A PERSONAL HISTORY OF SIMULATION IN THE UK AND EUROPE, 1964-2001

RICHARD N. ZOBEL

Department of Computer Science
University of Manchester
Oxford Road
Manchester M13 9PL, U.K.
Email: rzobel@cs.man.ac.uk

Abstract: Simulation has come a long way since the author started at Sperry Gyroscope Co. at its site at Bracknell, Berkshire, England in 1964. Then the main activity was military simulation using valve analog simulators. Following a move to a lectureship at Manchester University in 1966, a progression from analog to digital simulation via hybrid analog and digital systems (the latter being the topic of the authors' PhD thesis), was followed by an incursion to digital signal processing and the design of several digital computers and back to simulation. The paper follows the authors frequent involvement in both simulation and the activities of the UK Society, the European Societies, the formation of Eurosim and with the American and European branches of the Society of Computer Simulation. The paper also traces the authors' involvement in various developments in technology, software and applications of simulation, and in particular with increasing international involvement with simulation conference management. These activities have brought with them a great deal of friendship, happiness, foreign travel and satisfaction. May this continue for years to come.

THE EARLY DAYS

My memories of working on the huge 2000 operational amplifier, 5000 valve, So!artron 247 analog computer are both rewarding, frustrating and satisfying. On a bad day, the damn thing refused to work properly. On a good day, when all 5000 valves worked at the same time, we successfully simulated 300 missile firings! There was the day when Maurice Sly thought that a fuse was blown and it was just one of my colleagues stamping on a nearly empty inverted plastic coffee up, after Tinker Barnes had previously demonstrated the new coffee machine (to replace the
tea lady) which poured hot water, coffee powder and sugar into a missing cup. Both Maurice and Tinker contributed much to the success of the simulator section in addition to occasional falling foul of our misguided humour.

Perhaps the most amusing incident concerned the overheating of this vast analogue computer after the second tranch of equipment arrived. The total electrical power being consumed was around 27kw, all exhausted into a modest sized room, with high partitioning, but open to the high roof of the building. At first, all was well but then it was winter. As Spring came and went and Summer began the temperature of the machine, the lab and the staff steadily rose until in the afternoon we sat in front of a machine that was becoming increasingly inaccurate because of the geranium transistors (my spelling!) in the time division multipliers, sweating profusely and unable to think properly. we went to see the management demanding air conditioning. The initial response was "Air conditioning - I do not have air conditioning in my office and I am the Director of the section"!

After visiting the laboratory, the management were convinced that there was a real problem and air conditioning was ordered. Of course, this needed designing, manufacturing, installing and testing. However, the conditions were unacceptable and the machine overheating. After some discussions, mainly over pay, it was agreed that until the air conditioning was installed the simulator laboratory staff would work from 3 am. until II am., starting the following Monday. I duly got up at 2.30 am. and left sleepy eyed at about 2.50 a.m. on my bicycle.
"'eIlo, 'ello' said the police constable, "and where do you think you're going at this time of the morning?"
"To work" I said.
"And where would that be, Sir?" he said.
"Sperry Gyroscope Company' I replied.
"At this time of the morning?" he asked.

 ISSN 1473-804x online, 1473-8031 print

I explained the situation with some difficulty.
"Have you any identification?" he asked.
I thought for a moment and then decided that the only way to convince him was to show him my security pass, which seemed to convince him (you were not supposed to do this).
After this it was "Good morning Sir", "Good morning Officer", and all was well.
Eventually the air conditioning arrived and the episode ended.

Then there was the day that there was the ever increasing smell of a hot transformer. Not our problem, except that after lunch, we discovered that it was one of our mains regulators, which was by then in flames, with the simulator still operating normally! Then there was the Mark I electronic unit, obtained from the production line to assist with a post firing simulation, which was connected using a wiring list which was backwards leading to an unprotected ac supply from a Jones plug finding an electrolytic capacitor at the far end of the main cable harness, leading to the inevitable fire! and then

MIXED ANALOG/DIGITAL SYSTEMS

The nature of many systems being simulated from the 1970's onwards, required decision making and hence analog computers acquired some logic devices with comparators to provide the conditions for changes to occur. The latter were then, as now determined by timing, current state and inputs. However, the finite state machine still at that time remained in the realm of the emerging digital computer. At the lower level, new hybrid computing systems were being developed along with the possibilities of eliminating the analog patch-board with its "mad woman's knitting" [Zobel 1970]. At this time there were also many arguments about the errors in analogue computers versus the correctness and repeatability of the new digital machines. Of course the digital machines were more repeatable, but much slower. However, the accuracy debate continued for many years until the intricacies of numerical integration and stiff systems were better understood.

