Proceedings of the 1980 Winter Simulation Conference
T.I. Ören, C.M. Shub, P.F. Roth (eds.)

A MODULAR APPROACH TO SIMULATION OF ROBOTIC SYSTEMS

* D. J. Medeiros
* R. P. Sadowski
** D. W. Starks
*** B. S. Smith

*School of Industrial Engineering
Purdue University
West Lafayette, Indiana 47907

**Pritsker and Associates
Fort Worth, Texas 76112

***McDonnell Aircraft Company
St. Louis, Missouri 63166

This paper presents the results of a simulation effort performed as part of the ICAM robotics program. The initial Q-GERT simulation model is briefly described, followed by the presentation of a modular approach to simulation of robotic systems. In this approach, predefined Q-GERT modules, described by simple block diagrams, are assembled to produce a complete model. The system design logic is discussed and the available modules are briefly described. Examples are presented in block diagram format. The advantages and limitations of the approach are discussed.

1. INTRODUCTION

The Air Force Integrated Computer Aided Manufacturing (ICAM) Program was initiated to foster the development of high technology manufacturing systems for low-volume batch manufacturing. Because of the low volume, specialized transfer machinery is uneconomical. Robots, which are programmable, can be used for such applications. The robot is "taught" to perform a task by leading the arm through a series of motions, either manually or with a set of controls. If a large number of different parts are to be processed, which is typical of low-volume batch manufacturing, much productive time is consumed in programming. Therefore, one thrust of the ICAM program is the development of a robot programming language similar to the APT language which is used for programming NC machines. Programs would be created off line and transferred to the robot controller as required.

Another difficulty in using a robot is that part orientation must be known. This requires the use of a vision system which is capable of recognizing many different part types and calculating their orientation. Such vision systems are presently in the developmental stage.

As part of the ICAM program, McDonnell Douglas is integrating a robot, a vision system, and a machine tool into a demonstration work cell which will be used to rivet aircraft parts. Purdue University was responsible for modeling the work cell. Simulation was required because no such system as the demonstration cell existed at that time. Before such a cell was constructed, it was useful to be able to determine its output rate, reliability and other operational parameters.

The objective of the modeling effort was to provide a means of analyzing the technological capabilities and cost effectiveness of various types of manufacturing cells. Further, personnel with little experience in simulation should be able to use the developed models, so that many alternatives could be easily evaluated.

The first task was to create a model of the envisioned demonstration cell (Fig. 1). Once this model had

80CH1617-0/80/0000-0207$00.75

been completed, the second task was to design a simulation capability for advanced work cells with more complicated operations, and possibly more than one robot.

