AUTHORS' BIOGRAPHIES

ROBERT S. HIGHT, Jr. is a Software Design Engineer at Northern Telecom in the Research Triangle Park, N.C. He received a B.S. in biology/computer science from Randolph-Macon College in 1980, and a M.S. degree in computer science from Florida Institute of Technology in 1983. Prior to joining Northern Telecom in 1984, he was involved with software design and development at the Kennedy Space Center, Florida (1980-1984). From 1984 to 1986 he was a member of the process control group responsible for the design and development of software to control the linecard production module. His work there included database design, statistical quality control systems, and simulation. Currently, he is involved with design and development of data communications products to enhance Centrex services.

Robert S. Hight, Jr.
Northern Telecom, Inc.
4001 E. Chapel Hill-Nelson Hwy.
Research Triangle Park, NC 27709, USA
(919) 992-8029

WILLIAM A. SMITH, JR. is a Professor of Industrial Engineering at North Carolina State University in Raleigh, NC. He is a Fellow and Past President of the Institute of Industrial Engineers and has served as Chairman, Public Affairs Council of the American Association of Engineering Societies and the National Productivity Network. He received a B.S. from the Naval Academy in 1951; M.S. from Lehigh University in 1957; and the D. Eng. Sc. from New York University in 1966. He was at Lehigh University from 1957 to 1972, where he pioneered in computer application in engineering education and served as Director of the Computing Laboratory during 1957-67. Joining North Carolina State University in 1973 as Head of the Industrial Engineering Department until 1982, he concurrently served as Director of the Productivity Research and Extension Program. During 1984-6, Dr. Smith served as full time Consultant on Automation Engineering working on design, development, and implementation of the line card production module at Northern Telecom, Research Triangle Park while on academic leave. He is currently involved with development of programs in quality and productivity improvement, management systems engineering, organization effectiveness, and operations planning.

William A. Smith, Jr.
Industrial Engineering Department
Box 7906
North Carolina State University
Raleigh, NC 27615
(919) 737-2362

Proceedings of the 1987 Winter Simulation Conference
A. Thesen, H. Grant, W. David Kelton (eds.)

AUTOMATIC PROGRAMMING OF AGVS SIMULATION MODELS

by

Mark K. Brazier
and
Robert E. Shannon

Industrial Engineering Department
Texas A&M University

ABSTRACT:

This paper presents a knowledge based modeling system that allows a manufacturing engineer who has very limited knowledge of simulation methodology to quickly and correctly, develop and run a simulation model of an automated guided vehicle system (AGVS). The modeling system is capable of guiding and assisting the engineer with a level of "expertise" comparable to a trained simulation specialist. The modeling support program is an automatic programming system, written in Turbo-Prolog which generates the computer code for the required model and experiment in the SIMAN simulation language.

INTRODUCTION:

Accounting for 30 to 80 percent of the total cost of production, efficient material handling is critical to a manufacturing facilities success [Apple 1972, Hill 1986]. With the recent advances in microprocessor technology and control systems, automated guided vehicle systems (AGVS) have become a key component for factory modernization and increased productivity.[Bose 1986].

The concept of an automated guided vehicle system (AGVS) embraces all transport systems which are capable of functioning without driver assistance [Muller 1983]. When first introduced to industry, the AGV was primarily in competition with driver operated material handling systems such as fork lifts, pallet carts, tow tractors etc. However, as micro-processor technology has evolved, so too have the capabilities and types of AGV systems. Today's AGVS is now in direct competition with other forms of industrial transportation (e.g. roller conveyors, belt conveyors, chain conveyors etc.).

AGVS applications within the United States have grown considerably during the last decade and are currently expanding at a 30% annual rate. Systems can be found in nearly every segment of industry. Where technically feasible, AGVS implementations are limited only by their operational considerations and economic viability [Lasecki 1986].

As a material handling system, AGVs offer several potential advantages over more traditional forms of material handling including [Muller 1983]:

1. Improved resource utilization.
2. Better control of material movement.
3. Increased throughput.
4. Reduced product damage.
5. Adaptable to routing changes.
6. Easily interfaced to existing material pickup/delivery points.
7. Highly accurate load positioning capabilities.