DIGITAL SIGNAL PROCESSING

After an incursion into hybrid computing, I became involved in the emerging area of digital signal processing, leading to the construction of MOSAIC, a Modular On-hue Signal and Instrumentation Computer in conjunction with Data Laboratories Ltd. This was based on TTL technology with a 74180 4-bit based ALU as the most complex IC in the system! It computed a 1024 point FVF around 10 times faster than the PDP1 1, a current leader at that time. This was because of the availability of 16 user registers which permitted the butterfly inner arithmetic loop as register to register arithmetic, a feature which RISC processors adopted many years later. In addition, there was a hardware sine/cosine lookup table and a custom FFT address generator. Of course, the only system software was a monitor rather than an operating system and the machine was programmed using a home made assembler. The memory was 4k*16 bits core, there was no disc and input/output was by means of a te!etype and paper tape. Of course, this was in the early 70's.

Subsequently, a 4 processor (Intel 8086) parallel machine was built by the author, which was based on semiconductor memory associated with each processor. A parallel FFT was achieved by designing a "move" processor which enabled either source or destination addresses to be sequential or bit reversed. The performance was quite impressive for its time. Figure 1 shows the architecture. It is noteworthy that the backing store was a pair of 8" floppy disc drives. The performance was simulated and verified against the real data, as shown in the paper presented at the Reno SCSC [Adib and Zobel, 1986].

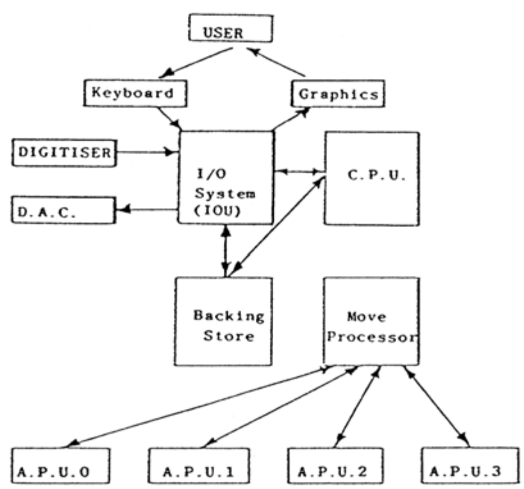


Fig. 1. MicroHARP System Architecture

Later, desktop computers arrived in the form of the Sirius machine, based on the Intel 8088 8-bit processor, with *5.25"* variable speed floppy drives. be designed in only a few minutes. To this was added IIR filters, FFF, and other DSP algorithms, leading to the development of the system shown in figure, 2 [Zobel 19881.

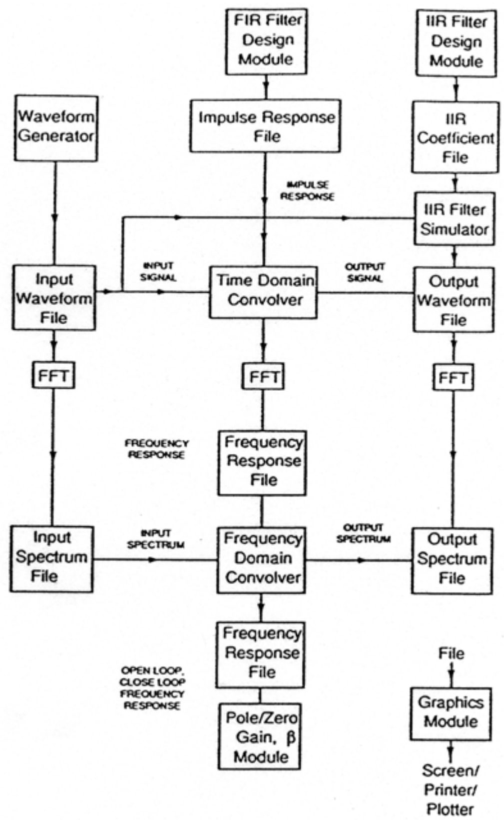


Fig. 2. DSP Software Module Block Diagram

SIMULATION MODEL DATABASES

Now we come to the first of the modern problems. It is essential for the future of simulation that practitioners are able to construct verified and validated simulations in an ever shortening time scale. Engineers have long constructed prototypes and preproduction products from a kit of parts and subsystems. At long last, software engineers have gone down this path and have developed object oriented programming systems (OOPS), object oriented analysis (OOA) and object oriented design (OOD). More recently, they have developed a platform independent OOPS, namely JAVA. This promises to bring to programming what 051 brought to networked hardware and software systems. In simulation we still build systems out of models of parts and subsystems, but fail to provide effective mechanisms for storage and re-
This enabled Finite Impulse Response (FIR) Filters to use of models previously constructed for particular projects.