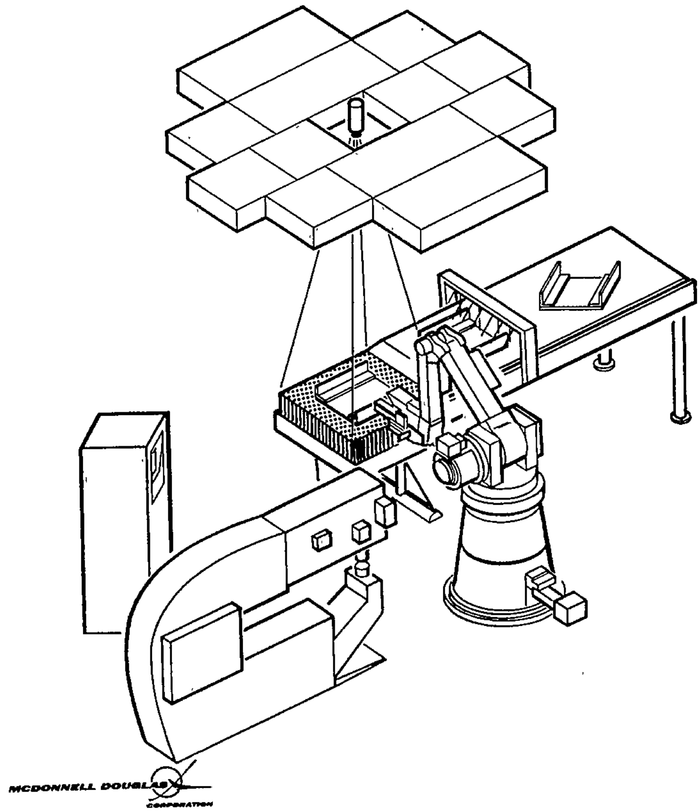

Figure 1. Demonstration Cell

2. THE INITIAL MODEL

The Q-GERT simulation language was selected for use in the modeling effort for two reasons. First, this language is very easy to use and models can be created and implemented on a computer quickly. More importantly, the network itself provides a vehicle for communication with individuals whose experience in simulation is limited. After the basic node types are explained, it is easy to follow the flow of entities through the network and thus determine if the logic used in building the model is correct.

The first model was intended to be used for the demonstration cell only. The model was rather large, containing approximately 95 nodes and 200 lines of FORTRAN code. Most of the modeling effort was directed at accurately representing the hierarchy of the system above the work cell level.

This hierarchy consists of three levels of computers. A main frame stores the part program, which is a description of the part and the operations performed upon it. Center control, a minicomputer, is responsible for selecting the parts to be processed and routing these parts through the system. It requests the part program from the mainframe and transmits the information to one of a group of work station control computers. The work station control, either a mini or micro computer, is responsible for directing the activity within the work cell. Center control stores information on the characteristics of each work cell so it can route its part programs to an appropriate cell. After the work cell is initialized, the part may be produced.

A workpiece is loaded onto a table equipped with a camera, where its orientation is calculated. The robot picks up the workpiece and brings it to an automatic riveting machine. The workpiece, which was previously assembled with tape, is moved by the robot between riveting machine cycles, thereby inserting a row of rivets into the workpiece. After completion of the riveting operation, the robot deposits the part on a conveyor for removal from the cell and moves to the initial staging area to pick up the next part.

At first, it was anticipated that the hierarchy would greatly influence the capacity of the system. Thus, a great deal of effort was devoted to accurately representing the data transfer between machines and the line failures or system failures which could occur, while the activities within the manufacturing cell were modeled with considerably less detail. The model could represent up to five work station controls with associated robots, vision devices, and automated machine tools. Each cell could have a different operating efficiency, and each part program could be restricted to a subset of the manufacturing cells.

The simulation model was employed to analyze various work cell designs and to select, from a number of components of differing reliability and work rate, the most cost effective units. In addition, a sensitivity analysis was performed on several parameters which defined the operation of the system. Results of this analysis revealed that the hierarchy was not a dominant factor in the system. Even when failure rates were increased to the 80-90% range, the effect on the system was minimal. The average time to complete a single work piece was almost linearly related to the robot task time.

These results indicated that more detail was required at the work cell level, and much less above this level. Therefore, the emphasis of the second model was directed at the work cell.

3. THE MODULAR APPROACH

More was required of the second model than merely a change of emphasis within the simulation. It was desired to include the capability of modeling other types of manufacturing cells than the demonstration cell, in a model intended for use by persons without much experience in simulation.

It is possible to create a general model which contains the elements of several potential cell configurations. However, for any one application much of this model would be overhead. In addition, the intricate branching structure required makes explanation and understanding difficult. Because of these difficulties, a building block approach to modeling was employed. This approach requires the user to construct his models from a group of pre-defined modules. The disadvantage of this approach is that it places the burden for linking the modules correctly upon the user. However, the flow of workpieces through the system is transparent, which facilitates the communication process among users.

The scope of the system was restricted to applications in which the robot moves the workpiece through a manufacturing cell. This cell may consists of one or more robots, automated devices such as machine tools, and staging areas which hold the work pieces and at which the orientation operations are performed. Three major configurations were considered: the robot working in conjunction with an automated device, the robot loading and unloading automated devices, and the robot loading automated devices which automatically eject completed workpieces. In addition to this, provision had to be made for generating the workpieces, allowing the robot to pick up a workpiece, and terminating the simulation when the correct number of workpieces had been processed.

