IE Interfaces
Vol. 16, Special Edition, pp. 82-86, December 2003.

Simulation-It's Expanding Role in E-Manufacturing

Ken Ebeling[1] · Sung-Youl Lee[2†]

[1]Industrial and Manufacturing Engineering, North Dakota State University, Fargo, ND 58102, U.S.A.
[2]Multimedia Engineering Division, Kwandong University, Kangwon, 215-802

E-Manufacturing 환경에서의 시뮬레이션의 역할

캔 애블링[1] · 이성열[2]

[1]노스다코타 주립대 산업공학과 / [2]관동대학교 멀티미디어공학부

This paper traces the expanding role of simulation from its early beginning on mainframe computers to the 21st Century's enterprise manufacturing environment of remote access and control. It includes an examination of the current and future role of integrated graphic animation as a primary medium of technical communications. The paper concludes with an example application of distance learning in the design, analysis, and operation of Programmable Logic Controllers on the Factory Floor of the future.

Keywords: simulation, distance learning, e-manufacturing, PLC

1. Introduction

For the benefit of those who have been introduced to computer simulation through a graphic user interface(GUI), I would like to emphasize the expanding role of simulation by starting with it's sometimes forgotten origins in the late 1950's and early 60's.

First and foremost, of course, was the development of physical scale models in fluids and hydraulics. Many of our engineering colleges still have labs for the study of compressible and non compressible flow. In industrial engineering, we have our tabletop Computer Integrated Manufacturing (CIM) labs.

With our first computers - the analog computers that were deployed in World War II- came our first continuous simulation systems. These analog computers were frequently found in Civil, Electrical, Mechanical, and even Agricultural Engineering labs. They were constructed from electrical circuits composed of power supplies, resistors, capacitors, voltmeters and ammeters. Analog circuits were designed and hand wired to solve problems in fluids, dynamics, power transmission, power distribution and a wide variety of other application areas.

Then about a decade early sixty's, the concept of a user terminal for I/O became a reality. The first terminals were limited to alphabetic characters. It wasn't until the late 60's that terminals were developed with a graphic capability.

It was during this time, in the late 60's that IBM came out with the first analog simulation package. I believe it was called ACSL. Its greatest claim to fame was that it eliminated the hard wiring of circuits. Rather, the numerical methods utilized in the programming language would allow one to change the circuit logically rather than physically. The first versions of this continuous system modeling language went into great detail explaining why the digital computer approach was better than the

† Corresponding author : Sung-Youl Lee, Multimedia Engineering Division, Kwandong University, San7 Imchunri Yangyangeup Yangyanggun Kangwondo, 215-802, Korea, Fax : +82-33-671-4144, E-mail : sylee@kd.ac.kr

analog approach. After IBM discontinued supporting their continuous system modeling language, a number of third party software houses picked up the challenge. Today we have packages like VisSim, Matlab, Spice, Labview, and a wide variety of special purpose packages.

2. The Origins of Discrete Event Simulation Systems

It was also about this same time that IBM commercialized a discrete event simulation package, called GPSS (GPSS, 1967). Actually, GPSS was originally developed as an analytical tool to help IBM application engineers and their customers select the best IBM 360 computer system. The IBM 360 was the computer industry's first complete line of computer systems; the model 30, which was the smallest, on up to what was then the largest, model 90. The initial purpose of GPSS was to model data processing applications and balance the flow of data in the computer system composed of hard disk and magnetic tape storage devices, cardpunches, line printers, memory, and CPUs. Also about this time Alan Pritsker and his graduate student, Philip Kiviat, developed the concepts which resulted in the GASP (Pritsker and Kiviat, 1969) simulation system. GPSS was written in assembly language where as GASP that employed a library of Fortran routines that were integrated with several user written event routines.

About the same time, the Rand corporation, which at the time was known as the United States Department of Defense think tank, developed computer simulation concepts that led to the development of SIMSCRIPT (Kiviat *et al.*, 1968) as a simulation system and programming language. While the machine language code generated by SIMSCRIPT executed very rapidly, the initial versions of the compiler were very slow.

Shortly after IBM unbundled the pricing of their software from their hardware as required by the US Department of Justice, IBM sold the rights to third parties and number of others entered the simulation software market. Among the new entrants to the market was C. Dennis Pegden and the SIMAN (Pegden, 1984) modeling language that he developed. Of particular significance was his contribution in explicit modeling of material handling resources and evolutionary capability to run on mini computers and later personal computers (PCs).