There are of course exceptions to this. The IC industry uses libraries of basic gates and flip-flops, from which subsystems such as adders, rnultiplexers, counters, buses, etc are constructed. Within the very limited number of basic types, libraries of verified and validated (within both functional and timing aspects) re-usable logic modules enable designers to construct very complex logic systems, e.g. microprocessors, whose simulation is so accurate that they (mainly) work first time after simulation has been used to verify functional and operational speed design parameters.

Further, some engineering industries, notably hydraulic, nuclear and control, engineers have constructed libraries of re-usable simulation models for their specialist interests. Unfortunately, modern systems tend to be multidisciplinary. Take for example, aircraft, video cameras, refineries and supply chains. Each relies on complex interactions involving several disciplines, compounded by a wonderful admixture of continuous and discrete event systems and controls.

Simulation is essential for system and design verification, manufacturing, and operational predictions and scheduling, plus provision for training staff in safe, economic operational procedures.

Ideally, the required simulations should be constructed using a database of re-usable simulation models of the required parts and subsystems (objects), gleaned from previous projects. Regretably we do not place such models in a database for re-use, except in a haphazard manner with incomplete data.

Current and previous work at Manchester first concerns determination of appropriate selection of a database management system. This started as Postgres, an extension of the lngres relational database systems with additional features to handle large binary objects, etc., and progressed through a variety of transient database management systems to 02 and currently to POET, the latter being object oriented database management systems (OODBMS). Following this are considerations of what constitutes the essential components of a simulation model. Methodologies for classification of models, submodels and library models follows, leading to the first tentative construction of a database using models from a variety of different applications, continuous, discrete event and mixed. An early example taken from Zobel [Zobel, 1993] shows some possible classifications in terms of science and engineering (figure 3a) and at a more detailed level, in signal processing (figure 3b). These would now be considered to be rather naive and simplistic.

Of course, re-use itself is a topic requiring much study. Seldom are models of the right complexity and specificity for a new application available. This leads to construction from simple library models which is time consuming or from editing existing models. Both result in new models which require further time consuming verification and validation. The issue of verification and validation obtained through the reuse of models constructed from verified and validated components, library functions and subsystems is a currently burning issue requiring attention.

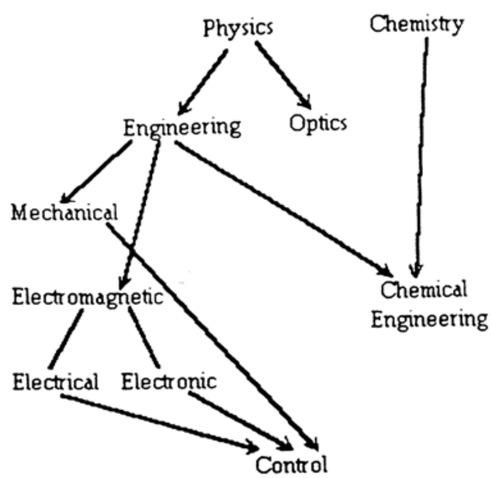


Fig. 3a. Possible Classification for Engineering Disciplines

Some early work on this aspect, related to a discrete event model of the handling of files in a print queue and subsequent printing, is discussed by Han [Han, 2000]. This work considered the complex model split into a set of submodels, each of which was considered as either a detailed model or a simplified model. All combinations of these models were evaluated in respect of their verification and validation Some of these combinations were inappropriate, but most gave a satisfactory performance, allowing re-use of a variety of models to suit the requirements of specific simulation studies in relation to depth of modelling of each subsystem.

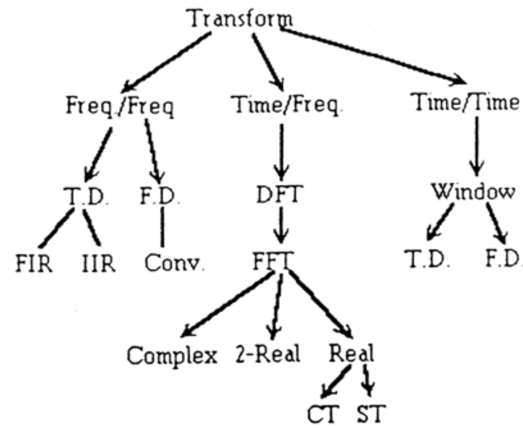


Fig. 3b. Early Object Classification of Digital Signal Processing (DSP) Simulation Models

DISTRIBUTED SIMULATION

As the number of networked computers in organisations, particularly universities, has rapidly expanded, it became obvious that many machines were often inactive due to their users being at meetings, away from departments, at home during the night, lecturing etc. With appropriate software and permissions, it was clearly possible to run simulations distributed over many machines, giving good performance, but with some communication overhead as described by [Kucuk. 2000]. Kucuk presented, in his PhD thesis, a method for splitting some electrical circuit problems over a number of computers distributed over a local area network, showing how the solution time could be substantially reduced over that required for solution on a single instance of the same type of computer.