In designing such a system, several factors must be considered. It is desirable to keep the number of modules to a minimum, so that selection of the proper modules is simplified. It is also necessary that the number of modules required for any one model be kept small. These two goals are somewhat contradictory. As the number of modules per model is decreased, each module by necessity becomes more specialized, since it encompasses a larger portion of the system. This in turn requires the creation of a larger number of modules to allow the modeling of many types of systems. The limit of this process is reached when there is a separate, self contained model for each system, which is the normal practice in simulation. At this level, some amount of skill in modeling is required.

Creating the modules involved grouping of the system functions into sets of related tasks. These sets were then structured to provide system flexibility by limiting the number of initial and final module states. This increased the number of ways in which the modules could be interconnected, since matching of states is required for linkage. At the same time, an attempt was made to reduce the amount of duplication between modules. Thus each of the 5 basic modules performs a different operation, as shown in Table 1.

The first module starts the simulation. It makes provision for the hierarchy above system control and also provides the option to ignore this hierarchy. It generates the correct number of workpieces and feeds them, one at a time, onto the initial staging area where they await orientation. The next module represents the orientation operations and the pickup of the workpiece by the robot. A module is included to terminate the simulation and collect statistics on workpiece throughput time and rate.

Module	Function
0-a	transfer part program through hierarchy, generate workpieces
0-b	generate workpieces
1	workpiece orientation & pickup
2-a	robot and machine interaction
2-b	robot and machine interaction, workpiece putdown
3-a	machine loading & unloading
3-b	machine loading & unloading, workpiece putdown
3-c	machine loading & unloading with storage for incoming workpieces
3-d	machine loading & unloading, workpiece putdown with storage for incoming workpieces
4-a	machine loading with automatic eject
4-b	machine loading with automatic eject and storage for incoming workpieces
5-a	removal of completed workpiece & system shutdown (with 0-a)
5-b	removal of completed workpiece & system shutdown (with 0-b)

Table 1. Module Functions

A group of modules represent the robot interacting with one or several idential automated devices. Each module has two termination options: either the robot is left holding the semi-finished workpiece, or the robot may put this workpiece down at a staging area.

In addition, in the modules representing machine loading, an optional intermediate staging area is included. If all automated devices are busy when the robot is holding an incoming workpiece, it will deposit that workpiece at an intermediate staging area, and will load it onto an automated device when one becomes idle. This buffer storage allows for differing work rates among a group of serially organized automated devices.

Four of these modules have 2 or more variations. For example, module 2, which represents a robot and automated machine working in conjunction (as in the demonstration cell) has two subcases. The first of these ends with the completion of the machine operation; it is used when additional machining immediately follows. The second includes putdown of the semifinished workpiece at a staging area, and so allows delays and queueing of workpieces. Since the operations are essentially similar, it would have been possible to isolate the workpiece putdown as a separate module and dispense with module 2-b. However, this would increase the number of modules which must be interconnected to create a system model, and thus render the modeling effort more difficult.

Once the work cell tasks had been divided into modules, it was necessary to find a means of describing the operation of the modules at a higher level than the Q-GERT networks. Accordingly, block diagrams were created which illustrate the essential features of each module, without details such as failures and queueing behavior. A block diagram and the associated Q-GERT network for module 2-a is contained in Fig. 2. The hexagonal block represents the automated device resource. An open block means resource allocation, and a slashed block is deallocation.

The manufacturing cell possesses a limited set of resources; thus, the status of the system can be easily ascertained by examining the status of its resources. In addition, the status of the robot and staging area resources defines the entry and exit conditions for the modules. Linking of modules involves matching the entry condition of one module with the exit condition of the previous module. For example, if a module ends with the robot depositing a semi-finished workpiece at a staging area (staging area resource in use), the following module must represent part pick-up from a staging area. If a module ends with the robot holding a workpiece (robot resource in use), the next module must represent the robot interacting with one or more automated devices. This approach allows a great deal of flexibility in modeling; many serial operations may be represented within a single manufacturing cell model. Table 2 shows the permissable module linkages.