The introduction of simulation packages that ran on mini computers and later PCs was of particular significance for two reasons. First, it enabled the development of 'on-line' animation capabilities and second, it opened the door for the development of graphic user interfaces (GUIs). The GUIs, whose development and refinement continues with the evolution of Window based operating systems, enables the system modeler to construct logically complex models much more effectively than with conventional programming methods. This is especially significant in creating the graphic animation to be associated with a particular simulation model.

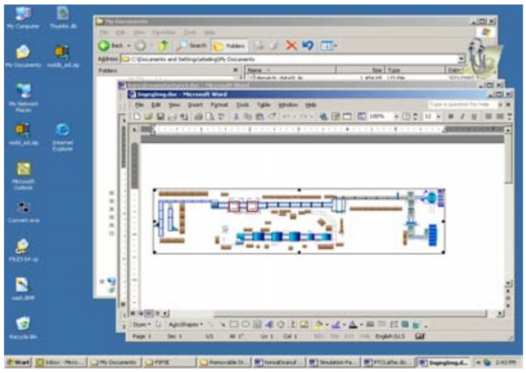

Figure 1. ProModel animation of process line.

While computer simulation has always been computationally constrained, the addition of animation can magnify the problem. Since the development of PC based general purpose simulation, graphic animation has been 2D icon based. Fortunately, the capability of our computers has been doubling every two years. The Giga hertz processors with 32 and 64 bit capabilities are currently at a stage of development that will make solid modeling practical in the not to distant future, if it isn't already. Our industrial design CAD/CAM systems are in the middle of the conversion process from wire frame modeling to parametric based solid modeling. Can next event simulation be far behind? To get a clearer understanding of this important point, let us look a little closer at the use of graphics as a medium of communication.

3. Solid Modeling as a Medium of Communications

Graphics as a medium of communication is of primary importance to the Engineering profession. We all know the frequent reference that a picture is worth a thousand words, not to mention the impreciseness of our cultural languages. After many years of manual drafting, the computer has brought about a significant evolution that continues today. To understand this evolution and its impacts on animation in our simulation models, one needs to be familiar with the following types of graphical modeling approaches:

Wire frame : Two-D animation, also called icon graphics, is the principle form of simulation animation today. It is widely used because it is relatively efficient and can dramatically improve user confidence in a simulation model. Typically, icon graphics are used to depict a model entity where the icon is moved over a background to illustrate the entity flow through the logical process being modeled. A few modeling packages will support 3D wireframe but generally the ambiguities associated with 3D icons preclude their effective usage for communication purposes.

Coloring : Coloring can be applied to 2D icon graphics to dramatically improve the appearance and effectiveness of simulation graphics. Usage in 3D is typically limited to backgrounds since rendering and shading is computationally intensive.

Solid Modeling : Product designers have used solid modeling tools to model complex assemblies for several years. Major multi-national manufacturers such as Ford, General Motors, DaimlerChrysler, and John Deere are leading the development of modern factory layout design and analysis technology. These large automotive manufacturers use a variety of manufacturing engineering tools from a variety of vendors, the integration of these tools and the exchangeability of the data they produce have become key concerns. The National Institute of Standards and Technology recently adopted Simulation Data eXchange to standardize the transfer of data between factory layout models and discrete event simulation models in the United States.

Parametric Modeling : Is an extension of solid modeling that facilitates the exploration of alternative designs with minimal input from the modeler by employing dimensional variables which can be changed on the fly. Parametric modeling is typically combined with Feature Based Modeling that utilizes parent/child relationships to remember the ordering of design or manufacturing features.

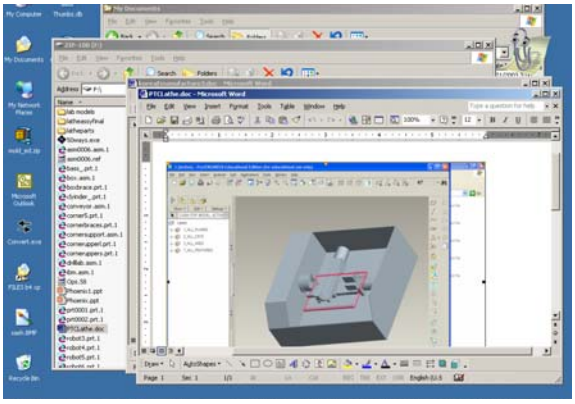

Figure 2. ProEngineering model.