Further, the military had began to realise that considerable benefit could potentially be gained from linking training simulators together using networking, which had lead to Simnet. Subsequently, the Distributed Interactive Simulation (DIS) standard was evolved to provide better interaction and performance, leading to the public standard IEEE 1278.

At about this time the author became Chairman of UKSC (later to become UKSim). As a result, I took over from the previous Chairman, John Stephenson, of Bradford University. the task of assisting in the setting up of the UK Simulator

Advisory Group (UKSAG). This Group, largely made up from DTI, MOD and defence companies, advised the UK Government on training simulators associated with UK defence systems and military training products for overseas customers.

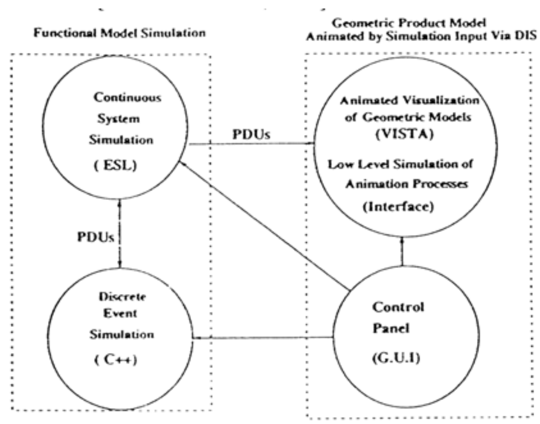
Through this organisation I learned about DIS and its applications and subsequently, about the High Level Architecture (HLA). My interest was mainly concerned with industrial applications of distributed simulation, leading to the work in manufacturing, space and driving simulations described below.

APPLICATIONS IN MANUFACTURING

There has been much work reported from academia and industry on virtual design and virtual manufacturing. After evaluating this work, it was decided to contribute in this area using DIS as a vehicle for connecting simulations together. Of course, some of the network protocol data units (PDUs), used for communication between simulations, are specific to military applications, e.g. the fire PDUs indicating that the vehicle has fired a weapon or has been hit by the fire from a weapon. Other PDUs, more relevant to industrial applications, needed to be defined.

The particular vehicle chosen for research study was a printer, in this case a dot matrix printer, was chosen because both a working and broken printer were available for study. The latter was dismantled to permit study, evaluation and modelling to be performed and models verified and validated against a working example.

A printer comprises a discrete event system in which files arrive from a computer or from a local area network (LAN) of computers, and put into a queue. The printer software system takes jobs from the print queue and sends them to be printed. The latter system is a mixed discrete event and continuous system, with the electromechanical parts represented by a continuous systems simulation. Such parts or subsystems comprise the print head and its feed belt and stepper-motor/gear system, the ribbon motor and drive, including the non-reverse gear mechanism which keeps the ribbon moving on when the head reverses direction on a carriage return-line feed action, and the carriage motor and mechanism for advancing the paper on line feed and page feed. A key feature here in figure 4 is the need to balance the network traffic with the inclusion of low-level simulation of animation processes, local to the animator [Habibi and Zobel, 1997].



Network Connections

Fig.4Integrating Networked Functional Simulation and Animated Visualisation

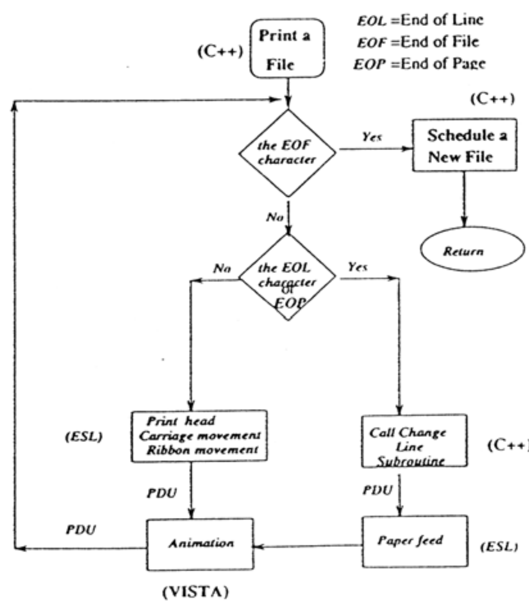


Fig.5 Structure of Integrated Simulation and Visualisation for Printer

Thus in this system, one computer, a SunOS based workstation, is used to simulate the discrete event system, implemented using C++, and a second computer, which implements the continuous system in ESL (the European Continuous Systems Simulation Language) uses a Solaris Workstation, whilst a third machine animates the printer simulation on a Silicon Graphics machine using the VISTA European Space Agency Animation System running under Irix. The communication uses TCP/IP on a LAN under the DIS protocol adapted for industrial applications. Figure 4 illustrates the general scenario, whilst figure 5 indicates the more detailed situation for the end-of line, end-of-page and end-of file situations.

