the use of these included the detection of range coincidence between missile and target, transfer from boost phase to guidance phase, and of out of scale variables.

THE EMERGENCE OF DIGITAL SIMULATION

As digital computers became faster and cheaper their advantages became obvious, and the demise of the traditional analogue computer became inevitable. However, there was still much to learn about digital simulation. Perhaps the most obvious problem was the loss of “feel” and the immediacy of the analogue system. The real-time interaction was much better than a 24 hour turn-around and the numerical teletype printout for the digital simulation runs.

Processing Power and Memory

Initially, the processing power and memory limitations of the digital computers of the 1960s and 70s severely limited their use. However, the appearance of the microprocessor and advances in both random and serial access memories began to change this. Progress has of course been and continues to be rapid, leading to the current substantial power of both PC and Workstation.

Digital Equivalents of Analogue Functions

Initially, most attempts concerned the reproduction of analogue techniques using numerical equivalents. Some were quite successful, especially in the area of digital equivalents of analogue filters. A particular concern was the appearance of problems associated with word length and recursion convergence. Those of us with knowledge of both continuous and discrete (sampled) systems equated word length with gain, recursion with feedback, and convergence with stability.

Problems of Integration

The analogue integrator is a true integrator, admittedly with imperfections at both low and high frequency. However, sampled data approximations are fraught with other problems. It has taken many years to recognise that one integrator algorithm is not the answer to solving all differential equation systems. There is a black art (aka experience) in knowing which to use in a particular circumstance.

Speed

In the early days of analogue computers, digital computers were very slow compared to their analogue counterparts. This was partly due to the parallel nature of the latter. Thus the concept of real-time simulation was restricted to the analogue machine. The requirement for real-time simulation arose from two different requirements. The first was

Fig. 2 Solartron 247 Analogue Computer System

the need for including either humans or real hardware or both in the loop (time constants are fixed) and the second was the need for large numbers of simulation runs to be completed in a reasonable time (not just a lot too late). The first requirement remains in respect of training simulators. Modern digital systems employ multiple processors for different activities (such as input/output, graphics, sound, etc), parallel processors for partial differential equations or large numbers of ordinary differential equations, and networked systems for large training simulation exercises.

LANGUAGES

Those of us using simulation are accustomed to using simulation packages for speed and convenience. Newcomers however are often ignorant of the existence of such packages. In the early days only Fortran and machine code were available.

General Purpose Languages

Simulations have been and continue to be written, often inappropriately, in general purpose programming languages. Concerns over appropriateness, convenience, speed of execution and compatibility with other software tools such as databases and graphics are often ignored to the disadvantage of such inexperienced users. For simple simulations this probably does not matter, but for real, larger problems it does, as they are usually inappropriate. However, until simulation is taught as a general purpose tool, this is likely to be a problem that will persist for some time to come.

Simulation Languages

As the use of simulation has developed, increasingly more sophisticated packages have been developed, taking advantage of the increasing power and utility of computers. ACSL, ESL, Simplex 3 and Arena are examples of the many systems currently on offer supporting not only model development and simulation development, but also all of the tools necessary to support simulation studies for continuous system, discrete event or mixed applications.

Equation Solving

Historically, model equations had to be manipulated by hand before being programmed into a simulation language. Modern simulation systems do this automatically, using functional programming techniques.

Library Functions

Simulation specific techniques and models are now provided via libraries including mathematical functions such as integration, function generation, axis transformation, dead space, logic functions, queues, stacks, etc., supported with graphical model building using icons and interconnections.

GRAPHICS

Graphics did not arrive until the 1970s. Before this, pen recorders, numerical results tables and character based graphs printed on teletypes were the order of the day. The arrival of graphical