The need for extensive studies durimg the design and planning of AGV systems has been widely accepted as necessary if managers and engineers wish to understand the impact of alternative proposals without the risk of costly implementation failures. Because of their high cost, space and maintenance requirements, there is no question that AGV systems must be thoroughly evaluated prior to their implementation if their benefits are to be fully realized.

Due to their complex, dynamic behavior and high degree of interaction with all the other components of the manufacturing system, it is difficult to characterize their performance analytically [Kuhn 1983]. Alternatives to simulation modeling of material handling systems do exist. Operations research methodologies, particularly queueing network models have been used to study a number of performance measures such as machine/station utilization, throughput rates, machine queue data, work in progress (WIP), etc. Their primary advantage is speed of use, deterministic output (ease of interpretation) and low cost due primarily to their limited data input and computer processing requirements.

However, certain unrealistic assumptions such as deterministic service times and infinite station buffers must be made in order to use them. A significant disadvantage with respect to studying AGVS is that vehicle blocking interactions cannot be studied, nor can the effects of machine failures, dispatching and routing decisions, vehicle maintenance cycles etc. With the availability of third and fourth generation simulation software and lower cost computers, the relative ease, speed and cost advantages of queueing network models has lost much of its attraction.

It is important to note that a simulation model does not deal directly with aggregate systems behaviors. Instead, it focuses on the individual behaviors of entities within the system, documenting each of their changes over time. Because each object's behavior is considered in as much detail as the underlying program's logic will permit, a model can be developed to duplicate a system's behavior to any desired level of realism or completeness.

The need to conduct simulation studies in the design and planning of AGV systems has been widely accepted as necessary, especially since managers have found them to be very cheap form of insurance against costly mistakes [Church 1982, Kay 1984, Mills 1983, Duffau & Bardin 1985].

GOAL:

Development of a valid simulation model is critical to the design teams effective evaluation of a proposed system. Unfortunately, the skills and knowledge necessary to develop valid models is in short supply [Shannon 1985]. Although properly trained simulation specialists and consultants are obtainable, they do not usually have the understanding of the manufacturing systems needs and objectives to conduct a proper study without extensive interaction with the design engineer. A highly desirable alternative would be a simulation system that allows the manufacturing engineer or designer, familiar with his facilities needs, to do the simulation modeling himself without the requirement for elaborate training or the assistance of outside experts. This system would be capable of providing expert guidance and assistance to the engineer in the development of a valid simulation model in a relatively short time span.

The objective of this paper is to present a microprocessor based simulation system developed at Texas A&M University to address some of these needs. This system called SIMTOOL, is an artificial intelligence based, interactive simulation tool programmed in Prolog to assist the user (through automatic programming) in developing programs, written in SIMAN, for the simulation of AGV systems.

Automatic programming (AP) is an applications area of artificial intelligence (AI) concerned with automating some aspects of the computer programming process [Barr & Feigenbaum 1982]. The goal of an AP system is to improve the environment in which a program is specified. This improvement is achieved through two key elements; (1) a reduction in the amount of detail that the programmer is confronted with and (2) a specification language more natural to the user's problem domain and way of thinking. By improving the programming environment, the programmer can more easily assemble or modify complex programs and can do so with increased confidence.

Bonn et al [in Barr & Feigenbaum 1982] have characterized AP systems as having four general identifying characteristics:

1. Specification Method - The reported system uses a combination of graphics, pulldown menus and form (template) completion.

2. Target Language - SIMTOOL writes the "goal" program in SIMAN [Pegden 1986].

3. Problem Area - The domain or area of intended operation is the design and analysis of AGV systems in a manufacturing facility.

4. Method of Operation - SIMTOOL uses expert, rule-based knowledge about AGV system modeling using Prolog first order predicate calculus (horn-clause format).

CHARACTERISTICS:

SIMTOOL is an interactive simulation tool written in Borland's Turbo Prolog for the IBM PC and compatibles. It consists of approximately 6,000 lines of code containing 752 rules. It requires a minimum system configuration of 640K of RAM and color display/graphics capability. It is designed to assist the user who has very limited (or no) training in simulation methodology to quickly and correctly develop and run discrete event AGVS simulation programs.