APPLICATIONS IN SPACE SIMULATION

The space simulation concerned a ground station, a space shuttle ascending into orbit and an orbiting space station. The space shuttle, after orbital insertion, is required to match orbits with the space station and then to rendezvous and dock with the space station as describe by Tandayya [Tandayya and Zobel, 1998]. To achieve this, simulation of the ground observing station, situated on a rotating Earth, a space shuttle ascending into orbit and carrying out orbital manoeuvres and the space station are each simulated on different computers running ESL, the European Space Agency continuous systems simulation language, linked using DIS protocol data units (PDUs). These in turn are linked to the VISTA animator which animates the geometric models of the Earth, Space Shuttle and Space station, against a star field. For this purpose, Sun workstation Solaris Computers are used for the simulations and a Silicon Graphics Irix computer is running VISTA used for the animation.

The key element here is the VISTA animation software which permits animation of the geometric objects from remote sources, rather than from an internal generator associated with the animator.

An important issue is the DIS feature of a stealth viewer. DIS works by enabling each simulator to create a personal view from the position and attitude of the vehicle concerned using a remote terrain database for the background view, with all active local other vehicles superimposed as geometric figures, scaled for distance, at the correct relative position and attitude, and animated using the motion supplied via the PDUs from all of the other relevant players. This allows a non-participant observer to receive the PDUs and have access to the terrain database, in order to reconstruct the local view at any position and in any direction as a stealth viewer. This is particularly important in space simulations as distances vary from local to 10's or even hundreds of kilometers. but are still relevant to the exercise.

Figure 6 shows the schema for external animation of geometric models from a library, which can be assembled, have material details added, and are rendered using external position, depth and attitude information. The assembly concerns geometric models having separate parts, e.g. a helicopter with a moving rotor and tail rotor, or a space shuttle with a robot arm and moving cargo doors.

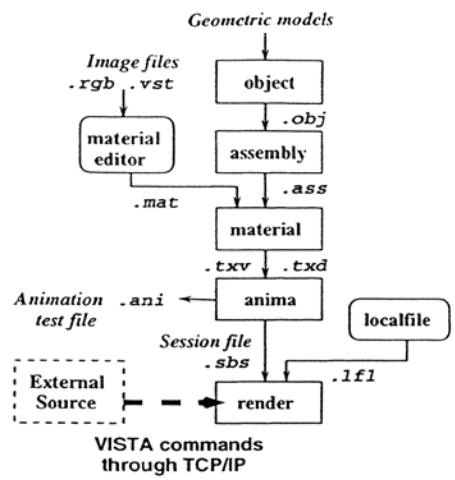


Fig. 6 The Process of Visualisation and Animation on the VISTA Software

APPLICATIONS IN DRIVING SIMULATION

Driving simulation has traditionally concerned a vehicle in which the driver sits (or simple screen which the driver sits in front of) with an animated road scene in which the drivers vehicle proceeds. Other vehicles take a relatively passive part in the simulation.

The simulation described here, provides a map which the driver uses, with optional visualisation of the scene, in which other vehicles may be driving simulators or geometric objects controlled using artificial intelligence (Al) drivers. This poses the problem of network traffic since it is necessary to use TCP/IP as the protocol for transporting the PDUs using a multicast approach.

The essential advance here is the recognition that, unlike military systems where information is only released to friendly forces and even then on a need to-know basis, when driving it is normal to signal to other motorists ones intentions when considering a manoeuvre.

Hence the proposition by Tandayya [Tandayya and Zobel, 2000bj that signals can suggest to other motorists a predicted motion when, for example turning right or left is signaled at a junction. Such motions can be pre-calculated as an "instant model" to replace the prediction mechanism of DIS and substantially reduce the consequential network traffic associated with the particular maneouvre.

This entails identifying the vehicle heading in relation to the traffic lights at a junction, the region where turning may take place and also the region in which other vehicles need to be alerted of the predicted motion as shown in Figure 7.