Design Optimization : A further extension of feature based parametric modeling that allows the designer to formulate an objective function associated with a range of allowable design parameters. The CAD system such a Pro/ENGINEER (Pro-Engineer, 1998) can then be used to solve for the optimum design without needing to go through the iterative process of evaluating numerous combinations of design parameters.

Virtual Reality : A few automotive manufacturers are performing highly advanced factory visualization work. Their technology advancements have allowed engineers and managers to walk through a 3D factory layout in a virtual reality environment. More advanced applications of VR provide for the incorporation of intelligence within the objects of the VR animation. For most manufacturers, the computer hardware costs alone are prohibitive. As hardware costs decrease and computer chip capabilities increase, it will become common for industrial engineers all over the world to walk through 3D factory designs as part of the factory layout process.

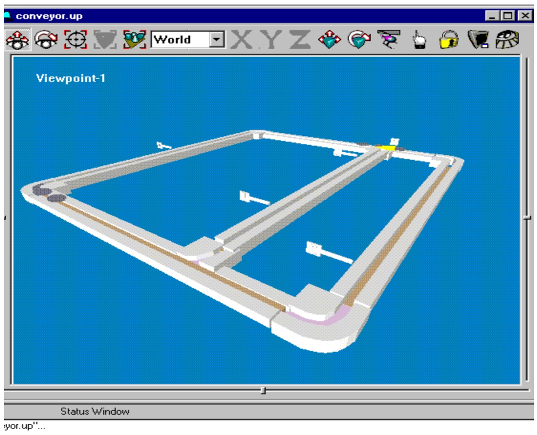

Figure 3. Virtual conveyor system.

Correspondingly, in architecture, engineering, and construction (AEC), every building is its own prototype, a one of a kind original. This approach places high value on analytical tools that enable designers to simulate and visualize the behavior of a proposed structure under varying conditions. Prescriptive building codes provide static formulas that regulate passive aspects of building performance, such as structural 'dead loads' or thermal insulation values. However, descriptive aspects of dynamic building behavior-from lighting and acoustics to seismic and wind loads to fire resistance and escape routes require correspondingly dynamic mathematical simulations that can best be understood through visualization.

Integrated Software Suite : Ideally, and integrated software suite would support a wide variety of applications from mechanical design, drafting, analysis, and animation for a wide variety of applications such as product design, manufacturing process modeling from NC programming to chip removal, facility planning and layout, production scheduling and control over the life cycle of the organization. Of course, this means realization of the goals of concurrent engineering where all stakeholders have a continual window of observation throughout the life cycle of all the organization's products.

4. Conclusions

It is obvious that we have a long way to go to realize an integrated suite of simulation software. Using the MicroSoft Office software suite as a starting place and benchmark, I would like to conclude this presentation by briefly looking at how these concepts have been realized across the wide range of industrial simulation applications. Historically, simulation, continuous and discrete, has been effectively utilized in training. Other industrial applications of discrete event simulation include manufacturing, material handling, and service industries.

Training : Unfortunately, virtual reality is often associated with computer games rather than industrial applications. Gaming is usually associated with entertainment rather than industrial training. It is true that the entertainment industry has been a leader in the utilization of computer generated animation in the creation of cartoons and the Sony PlayStation 2 has brought this technology into the home. On the other hand however, the general public is relatively unaware of the role that flight simulators have contributed to the development of visual animation. Similarly, the virtual reality work accomplished by Division (now owned by Parametric Technology Corporation) in modeling off shore drilling rigs is another good example of industrial simulation that assists in the physical design and operation of very complex and highly detailed industrial facilities. More recently the development software with a GUI interface by Sense8 makes it possible to download solid modeling objects developed within a Pro/ENGINEER CAD system. Then after imparting appropriate intelligence to the objects, it is possible to develop virtual training systems that support real time interactive training over the web. A specific example of this is our effort at NDSU to develop a virtual computer integrated manufacturing cell that can be operated interactively over the Web from many dormitory locations on campus.

