Proceedings of the 1984
Winter Simulation Conference
S. Sheppard, U. Pooch, D. Pegden (eds.)

SIMULATION OPTIMIZATION FOR DECISION SUPPORT IN OPERATING A ROBOTIC MANUFACTURING SYSTEM

C.C. Chu and J.J. Talavage
School of Industrial Engineering, Purdue University
West Lafayette, Indiana 47907

ABSTRACT

The operation of a robotic manufacturing system can be a complex task for which little experience is now available. Simulation has often been used as a means of modeling large complex systems. Optimization methods use such models to make good choices for system parameters. This paper describes a simulation-optimization approach combined with pattern recognition to develop an operating procedure for a manufacturing system which contains robots. This procedure is adaptive in the sense that it is updated on a periodic basis to account for changing shop load and pending orders.

INTRODUCTION

An automated manufacturing system usually consists of three major components - machine tools, material handling devices, and computers to perform control functions. The application of automation to manufacturing systems has lead to better product quality, and faster production. These improvements are offset by the technological complexity in design and operation of these manufacturing systems.

Computer simulations have been applied to such complex system analysis problems for a long time. The reason for using simulation in automated manufacturing is its capability to model the behavior of nonlinear discrete-parameter dynamic systems. Recently, in the control of Computer Integrated Manufacturing Systems (CIMS), research efforts have been directed toward making simulation a significant part of the decision support system. This avoids direct trial-and-error experiments which would be prohibitively expensive and time consuming in order to control a CIM system.

In general, simulation is used for systems where great detail of modeling is necessary to make correct decisions. A disadvantage of using simulation is that the modeler must provide large amounts of data, some of which are system parameters that are difficult to specify. Thus, one of the current developments for simulation is to integrate other software to make simulation more powerful and easier to use. Among those software types, stochastic optimization has been integrated with simulation. Simulation by itself is not an optimizing method. A production controller who would use simulation to, e.g., maximize number of parts produced, often finds that he must iteratively specify many alternative configurations and operating policies by using his intuition without any assurance of approaching an optimal or even satisfactory solution. Combining optimization algorithms together with simulation (either on-line or off-line) can ease the controller's burden by automatically performing some of his search tasks in a fast and efficient method.

Law and Kelton [4] discussed the use of statistical experimental design and optimization techniques in a computer simulation model. They pointed out that carefully "thought-out" or "designed" experiments can be much more efficient than a "hit-or-miss" sequence of unsystematic simulation runs. The technique suggested by them is known as response-surface methodologies (RSM). The unembellished RSM has a couple of disadvantages. First, the RSM method is generally used with a simulation model to evaluate an objective function several times for each finite difference required along each factor or variable. This may lead to long computation times if each evaluation involves running a large simulation model. Secondly, Law and Kelton indicated that there can be no guarantee that the result of a RSM procedure will always identify a truly optimal system design.

In order to improve the efficiency of the combined optimization and simulation approach, Azadivar and Talavage [1] developed an algorithm called SAMOPT. The algorithm will optimize the response function of simulation models, even for systems that exhibit stochastic behavior. Unlike RSM, SAMOPT evaluates the objective function via a stochastic simulation only once for each finite difference. Secondly, under certain weak conditions, SAMOPT provides convergence to the optimum. Comparison of results has also shown that SAMOPT provides better solutions with less computation time in more cases than commonly used RSM.

In this paper, the SAMOPT optimization procedure will be used in conjunction with a SLAM simulation model of a CIM system, which contains such automatic devices as robots, to develop a control procedure for the system operating parameters. These control parameters include the robot movement speeds, the robot picking policies, and the speed of assembly.

THE INTEGRATION OF SIMULATION AND OPTIMIZATION TO DERIVE A CONTROL PROCEDURE

The Real System

The real CIM system considered in this paper was designed for an automatic material handling function which can link the production and assembly functions of an overall CIM system in the future. The physical facility intended for this system is still under development. The major hardware components include full scale electric robots, a CNC machining center, a bi-directional free flow conveyor, a carousel conveyor, two bar code automatic identification systems, and links from some of this equipment to a remote supervisor computer. Figure 1 shows the CIM system configuration.