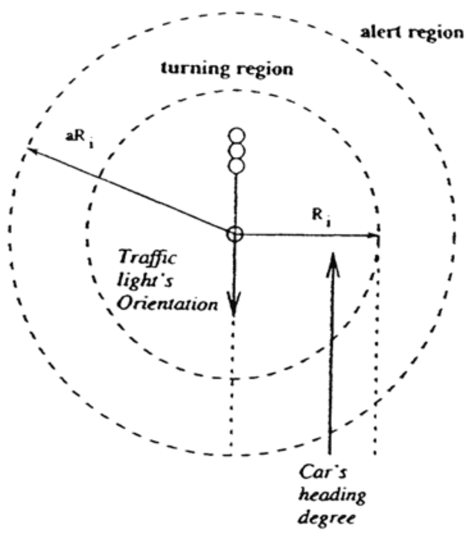


Fig. 7 Traffic Light Region

Figure 8 shows the details of the selection of an instant model for a local simulated vehicle and its transmission to provide selection data for the instant model to be used by each relevant remote vehicle. simulator. It is seen here that the local vehicle signals the exercise manager to update its binary vector (a vector of bits associated with the status of the vehicle and its environment), and its distance vector with respect to a list of junction and curve types.

This is used to communicate new signals and DIS data to other vehicles over the DIS network. The receiving vehicles then use this information to select an instant model to represent the motion of the sending vehicle as it carries out its planned manoeuvre. Where this is uncertain or where the ensuing motion is contrary to that signalled or intended (perhaps because at the last moment the driver sees a sign indicating an alternative action) the instant model invoked rapidly cause the DIS predictor to activate to override the instant model. Of course, if this happens there will be an increase of network traffic as a direct consequence of the failure of the instant model system due to driver action.

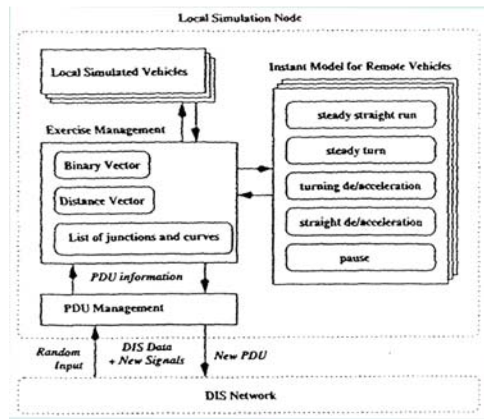


Fig 8 Instant Model fir a Vehicle

AGENT BASED SIMULATION

It was realised that this work also fell neatly into the area of Agent-Based Simulation, as defined by the chairman of the Agent-Based Simulation Workshop, held in Passau, Germany in April 2000. Hence, as this newly defined area of simulation was clearly of relevance to the authors, a paper was produced for this workshop [Tandayya and Zobel, 2000a].

The view held by the authors was that the driver and all the other actual drivers and Al controlled vehicles formed a set of interacting agents communicating over the internet, using DIS. Accordingly, the driving simulation system being developed was presented in terms of a multi-agent system.

In figure 9, we show that the drivers decisions are determined by a set of common rules along with occasional events, such as accidents and road works which give rise to changes in road signs and routes. Consequently, signals and movements of other vehicles combine with the route affected by signs concerning road works and accidents may affect the drivers decisions, giving rise to movement data and drivers signals to other vehicles, as shown the schema of figure 9.

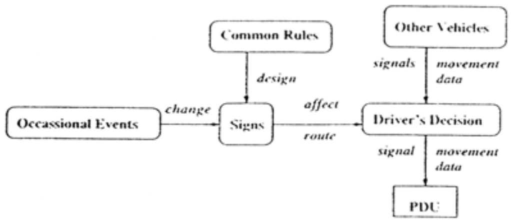


Fig. 9 interaction Diagram for an Agent-Based Driving Simulation system

SECURITY FOR DISTRIBUTED SIMULATION OVER COMMERCIAL NETWORKS

Distributed simulation is an interactive environment in which information is exchanged between agents participating within and across simulation domains. These agents are autonomous, processing their tasks independently under exercise management. In a large distributed simulation environment, these agents are organized in domains and are separately managed by some authority. The exchange of reachability information between agents across domains is achievable via each localized synthetic infrastructure over commercial network protocols such as TCP/IP.

Because distributed simulation is implemented over insecure TCP/IP protocol they inherit all of its security concerns. Most of the already existing "generalized" network security infrastructure that provides superb network security support is expensive and usually protected by those who designed them and those that need them are left with less adequate one. Therefore, it is necessary to provide additional security infrastructure that is affordable, effective and unrestricted in order to secure distributed simulation environment bearing in mind that it is important to reduced additional network overheads.

Distributed simulation implemented over TCP/IP protocol is vulnerable to all of the following security threats: Denial of Service, Information leakage, Integrity violation, and Illegitimate use. A security threat is the possibility of breaching poorly implemented security policies. Security threats are addressed through the use of security service and the means of providing this service is achievable through security mechanisms such as encryption and digital signatures. ISO 7498-2 defines five main categories of security service: Authentication, Access control, Data confidentiality, and Data integrity, Non-repudiation [Fisch and White 1999] and it therefore needs to seriously consider Data Availability which should provide solutions to denial of service attacks.