SIMTOOL employs extensive user-friendly data entry procedures including:

1. Context sensitive help functions,

2. Line input drivers with default text capability,

3. A graphics driven, rule based layout assistant,

4. Manual override and text editing of SIMTOOL generated data elements and code,

5. Wordstar style text editing,

6. Pulldown, popup and box menus,

7. Window status lines,

8. Virtual layout screen capability,

9. Full color windows with resizing functions,

10. Programmed function keys,

11. Extensive error trapping of user inputs and for model information completeness.

SIMTOOL is structured into four distinct logical processes to further enhance the user's perspective of the modeling environment while reducing the scope of complexity. These processes are:

1. Plant Layout - This process allows the user to graphically define his or her proposed AGVS network.

2. Data Tables - SIMTOOL converts the proposed network into proper data tables reflecting the node relationships and distances necessary for simulation modeling. Data table calculations are transparent to the user.

3. Generate Model - This process allows SIMTOOL to interact with the user to solicit parameters necessary to define the proposed models functional relationships and dynamic characteristics. Upon completion, an ASCII SIMAN program representing the proposed model and experiment is generated automatically.

4. Compile Model - This process invokes the SIMAN model and experiment processors for compilation of the ASCII model and experiment files generated earlier during

the Generate Model process. Upon completion, the SIMAN link processor links the two compiled files as a single executable file. SIMTOOL then submits the linked files to the run time processor for execution.

SIMTOOL's separation of the model definition phase into four distinct functional activities allows the user access to the model at several stages in it's development life cycle. In so doing, SIMTOOL provides the user with the ability to develop models over an extended period of time or to modify existing models with a minimum of effort.

The breakdown of SIMTOOL into distinct activities also serves a practical purpose. By separating each of the activities into modules that can be executed individually (via virtual memory using the DOS Linker utility), computer memory requirements for SIMTOOL are substantially reduced. This reduction in memory requirements is essential to SIMTOOL's ability to operate in the microprocessor environment.

SIMTOOL NETWORK:

The modeling process begins with a graphical description of the proposed network layout. A SIMTOOL network is a directed graph of nodes and arcs which represent the AGV system. Arcs represent the travel path or routes that the vehicle follows during navigation of the facility. All arcs are directed, that is to say, each represents an unidirectional path of traffic flow (a path being any sequence of nodes connected by arcs). A paths direction is based on a user specified parent-child precedence relationship between connected nodes. Nodes represent points where loading/unloading, vehicle queuing, route assignment, vehicle selection, and vehicle blocking can occur. Two types of nodes are employed in a SIMTOOL network: (1) work station nodes and (2) track intersection nodes.

SIMTOOL has a virtual screen and is dimensioned from the upper left hand corner. The current location of the cursor is always indicated on the top row of the screen. The user decides the scale of the graphic drawing i.e. each row and each column represents 1 foot or 10 feet. The user draws the proposed physical layout on the screen by placing the work stations in their proper location and drawing the proposed interconnecting track system (including track intersections).

Work station nodes represent AGV delivery and pickup points (i.e. machine cells, storage/retrieval systems, shipping/receiving docks etc.) typically found in the manufacturing environment. Each station is identified by the prefix "S" followed by the users assigned station serial number for identification purposes. Since each station node represents a pickup/delivery point for vehicles, each is equipped with features designed to support vehicle movements. These features include queues, robots/workers (for loading/unloading), and vehicle traffic control.

Five queues are provided for each station node. They are:

1. Track wait - Empty Vehicle
2. Track wait - Loaded Vehicle
3. Station Robot Wait - Vehicle Offload
4. Station Robot Wait - Vehicle Load
5. Machine Center WIP

Track Wait - Empty Vehicle, is designed to store empty vehicles in transit to another station for load pickup but delayed for traffic control at the current station. The second queue, Track Wait - Loaded Vehicle, is similarly designed except that vehicle statistics are maintained for loaded rather than empty vehicles. The third queue, Station Robot Wait - Vehicle Offload, is designed to accommodate vehicles delivering a load to the current station but delayed waiting for allocation of the robot/worker resource needed to offload. The fourth queue, Station Robot Wait - Vehicle Load, is essentially identical to queue type 3 except that it represents vehicles waiting for robot loading rather than offloading. The fifth queue, Machine Center WIP, represents the station work-in-progress (WIP) queue.