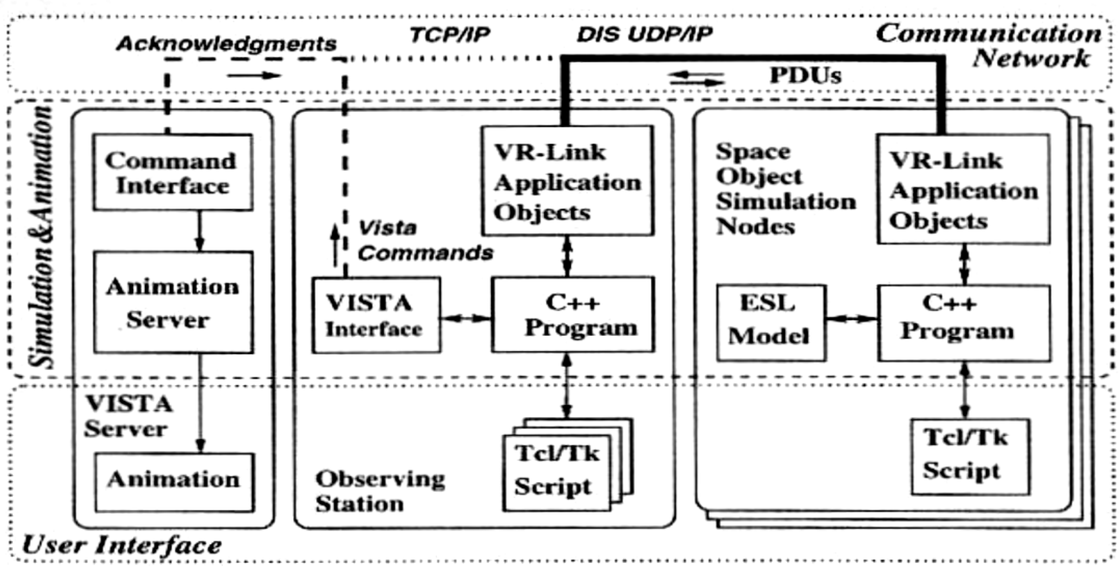

Fig. 3 DIS System Linking Distributed Simulations and Animator

screens, used almost universally now, allowed block diagrams of models and systems, full multi-channel graph systems, animation and virtual reality to develop.

COMPUTERS

Early digital computers were used for simulations, but they were slow, and integration and algebraic loops caused problems. However, the arrival of the microprocessor, lead to a revolution in simulation applications of increasing variety and complexity. The earliest personal computers were used experimentally for simulation and their potential was obvious in terms of the re-acquired feel and fast turn around for the systems being simulated.

Main-Frames, Minicomputers and Microcomputers

Mainframes were difficult to use because they were distant and any resemblance of the hands-on approach was lost, particularly in relation to the delay between submitting a program to the computer centre and receiving a print out (or list of programming or run time errors) of at least 24 hours. Graphical printout was inconveniently character based.

Minicomputers, such as the PDP11, were a significant advance at lower cost and allowed a more hands on approach. The performance limited applications to those which fitted the RAM and disk parameters and fell short of real-time for many applications.

Early microcomputers, based on the Intel 8088, were too slow and too small, but gave an insight to their potential as personal computers (PC). However, the rapid advances in speed, power and memory gave rise to desktop machines of significant use for many simulation applications. Of course, this still left the big problems to the much larger mainframes and computer centres.

PARALLEL COMPUTERS AND NETWORKED COMPUTERS

With the increasing reduction in the costs of processors and memory, and the improving application of memory management systems, the rapid development of parallel computers came about, leading to the possibility of tackling large scale simulations of such problems as weather forecasting and stress analysis in addition to the larger ordinary differential equation problems with large numbers of equations. The rapid advances in networking also lead to the development of distributed simulations, interconnected over private or public networks especially for training.

DISTRIBUTED SIMULATIONS AND VIRTUAL/ SYNTHETIC ENVIRONMENTS

Training simulators have been an important part of military systems provision for many years. Distributed interactive simulation came into being first with SIMNET an attempt to connect military simulators together. Following this the DIS system for interconnecting training simulators together successfully gave rise to the IEEE 1278 standard. The author and one of his PhD students, was involved in a project concerning space missions. The systems comprised networked Unix workstations, each running a spacecraft simulation, interconnected by VR-Link to an animator (VISTA) shown in figure 3 [4]. In this case, the space objects were a space shuttle and a space station with an observing station on the ground. The motions of these objects were simulated using ESL, and visualised and animated using VISTA, the latter two packages being provided by iSIM and ESTEC respectively. The background of the Earth, Sun, Moon, Planets and Stars were also provided by ESTEC.

Following this, the high level architecture (HLA) provided standards and a run time interface (RTI) for management of a federation of a network of simulation federates. A consequence of this for non-military applications is the need for provision of security for both obtaining access to an HLA federation and for authentication and passing of data and messages. The latter is discussed in detail in a technical and tutorial paper by the author [5].

SUPERCOMPUTERS AND GRIDS

In recent years, supercomputers, comprising of upwards of a 1000 processors with massive local and global memory have been developed and used for solving large problems. Perhaps the best known of these are the weather forecasting machines in Bracknell, UK and in Washington, USA. These machines basically solve the Navier-Stokes partial differential equation (PDE) for the global atmosphere for weather prediction. Other PDEs are solved for dynamic stress analysis, fluid dynamics problems and atomic reactions.