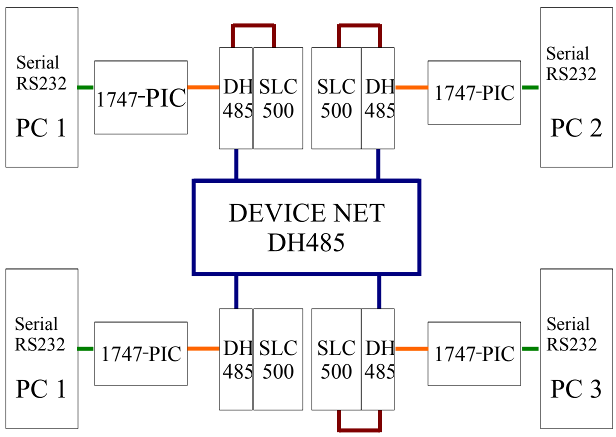

Figure 4. Network of four PLCs and four PCs.

Manufacturing : Manufacturing systems represent a class of processing systems in which entities are routed through a series of workstations, queues, and storage areas. Entities in manufacturing are parts and assemblies that have a controlled entry and routing sequence. Production is frequently performed according to schedules or triggered by low finished goods inventories or by customer orders. In highly automated facilities there is very little human involvement. Typically entity arrivals occur at set time, exhibit little variability in processing times through a fixed sequence where flow is influenced by equipment reliability, scrap rates, and material handling. Typical and classical applications include:

Methods Analysis
Plant Layout
Lot Sizing studies
Production, Planning & Control
Production Scheduling
Inventory Control
Supply Chain Management

Material Handling : Material-handling deals with the movement of parts and assemblies between and within work centers. While material handling systems are a key element in manufacturing systems, they are not uncommon in service systems. Early simulation packages had few explicit constructs. Today, many of the more than 500 different types of material handling equipment can be explicitly captured in the simulation model. This includes:

A wide variety of Conveyor Systems
Networks of Conveyors
Industrial Vehicles
Cranes and Hoists
Storage Carousels
Automated Storage and Retrieval Systems (AS/RS)
Automated Guided Vehicle Systems (AGVs)
Robots

Service : A service system is an entity processing system where one or more services are provided to customers. Entities are routed through a series of processing areas where a combination of one or more limited resources provide some type of service. Typically, the service will be intangible, perishable, and not storable. That is, production and consumption occur at the same time. Typical types of applications in a service system include:

Analysis and Design of service processing
(Call Centers and Help Desks currently receive considerable attention)
Evaluation of alternative processing systems
Systematic facility and workstation layout
Staff Planning and Scheduling

References

Bishop, Robert H. (2001), *Learning with LabVIEW*, Prentice Hall, Upper Saddle River, New Jersey.

General Purpose Simulation System/360 User's Manual, IBM Application Program, White Plains, New York, 1967.

Harrell, Ghosh, and Bowden (2000), *Simulation Using ProModel*, McGraw Higher Education, New York, New York.

Introduction to Pro/ENGINEER, Parametric Technology Corporation, Waltham, Massachusetts, 1998.

Kelton, W.D., Sadowski, R.P. (2001), and Sadowski, D.A., *Simulation with Arena*, McGraw-Hill Series in Industrial Engineering and Management Science, Boston, Massachusetts.

Kiviat, Villanueva, and Markowitz (1968), *The SIMSCRIPT II Programming Language*, Rand Corporation, Santa Monica, California.

Leeson, Daniel N., and Dimitry, Donald L. (1962), *Basic Programming Concepts and The IBM 1620 Computer*, Hollt, Rinehart, and Winston, Inc., New York, New York.

Owens, Robin (2001), Modeling Future Factories, 29-35, *IIE Solutions*, Atlanta, Georgia, August.

Pegden, C. Dennis (1984), *Introduction to SIMAN*, Systems Modeling Corporation, State College, Pennsylvania.

Pritsker, A.A., and Kiviat, P.J. (1969), *Simulation with GASP II*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

RSLinkx Lite for Allen-Bradley Programmable Controllers, User's Guide, Rockwell Software, Mayfield Village, Ohio, 1996.

Visual Basic.Net User Manual, Microsoft Press, Microsoft Corporation, Elk Grove, California, 2001.

Worldup Progrmmer's Guide, EAI Sense8products, Mill Valley, California, 2000.

Worldup User's Guide, EAI Sense8products, Mill Valley, California, 2000.