Our primary concern today is that of DDoS attacks. It is the most dangerous and very effective and it is capable of pulling down the fastest SONET technology if launched from hundreds of thousands of systems[Hancock, 2000]. The result of this is that most operating systems are likely to crash if such an overwhelming stream of traffic is burdened on it. Currently, there are no commercially available tools that effectively trace-back single DDoS source of attack, let alone multiple source DDoS of attack. One of the restraining factors is that Internet Protocol (IP) found on the network layer of the Open System Interconnect (OSI) is stateless, which should, indeed, be the case to avoid state explosion on that layer.

Currently research is being done, by the author's group, to design and implement a Path Discovery Algorithm, using already existing infrastructure requiring minimum ISP assistance, for effective trace-back of multiple DDoS source of attack for post-mortem. This is in a bid to protect not only distributed simulation environments, but also network environments as a whole.

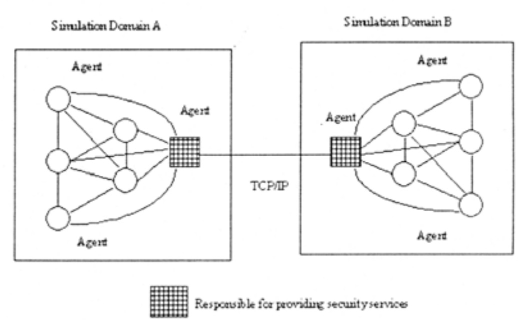


Fig. 10 Simulation over TCP/IP

NATIONAL AND INTERNATIONAL SIMULATION ORGANISATIONS

The author's first involvement in UKSC (now UKSim) is shrouded in the mists of time, earlier in the last century. However, his CV shows that he believes that he presented a paper entitled "Hybrid Computing and the Computer Science Course at Manchester" at the UKSC Conference on Computer Simulation, held at Bath University in 1969. The author, however, has little recollection of this except that he believes that lie met there two well established simulationists. namely Eugene Kerckhioffs from the Technical University of Delft and Ghislaimie Vansteenkiste from the Reichs University of Ghent, now close friends of lone standing.

I was subsequently asked to organise a UKSC conference in Manchester, which took place in Dalton Hall, a Quaker hail of residence of the University. This was eventually a success, hut was nearly a disaster. As the organiser I was not aware that the hall had just been acquired by the University and the Warden, Ted Fox, had successfully applied for an alcohol license which only came into effect on the day before the conference started!

There were regular successful UKSC conferences, but the most memorable were the 1975 conference at Bowness-in-Wnidermere in the Beautiful Lake District, and in April 1999 at Cambridge, at which a party of senior Chinese simulationists from CASS, the Chinese Association for Systems Simulation, based in Beijing attended, including Prof. Li Bo Hu and Prof. Wang Xingren. These two worthy men invited me to the 5th Chinese Symposium on Systems Simulation and Computer Science in the October of the same year, an event I remember with affection and pride.

In 1986 I attended my first SCS Summer Computer Simulation Conference, ii_1 Reno, Nevada, leading to my election as a Board Member of SCS and resulting in seven years service as a European Member of the Board.

In 1987, as a Board Member of UKSC, I began my marathon exercise of trying to persuade my colleagues from the other European Societies to form a federation, culminating in the formation of Eurosim at the last SCS Symposium, held in Edinburgh in 1989. Now a strong European voice, with both a newsletter and a Journal, I am honoured to have been associated with its foundation, and in particular with Felix Breitenecker, Dietmar Mueller and Juergen Hahn of ASIM, Kai Juslin of SIMS, Len Dekker and Eugene Kerckhoffs of DBSS, Giuseppe Iazeolla and Franco Macen of ISCS, Francis Lorenz and Henri Pierreval of Francosim, Jose-Maria Giron-Sierra of AES, Vlatko Ceric of CROSSIM, Milan Kotva and Mikolas Alexik of CSSS, Andras Javor of HSS, and Borut Zupancic of SLOSIM, and many others.

In the mean time, it became clear that UKSC could not he at the same time, a part of SCS and also a part of Eurosim. With the collaboration of the SCS Board and the UKSC Board. I persuaded all of my colleagues to separate the UK Society from SCS to form UKSim which then became a full member of Eurosim, whilst remaining in contact with SCS via mutual membership advantages.

In the mean time the author became more involved in the formation of SCS Europe and remains a Board Member of the European Council of SCS, having been involved in the organisation of several major SCS conferences in Europe, in addition to the formation of the SCS publishing house organised by my good friend Rainer Rimane, from the University of Erlangen-Nuernberg. I must also mention my good friends from SCS, notably Philippe Geril of the SCS European Office, AxeI Lehinami, Alexander Verbracht, Bernd Schmidt, Eugene Kerckhoffs, Ghi Vansteenkiste, Au Riza Kaylan, Yuri Merkuryev and many others, especially those from the USA and Canada, and of course the Pacific Rim Countries.