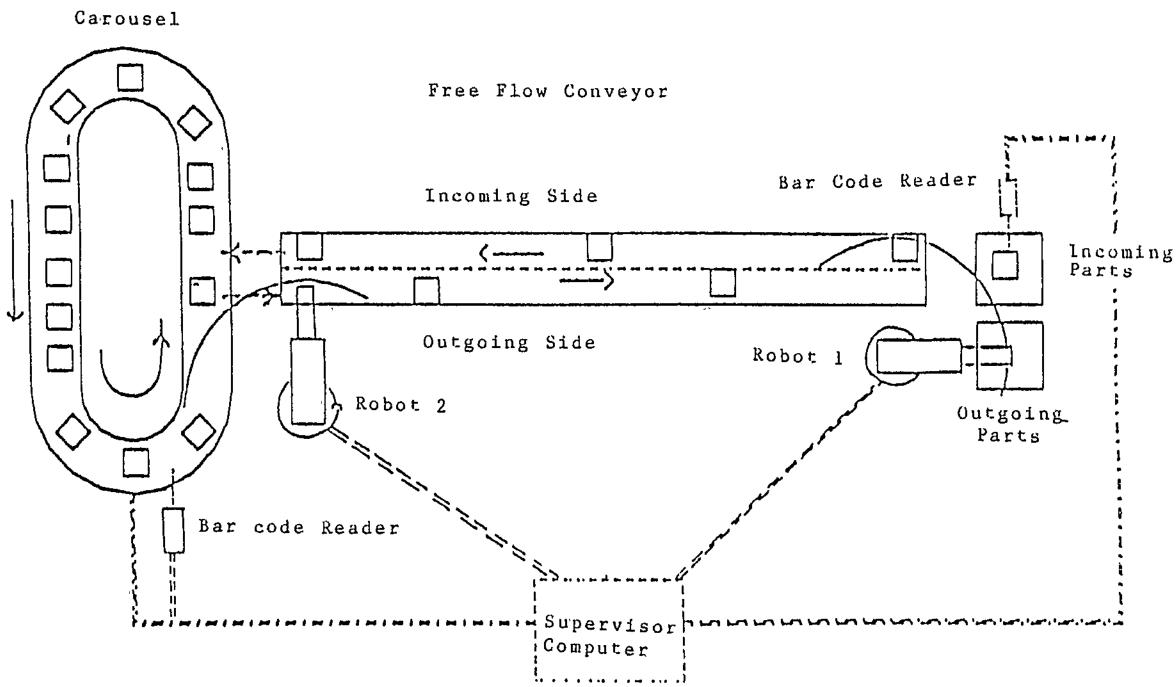

Figure 1: The Schematic Diagram of the Example CIM System

The material flow within the defined CIM system can be described as a fixed route. The incoming parts consist of several part types with an associated priority for each type. The interarrival time and product-mix may vary over time. As a part is entering the system, it will first be deposited at the auto-identification system which verifies its identity. After it is identified, the part enters the turn table and waits until the first robot is available to move it to the free-flow conveyor. It should be noted that each part type requests different robot handling movements. A path of robot movements can be generated by retrieving and executing the specific cycle for that particular part type. This cycle is defined by predetermined movement in space. After the part is on the free-flow conveyor, it moves to the other end and waits for the second robot to move it to the automatic assembly carousel. After it is on the assembly carousel, an assembly operation will be performed on that part. The assembled part is called an assembly and is assumed to be a different part type from its original. Thus, this assembly will require different robot movements for its handling. The assembly then follows the reverse sequence of robot handling and flows in opposite direction to leave the system.

Control of the Real System

It is desired to produce as many parts from the example system as possible without excessive wear on the robots. To do so, the robot speeds, assembly speed, and robot picking policies will be varied from time to time based on the system state in order to maximize production. The choice of those parameters will be based on off-line results obtained from simulation optimization.