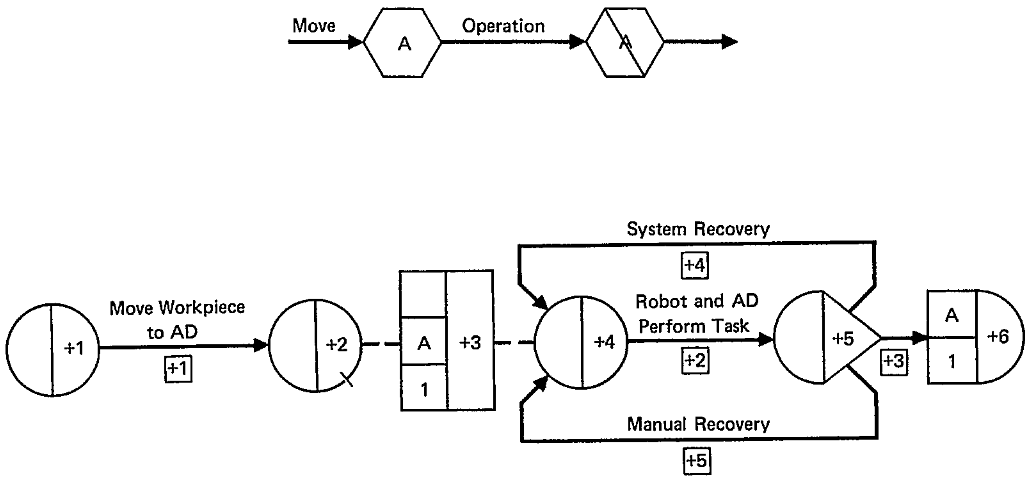

Figure 2. Module 2-a Block Diagram and Q-GERT Network

		To												
		0-a	0-b	1	2-a	2-b	3-a	3-b	3-c	3-d	4-a	4-b	5-a	5-b
From	0-a			*										
	0-b			*										
	1				*	*	D	D	*	*	*	*		
	2-a				*	*	*	*	*	*	*	*		
	2-b			*									*	*
	3-a				*	*	D	D	*	*	*	*		
	3-b			*									*	*
	3-c				*	*	D	D	*	*	*	*		
	3-d			*									*	*
	4-a			*									*	*
	4-b			*									*	*
	5-a													
	5-b													

D - dummy resource required

Table 2. Module Linkages

Because of this flexibility, the numbering of the individual Q-GERT nodes is variable, depending upon the order in which the modules are assembled to create a model. Therefore, Q-GERT input cards could not be created for the modules. Instead, input card templates (Fig. 3) were created. These templates contain the framework for the input cards, with node numbers in relative form. Activity times are represented by the underlined letters d and p which indicate distribution type and parameter set. Probabilities are indicated with an underlined pr and resources by underlined capital letters. A user's manual which accompanies the modules explains, in a step-wise fashion, how to create the input cards from the templates. The node numbering problems could have been eliminated if a subnetwork capability was available, but it would still be necessary for the user to complete the activity and resource information, since this is specific to an individual application.

REG,+1,1,1*	
ACT,+1,+2,d,p,+1*	time to move workpiece to AD
QUE,+2/AD-Q,(10)+3*	
ALL,+3,,A,1,+2/+4*	A: automated device resource #
REG,+4,1,1*	
ACT,+4,+5,d,p,+2*	task time
REG,+5,1,1,P*	
ACT,+5,+6,(6)+3,,pr*	task successfully completed
ACT,+5,+4,d,p,+4,,pr*	system recovery (task)
ACT,+5,+4,d,p,+5,,pr*	manual recovery (task)
FRE,+6,,A,1*	A: automated device resource #

Figure 3. Module 2-a Input Card Templates

The first model which was constructed was for the demonstration cell. It is a simple model, containing only 4 modules: the start and end modules, a part pickup module, and a module representing the robot working with an automated device. The flow diagram for the model (Fig. 4) separates these modules by the use of the dashed lines. The circles represent the robot resource, and the squares represent staging areas.

4. EXAMPLE

The following example shows how a simulation model of a system is constructed from the modules. A flow diagram only will be shown. Once the modules are selected and arranged, the task of creating the Q-GERT input cards requires only the renumbering of and addition of data to the Q-GERT card templates.