CURRENT AND FUTURE DIRECTIONS

There continue to be many burning issues in simulation. Distributed simulation continues to advance with increasing numbers of civil applications. The urgent need for simulation model re-use and of model databases as described in this paper needs to be addressed [Zobel, 1999]. Agent-Based simulation advances again this year with the second workshop at the University of Passau.

The merging of simulation and animation in many application areas leading to virtual reality applications in manufacturing, space, vehicle design and driving, engineering amid commerce. Web-based simulation has a great future, particularly iii relation to education in schools and in higher education, in addition to its uses in science and engineering. Perhaps, at last, science and engineering may be presented to the public in a more practical form leading to better understanding by the general public.

CONCLUSIONS

I hope that the historical aspects of this paper, with a few amusing reminiscences, will bring a wry smile to young and old, with the latter probably remembering some of their own experiences of long ago. I remember in Capri, at the first Eurosim Congress in 1989, having a lone discussion with Granimio Korn about the "good old days' of valve analog computers and of the enormous advances of the last 30+ years.

Of course simulation has made enormous strides in just the last five years and will continue. I am sure, to expand to support all human activities and perhaps lead to a reduction in war in preference for a simulated settlement of international disputes.

Simulation has brought rue a great satisfaction, many new friends and a lot of international travel to North America, to Western, Central and Eastern Europe, to the Middle East and to many countries of the Far East.

I warmly recommend simulation to you, along with membership of UKSim and SCS, may these bring you happiness, satisfaction, travel and many friends.

REFERENCES

Adib D. and Zobel R.N. 1986. 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). SCS, San Diego. pp 965-971.

Fisch E. A and White G. 1999. Secure Computers and Networks: Analysis , Design and Implementation, CRC Press LLC.

Habibi J. and Zobel R.N. 1997. Application of DIS Protocols for Integrating Simulation and Visuahisation. In *Proc. ESM '97 11th European 5mm/a/ion Multiconference,* (Istanbul, Turkey, June). SCS Europe, Ghent, Belgium. ISBN 1-56555-I 15-X. pp 107-114.

Han L. 2000. MSc Thesis. University of Manchester, U.K.

Hancock B. 2000. Networks: Denial of Service and Distributed Denial of Service. Exodus Communications lnc, Santa Clara, USA.

Kucuk B. 1998. PhD Thesis, University of Manchester, U.K.

Tandayya P. and Zobel R.N. 1998. VISTA Real-Time Animation Software in a Distributed Spacecraft System. In *Proc. SESP '98, 5th ml. Workshop on Simulation for European Space Programmes,* (Noordwijk, The Netherlands, November). ESTEC, Netherlands. ISSN 1022-6656. pp 183-190.

Tamidayya P. and Zobel R.N. 2000a. Agent-Based Simulatiomi in DIS Civil Systems. In *Proc. Agcnt-Basenl Simulation 2000,* (Passau, Germany, May). SCSI, Ghent, Belgium. ISBN 1-56555-185-0. pp 199-204.

Tandavva P. and Zobel R.N. 2000b The Instant Model for a DIS Road Traffic Simulation System. In *Proc. ESM'2000,* (Ghent, Belgium, Jumie). SCS Client Belgium. ISBN l-56555-204--0. pp 411-415.

Zobel R.N. 1970. Aspects of Hybrid Analogue and Digital Computer Techniques. PhD Thesis. Manchester University.

Zobel R.N. 1988. A User Friendly Software Environment for Design amid Simulation of DSP Algorithms and Processes on a Sun Workstation. In *Proc. ESM 88 European Simulation Environments Conf,* (Nice, France, May), SCS, San Diego. pp 65-70.

Zobel R.N. 1993. Object Classification of Simulation Models for Multidisciplinary Object Bases.' In *Proc. UKSS 93 United Kingdom Simulation Conference* (Keswick, U.K., September). UKSS, Manchester, U.K. pp 39-43.

Zobel R.N. 1999. New and Developing Areas in Modelling and Simulation in Europe. Invited paper in Proc. BICSC'99, 4th Beijing International Conference on System Simulation and Scientific Computing. (Beijing, P.R. China, October) CASS, Beijing, China. ISBN 7-80003-456-9. pp 10-14.

ABOUT THE AUTHOR

The author 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 naval surface to air missiles, using mainly valve analog computers. His Ph.D., obtained in 1970 at Manchester University, concerned hybrid analog-digital computing. As Lecturer and Senior Lecturer he became involved in digital signal processing, instrumentation amid 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 is 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 amid security for distributed simulation under commercial network protocols.