Each queue is identified by its user assigned station number during post-simulation analysis. This association is done automatically by SIMTOOL during model generation and remains transparent to the user.

The second type of network node is the intersection node. This node is designed to represent path or arc junction points where alternative vehicle routing can occur. Up to three incoming paths or three outgoing paths may access this node. However, isolated nodes (i.e. no input or no output) in the network are not permitted. The intersection nodes are identified by the prefix "I" followed by the user's assigned intersection serial number.

From the graphic layout, SIMTOOL will calculate the distances between nodes. The intersection nodes also provide a very useful mechanism for calculating optimal path inter-node distances. Since vehicle routing decisions for optimal path selection are made only at intersection nodes and not at station nodes, the number of nodes contained within the search space of a shortest path (exhaustive search) algorithm can be substantially reduced. This is the approach used by SIMTOOL during generation of the vehicle distances used in the SIMAN experimental file distance lookup tables.

SIMTOOL's distances are calculated using a method somewhat similar to Schildt's [1987] shortest path algorithm which uses Turbo Prolog's default search method of depth-first. However, the SIMTOOL algorithm is modified to make use of the intersection node search space reduction strategy. This is done by using the list handling mechanisms available in Prolog to group station nodes into "family lists" containing only those S-nodes found between any two I-nodes. This modification significantly improves the algorithms performance, especially for networks with large numbers of multiple path alternatives and station nodes.

MODEL PARAMETERS:

While the model layout process graphically depicts the network's physical relationships such as vehicle path direction, vehicle delivery/pickup points, track intersection or route decision points, etc., it does not fully characterize the intended model's static and dynamic descriptions. There is other information regarding characteristics of the stations, AGV's, the parts to be processed etc. required by the model. This information is entered through a system of selection from pulldown, popup, box menus and the answering of questions posed by the system. These parameters either relate to the jobs to be processed or the AGV system itself.

In the case of jobs to be processed, selection of the "Jobtypes" item from the "GEnerate Model" menu starts a series of interrogations in which the user is asked to specify:

1. Number of job types and their frequency of occurrence,
2. The routing sequence for each job type,
3. The processing time for each station and job type,
4. The arrival/departure station name.

Likewise the selection of "Vehicle Parameters" leads to an interrogation of the user to define:

1. Number of vehicles,
2. Empty and loaded vehicle speeds,
3. Initial vehicle positions within the network,
4. Number of robot/workers for vehicle loading/unloading at each station,
5. The processing time for loading/unloading of vehicles,
6. AGV track Zone length (used to control how close two vehicles can follow each other).

The experienced user of SIMAN will recognize immediately how this information is used in the experiment file to be created. The information entered by the user is stored in system generated files. By so doing, parameters can later be modified, either as a group or individually at the user's convenience. This capability provides a significant time savings when multiple iterations using the same basic model are necessary.

MODEL GENERATION:

Once the proposed AGV system has been completely described, generation of the SIMAN model can begin. This process is straight forward and only requires selection of the "Generate Model Code" option found in the "Generate Model" pulldown window menu. SIMTOOL then automatically generates both the ASCII SIMAN model file and the experiment file for the proposed system. This process remains entirely transparent to the user. The experienced simulation modeler can then examine, print out or modify these files as desired by going into the SIMTOOL editor.

The translation will be apparent to the user of SIMAN with the possible exception of a couple of modeling tricks. For example, vehicle movement on the track is controlled using the same scheme as most real world systems. Intersections are declared as resources to be seized before proceeding and released after clearing. This precludes two vehicles going through the intersection from different directions at the same time. Likewise, SIMTOOL has used the user specified "Track Zone Length" to create for each segment of track between nodes, a set of indexed resources which are seized and released as the vehicle proceeds down the track. This mechanism keeps vehicles from following too close and also insures that two vehicles are not occupying the same space.