The system to be modeled consists of a single robot, three automated devices, and three staging areas. Workpieces are oriented at the first staging area, then picked up by the robot and loaded onto a lathe for rough sizing. When this operation is completed, the robot unloads the workpiece, then loads it onto a second lathe for finishing. Because the roughing operation is faster, intermediate storage is provided between the two lathes. The robot unloads the second lathe and inserts the workpiece into a grinder, which ejects the completed workpiece onto a conveyor. The conveyor then removes the workpiece from the cell.

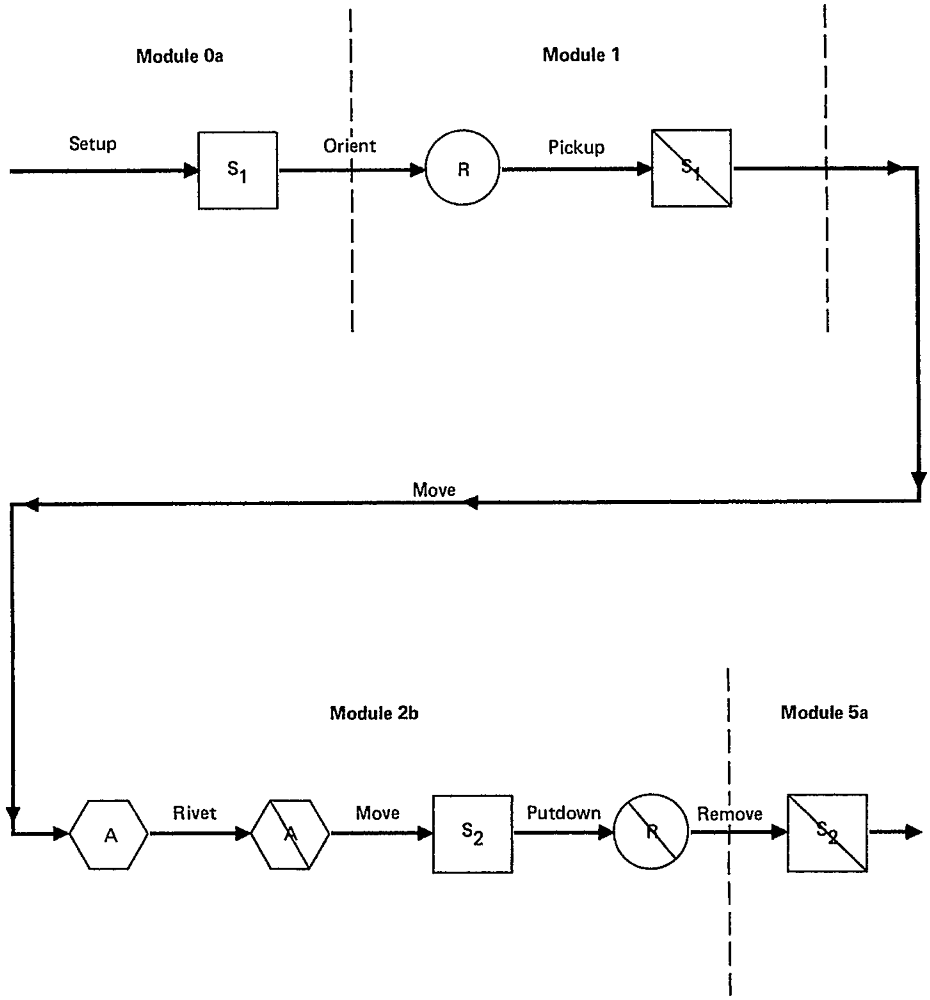

Figure 4. Flow Diagram for Demonstration Cell

Fig. 5 contains the flow diagram for this module. The semicircular block with dashed lines represents a decision point, at which the workpiece is routed to either the automated device or intermediate storage. The triangles represent dummy resources; these are required to prevent system deadlocks. The dashed lines separate the modules of which the model is composed. Since either set of start and end modules could be used, depending on whether the hierarchy is to be modeled or ignored, these modules are not included in the diagram.