Recently, projects have evolved to link such supercomputers together to solve even larger problems as parallel/distributed architectures, or to obtain finer grain solutions, by using grids of supercomputers [6,7]. Although an excellent concept, much work is being done on algorithms and optimisation as was done for parallel computing.

THE FUTURE

The increasing power of individual and networked computers, coupled with the universal availability of powerful graphics and virtual reality is making simulation a universally useful tool set. The recognition of the validity of discrete event, continuous and mixed simulations as a normal part of project planning, design, implementation, testing and training in terms of engineering, management and commercial activities is of great significance. Consequently, new areas of application of simulation are emerging. Of particular importance, are those connected with software agents [8], distance learning and web-based applications [5].

CONCLUSIONS

It is clear that simulation technology and applications now pervade almost every human activity and endeavour. As the tools become more widely used and integrated, and the use of simulation is taught more widely as a valuable addition to applied mathematics, the acceptance and future of simulation seems bright. The history of simulation is richly sprinkled with the names of stars in every subject. It probably starts centuries ago, perhaps with Leonardo da Vinci or earlier, but certainly the work of Lord Kelvin, and of Hartree and Porter gave simulation a major kick start. The foundation of the Society for Computer Simulation (SCS) in 1952, only 4 years after the first stored program digital computer was demonstrated at Manchester was an event of major significance. The establishment of the European Simulation Societies in Europe around 35 years ago has also contributed to its successful development. The author is grateful to have had the opportunity of taking part in this activity and of the consequences for his international travel experiences and his many friends gathered over the years.

ACKNOWLEDGEMENTS

The author would like to acknowledge all of the contributions of his colleagues and postgraduate students in the Dept. of Computer Science at Manchester, in SCS International, the Board and Member Societies of EUROSIM, in iSIM and ESTEC, and all of his colleagues and friends worldwide who have in various ways contributed to a fascinating career mainly concerned with simulation and its areas of application.

REFERENCES

[1] Adib D. and Zobel R.N. 'Simulation of uHARP, A Multi-Microprocessor Based Hierarchical Array Processor for Signal Processing.' In *Proc. SCSC 86, Summer Computer Simulation Conf.*, Reno, Nevada, USA, July 1986, pp 965-971. SCS, San Diego.

[2] Hartree D.R. and Porter A. 'The Construction and Operation of a model Differential Analyser' In Memoirs and Proc. of the Manchester Literary and Philosophical Soc. 1934-45 Vol. LXXIX pp.51-81

[3] Hartree D.R. and Porter A. 'The Application of the Differential Analyser to Transients on a Distortionless Transmission Line'. Jnl. Institution of Electrical Engineers. Vol. 83, No. 503, Nov. 1938

[4] Tandayya P. PhD thesis, 2000, Manchester University. p151.

[5] Zobel R.N. 'Security for Internet and Web-based Applications' Invited paper in 4th Int. Conf. on Information Integration and Web-Based Applications and Services (iiWAS2002), Sept. 2002, Bandung, West Java, Indonesia. ISBN 3-936150-18-4. pp. 6-16.

[6] Foster I., Kesselman C., Tuecke S. 'The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration'. Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.

[7] Foster I., Kesselman C., Tuecke S. 'The Anatomy of the Grid: Enabling Scalable Virtual Organizations'. Int. Jnl. Supercomputer Applications, 15(3), 2001.

[8] Iwu O.F. and Zobel R.N. 'Network Attack Profiling: Using Agent Based Simulation to Gather Forensic Information'. In Proc. 3rd Workshop: Agent-Based Simulation, Passau, Germany. Apr. 2002. SCS Erlangen,. ISBN 3-936150-17-6, pp180-186.

ABOUT THE AUTHOR

RICHARD ZOBEL graduated in Electrical Engineering from London University in 1963. His first experience of simulation was obtained during 1962-66 at Sperry Gyroscope whilst working on military projects, using valve analog computers. His Ph.D., obtained in 1970 at Manchester, concerned hybrid analog-digital computing. As Lecturer and Senior Lecturer he became involved in digital signal processing, instrumentation and design environments with special emphasis on the simulation aspects of real-time embedded systems. He is a former Chairman of the United Kingdom Simulation Society (UKSim), Former Secretary of the European Federation of Simulation Societies (EUROSIM), and was a European Director of SCSI, the Society for Computer Simulation International. His current research work concerns distributed simulation for non-military applications, model re-use, distributed simulation model databases, issues of verification and validation of re-useable simulation models and security for distributed simulation under commercial network protocols. He is now semi-retired, but still very active, both in teaching and research.