The Simulation

A simulation model based on SLAM simulation language was developed to represent the system. The simulation model utilizes network, discrete, and continuous modeling concepts. In order to consider the interactions among material handling equipment in this model, the detailed movements of both robots and conveyors are simulated in continuous modes. That is, a robot loading/unloading operation is described by using a continuous path (i.e., trajectory) which is formed by a set of points. The overall configuration and many complex logical decisions of the simulated CIM are described by using the associated discrete and network model. For more information about the model see [3].

In the following, several factors associated with optimizing the simulation model of the CIM system for purposes of control will be discussed.

System Objective. A system objective represents a desired goal of operating a system for a given time interval. There are many possibilities for setting this objective such as to minimize the average time in system for a specific part type, or to maximize the utilization for certain equipment types, or others, Our objective will be to maximize production and robot utilization.

Control Interval. As mentioned, the system objective is measured over a given time interval. This time interval is called a control interval in this paper. If this interval is too long, the system may not react as quickly as the changing environment. If the interval is too short, the system may not be stable enough for making an estimation of the system performance. The general rule, which is suggested here, is to choose a sufficiently long period, but one for which the system will still be in its transient state. (For purposes of pattern recognition, we want the system to "remember" its initial state.)

Decision Variables. The decision variables are the parameters that govern the operation of the CIM system. These variables can be any factors which affect the operation of the system. For example, they are robot picking rules, assembly speed, and robot speeds in this paper. Generally, the more variables one uses, the more time and expense associated with the control decision. Therefore, it is better to consider only major factors.

System Status. The status of the system is based on an instantaneous measurement from many of its components (i.e., equipment). One specific example of system status can be described by the initial status for resources, the initial contents of queues, and the initial events in the event calendar. In this paper, the system status is assumed given entirely by the queue contents (so, e.g., the status of the robots when a decision is implemented is assumed to be unimportant to later operation over the control interval).

The SAMOPT Optimization Algorithm

The optimization algorithm (SAMOPT), used in this research, was developed by Azadivar [2]. The basic idea of SAMOPT is to optimize the response function of a simulated system by using a stochastic approximation method. For more details about the algorithm, the reader is referred to [1,2].

Interfacing the SAMOPT algorithm with the simulation model of our CIM required little effort; however, there are a couple of details worth noting:

1. Input: SAMOPT can handle optimization problems with up to 10 decision variables and 30 linear constraints. (Our problem has five variables and no constraints.) These limits may be increased by increasing the dimensions of the arrays.

2. Computer: In this paper, the optimizations were performed on two computer systems - the CDC 6500 computer and CYBER 205 supercomputer. In the Purdue CDC 6500 computer, there are restrictions caused by the overlay structure of SLAM. It was necessary to put the SAMOPT model in the PROGRAM MAIN and the simulation model in the subroutines. Thus, the execution of the simulation model is called by SAMOPT from the main program. The objective function was computed at the end of each simulation run and passed back to the main program for comparison. However, in the CYBER 205, there were no such constraints.

THE CONTROL PROCEDURE

The control procedure can be separated into two phases. The first phase was performed off-line to create the control database which contains the control information for the real system operation. The second phase performs on-line real time control of the real system operation by applying the prestored information from the first phase.

In the first phase, the integrated simulation and optimization approach was used to create the control database. Based on this integrated model, the control database was built in the following steps:

1. A subset of the state space for the model was selected by observing "typical" state trajectories as the model was exercised for "typical" conditions.

2. The state space sample was analyzed by using a clustering analysis to group the similar states together. Then, a pattern merging algorithm was used to select only several representative patterns within each grouped cluster.

3. The optimal control values over the control interval were found by simulation optimization for each representative pattern in each grouped cluster.

After this control information is available, the second phase of on-line control can proceed. The procedure of this on-line real time control is as follows:

1. Identify the current system status from measurement of the specified variables (e.g., work-in-process at the turn table, the conveyors, and the carousel in this paper).

2. Perform the information retrieval by using a nearest neighbor pattern recognition algorithm to obtain the nearest representative state to the current state.

3. Apply the control alternative, which is the set of simulation-optimized decision variables associated with the retrieved representative case, to the real system.

4. At the end of the control interval, re-do from step 1.

RESULTS AND DISCUSSIONS

The developed control procedure has been used in the example CIM system. The real time control function has been validated in a computer controlled Puma robot. The detailed results can be obtained from [3].