After the system informs the user that the model has been generated, the user selects the "Compile Model" selection from the main menu and the computer executable code is compiled and linked for running.

SUMMARY:

The system discussed in this paper is an AI-based system that allows the manufacturing engineer with limited training in simulation methodology to quickly and correctly develop and run simulation models of an AGV system under study. Fairly complex AGV systems have been modeled in less than 30 minutes by users with no previous experience in simulation.

It is a demonstration of one approach to introducing AI technology into the field of simulation, namely what we call hybrid systems. Hybrid systems are those in which intelligent front and backend user interfaces are built around existing simulation systems. Other examples of this approach can be found in the literature. For example, see Murray 1986, Ford & Schroer 1987, Mellichamp & Wahab 1987 all of which use SIMAN as the underlying language or Nyan 1987, Seliger et al 1987 which use SIMULA.

The work reported on in this paper is only one part of a larger on-going research program at Texas A&M University to try to integrate AI and simulation technologies [Adelsberger et al 1986, Shannon et al 1986]. It is clear that the transition to significantly better and easier to use simulation systems is underway [Shannon 1986]. The increasing use of interactive graphical model construction and data input, graphical and animated output analysis, the embedding of more and more of the statistical analysis within the language, all are first tentative steps.

Simulation systems based upon the application of artificial intelligence and logic programming will hopefully generate a new more powerful environment for simulation modeling. The goal is to simplify and put at the fingertips of the semi-naive user, the expertise of the most knowledgeable and experienced simulation experts.

REFERENCES:

Adelsberger, H.H., U.W. Pooch, R.E. Shannon and G.N. Williams, "Rule Based, Object-Oriented Simulation Systems," Intelligent Simulation Systems, SCS Simulation Series, Vol. 17, No. 1, pp:107-112, January 1986.

Apple, J.M., Material Handling System Design, Ronald Press, NY, 1972.

Barr, A. and E.A. Feigenbaum, The Handbook of Artificial Intelligence, William Kaufman, Inc., CA, Vol. 2, 1982.

Bose, P.P., "Basics of AGV Systems," American Machinist and Automated Manufacturing, Vol. 130, No. 3, pp:106-122, March 1986.

Church, J., "Simulation Aspects of Flexible Manufacturing Systems - Design and Analysis," 1982 Fall Industrial Engineering Conference Proceedings, IIE, Atlanta, GA, pp:426-431, 1982.

Duffau, B. and C. Bardin, "Evaluating AGVS Circuits by Simulation," Proceedings of the 3rd International Conference on Automated Guided Vehicle Systems, Elsevier Science Publishing Co., Inc., NY, pp:229-246, October 1985.

Ford, D.R. and B.J. Schroer, "An Expert Manufacturing Simulation System," Simulation, Vol. 48, No. 5, pp:193-200, May 1987.

Hill, J.M., "The Changing Profile of Material Handling Systems and Controls," Industrial Engineering, Vol. 18, No.11, pp:68-73, November 1986.

Kay, J.M., "The Use of Modeling and Simulation Techniques in the Design of Manufacturing Systems," International Conference on the Development of Flexible Automation Systems, Institute of Electrical Engineers, NY, pp:55-61, 1984.

Kuhn, A., "Efficient Planning of AGVS by Analytical Methods," Proceedings of the 2nd International Conference on Automated Guided Vehicle Systems, IFS (Publications) Ltd., United Kingdom, pp:1-10, June 1983.

Lasecki, R.R., "AGV System Selection Methodology," ULTRATECH Conference Proceedings, Society of Manufacturing Engineers, Vol. 1, Section 3, pp:81-96, September 1986.

Mellichamp, J.M. and A.F.A. Wahab, "An Expert System for FMS Design," Simulation, Vol. 48, No. 5, pp:201-208, May 1987.

Ford, D.R. and B.J. Schroer, "An Expert Manufacturing Simulation System," Simulation, Vol. 48, No. 5, pp:193-200, May 1987.

Hill, J.M., "The Changing Profile of Material Handling Systems and Controls," Industrial Engineering, Vol. 18, No.11, pp:68-73, November 1986.