5. SUMMARY

A system such as this is a viable means of providing simulation capability to persons with limited training in modeling. It is easy to use and has a reasonable amount of flexibility, but is limited to one type of system and thus does not possess the generality of a simulation language. It does, however, provide the user of a system with the means by which he can construct a model, and also with a higher level communication vehicle than a Q-GERT network model.

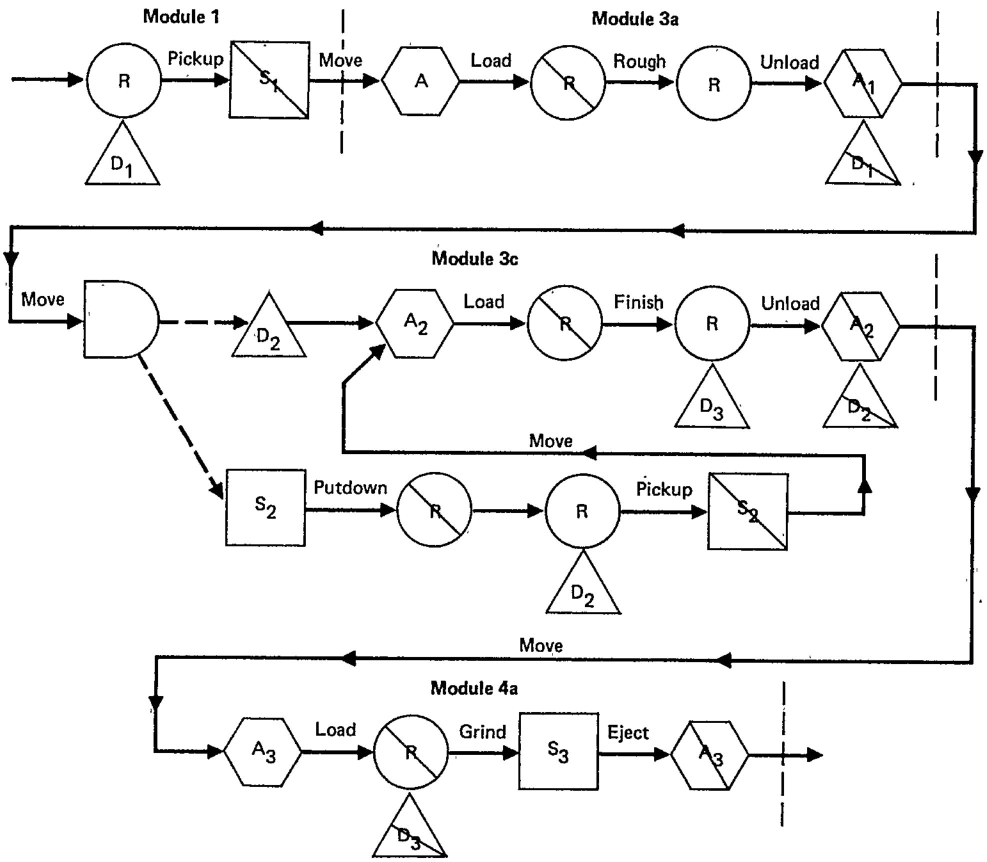

Figure 5. Flow Diagram for Three Machine System

REFERENCES

Mayfield, Jerry (1979), "'Factory of the Future' Researched", Aviation Week and Space Technology, 110, 10, pp. 35-37.

Pritsker, A. Alan B. (1979), Modeling and Analysis Using Q-GERT Networks, 2nd ed., Halsted Press, New York, 456 p.

Sadowski, R. P., Medeiros, D. J., and Welch, T. L., (1979), "A Q-GERT Simulation Model for Computer Controlled Robotic Manufacturing System", ORSA/TIMS national meeting, Milwaukee, Wisconsin.

Sadowski, R. P., Starks, D. W., and Medeiros, D. J., (1979), "Modeling Computer Controlled Robotic Systems for Batch Manufacturing", Proceedings of the 30th Annual AIIE Conference, San Francisco, California, pp. 562-565.