Table 1 shows an evaluation of the developed control strategy, which is adaptive since it changes with the state every control interval, to other non-adaptive control strategies in the example CIM system.

This evaluation is based on the same simulation model described previously where the simulation model was tested for a 10 hour period which is equal to 36000 simulation time units. Here, a variable input situation is assumed. That is, during the 36000 simulation time interval, the input rate of parts changed every 600 time units. The primary criterion to evaluate the different strategies is the total throughput of all part types and the utilization of the two robots.

There were four strategies compared in this evaluation. The first strategy called "Fixed Fast" uses the fastest robot speeds at all times. The second strategy, which is also non-adaptive, uses a slightly slower robot 2 speed. The third strategy, which is the developed adaptive strategy, will adaptively adjust its control parameters every 600 time units to react to the changing input situations. Finally, the fourth strategy, which is non-adaptive, is based on the averaged results of the adaptive strategy.

In Table 1, when a varying input stream is used, the developed adaptive strategy shows its superiority to the other three non-adaptive strategies. For example, based on the similar throughput rate, the adaptive strategy can handle as many parts as the fixed-fast rule, which uses the fastest robot speed, but can provide better robot utilization and slower average robot speeds. On the other hand, if the comparison is based on a similar robot utilization, the adaptive strategy provides more thorughput rate than the fixed average strategy. Thus, it can be concluded that under a varying input situation, the adaptive strategy can find a good balance point among the throughput, robot utilization, and robot speeds.

From this evaluation, we can see that the developed control strategy can be particularly useful when a CIM system is operated in a dynamic environment.

CONCLUSION

This paper has presented an integrated simulation and optimization approach to create the control information to support the operation of a robotic manufacturing system. The same approach could be used to build a decision support system for general computer integrated manufacturing systems.

REFERENCES

1. Azadivar, F. and Talavage, J., "Optimization of Stochastic Simulation Models", Mathematics and Computers in Simulation XXII, North Holland Publishing Company, 1980, pp. 231-241.

2. Azadivar, F., "Optimization of Stochastic Systems Through Simulation Using Stochastic Approximation Method", Ph.D. Thesis, Purdue University, December 1979.

3. Chu, C., "An Adaptive Decision Making Methodology for Material Handling Equipment in a Computer Integrated Manufacturing System", Ph.D. Thesis, Purdue University, December 1984.

4. Law, A.M. and Kelton, W., Simulation Modeling and Analysis, McGraw-Hill, Inc., 1982.

Table 1: An Evaluation of the Developed Control Procedures

Comparison of Varying Input Rates (High Volume Input)				
	Fixed Fast	Fixed Medium	Adaptive	Fixed Average
Rbt 1 Priority	15.63	15.59	---	14.75
Rbt 2 Priotity	2.921	3.111	---	4.870
Rbt 1 Sp Ratio	0.500	0.500	---	0.6533
Rbt 2 Sp Ratio	0.500	0.5789	---	0.6539
Assembly Speed	1.377	1.375	---	1.508
No. of Balking	0	20	0	121
Thruput Type 1	952	928	949	887
Thruput Type 2	560	545	559	513
Thruput Type 3	390	381	390	365
Avg TIS Type 1	145.2	267.5	175.8	337.0
Avg TIS Type 2	167.8	287.8	194.8	367.1
Avg TIS Type 3	160.2	321.1	186.2	339.1
Avg No. in Sys	8.502	18.02	10.18	17.40
Avg No. in Tur	2.906	4.390	3.415	9.675
Avg No. in Cvb	1.238	5.163	1.648	1.766
Avg No. in Car	1.667	6.678	2.201	2.714
Avg No. in Cvd	1.318	0.828	1.880	2.181
Cur No. in Sys	1	28	5	16
Cur No. in Tur	0	5	0	10
Cur No. in Cvb	1	9	2	1
Cur No. in Car	0	13	1	1
Cur No. in Cvd	0	1	1	2
Robot 1 Util.	0.7671	0.7507	0.8606	0.8724
Robot 2 Util.	0.7641	0.8312	0.8442	0.8697