Kay, J.M., "The Use of Modeling and Simulation Techniques in the Design of Manufacturing Systems," International Conference on the Development of Flexible Automation Systems, Institute of Electrical Engineers, NY, pp:55-61, 1984.

Kuhn, A., "Efficient Planning of AGVS by Analytical Methods," Proceedings of the 2nd International Conference on Automated Guided Vehicle Systems, IFS (Publications) Ltd., United Kingdom, pp:1-10, June 1983.

Lasecki, R.R., "AGV System Selection Methodology," ULTRATECH Conference Proceedings, Society of Manufacturing Engineers, Vol. 1, Section 3, pp:81-96, September 1986.

Mellichamp, J.M. and A.F.A. Wahab, "An Expert System for FMS Design," Simulation, Vol. 48, No. 5, pp:201-208, May 1987.

Mills, R.I., "Computer Simulation - A Feasibility and Planning Tool for FMS," Proceedings of 2nd International Conference on Flexible Manufacturing Systems," IFS (Publications) Ltd., United Kingdom, North-Holland Publishing, NY, pp:185-196, 1983.

Muller, T., Automated Guided Vehicles, IFS (Publications) Ltd.,UK, Springer-Verlog, NY, 1983.

Murray, K.J., "Knowledge-based Model Construction: an Automatic Programming Approach to Simulation Modeling," a Ph.D. dissertation, Computer Science Dept., Texas A&M University, College Station, TX, 1986.

Nyan, P.A., "A Comprehensive Environment to Object Oriented Simulation of Manufacturing Systems," Simulation in Computer Integrated Manufacturing, (Editor - K.E. Wichmann) European Simulation Multiconference, SCS Publications, Ghent Belgium, pp:21-25, 1987.

Pegden, C.D., Introduction to SIMAN with Version 3.0 Enhancements, Systems Modeling Corporation, State College, PA, 1986.

Schildt, H., Advanced Turbo Prolog, Osborn McGraw-Hill, Berkely CA, 1987.

Seliger, G.B., B. Viehweger, B. Wieneke-Toutataoui and S.R. Kommana, "Knowledge-Based Simulation of Flexible Manufacturing Systems," Simulation in Computer Integrated Manufacturing, (Editor - K.E. Wichmann), European Simulation Multiconference, SCS Publications, Ghent Belgium, pp:65-68, 1987.

Shannon, R.E., R. Mayer and H.H. Adelsberger, "Expert Systems and Simulation," Simulation, Vol. 44, No. 6, pp:275-284, January 1985.

Shannon, R.E., "AI Based Simulation Environments," Intelligent Simulation Environments, SCS Simulation Series, Vol. 17, No. 1, pp:150-156, January 1986.

Shannon, R.E., R. Mayer and D.T. Phillips, "Knowledge Based Simulation Techniques for Manufacturing," ULTRATECH Conference Proceedings, Vol.1, Section 1, SME Paper MS 86-966, pp:237-262, September 1986.

AUTHORS' BIOGRAPHIES:

MARK K. BRAZIER is a Senior Information Systems Analyst with the Industrial Modernization Division of LTV Aerospace and Defense Company. Since joining LTV he has been actively involved in the development of LTV's Flexible Composite Center. He received a B.S. degree in Computer Science in 1978 and is a candidate for the MS degree in Industrial Engineering at Texas A&M University. Current interests include simulation language development, AI applications in manufacturing, process control and material handling systems. He is a member of Alpha Pi Mu, IIE, SME and SCS.

DR. ROBERT E. SHANNON is Professor of Industrial Engineering at Texas A & M University. His current research efforts are in the design of Fifth Generation Simulation Systems based upon AI/ES technology. He is the author of two books, *SYSTEMS SIMULATION: THE ART AND SCIENCE*, Prentice-Hall, (which won the H. B. Maynard Technical Book of the Year award and has been translated into Russian, Japanese and Farsi) and *ENGINEERING MANAGEMENT*, John Wiley & Sons, as well as over 50 journal and technical papers. Dr. Shannon is active in a number of Professional Societies including IIE (in which he is a Fellow), ORSA, TIMS, SCS and honorary societies such as Sigma Xi, Alpha Pi Mu and others.