Proceedings of the 1983
Winter Simulation Conference
S. Roberts, J. Banks, B. Schmeiser (eds.)

SINDECS-R: A ROBOTIC WORK CELL SIMULATOR

A. P. Robinson and S. Y. Nof
School of Industrial Engineering
Purdue University
West Lafayette, Indiana

A GASP based simulator has been developed as a tool for examining both robotic work cell layout problems and part flow control problems. The unique modeling requirements for such a simulator are described and demonstrated.

1. INTRODUCTION

A robotic work cell combines a group of machines and inspection equipment with single or multiple robotic arms. The arm(s) can act either as a materials handling device, loading and unloading machines and transporting parts, tool, or fixtures between machines, or can play a more active role in part processing by acting as a jig or fixture or making component assemblies. The development of sensors and sensor compatible controllers leads to increased use of robots in programmable, versatile production environments. Sensors keep the controller informed of the status of the robots work environment. Increasing costs of human labor are also influencing industry towards more intelligent robotic applications.

Performance of a robotic work cell depends on a number of variable factors. The logic of part flow through the system contributes greatly to any work cell/processing system. Also, specific to robotic cells, the robot's flexibility in responding to the system's demands must also be considered to accurately evaluate work cell performance.

The purpose of this article is to describe a generalized robotic cell simulator, called SINDECS-R, that has been developed to model robotic work cells based on input information about the above considerations. It is intended as a tool to aid in approaching the optimal design of a new robotic cell, and in structuring part flow through a new or existing system such that performance is improved.

2. JUSTIFICATION AND BACKGROUND

SINDECS-R has been developed specifically to investigate the effects of layout and flow control logic on the performance of a robotic cell production system. Investigations include examination of system performance with various combinations of control rules for a given layout. Another objective is the investigation of general cell layout optimization methods to determine their long term advantages.

SINDECS-R has been developed from SINDECS (Khator, Nof 1983) (Rovira, Nof, Khator 1983), which emphasizes features for user defined flow control logic. Readers are referred to the references for further details about SINDECS. Applying the simulator control logic to robotic work cells is useful for examination of system factors beyond simple performance measures. These features have not been addressed before in other robotic simulators, and are described below.

3. MODELING CONCEPTS

The original SINDECS is a simplified material flow control simulator. It models a facility as a given number of single or multi-server stations interconnected by an infinitely available flexible materials handling device. Some of the simplifying assumptions: when a part is dispatched to a multi-server station the choice of particular server within the station is assumed inconsequential. Since the transporter is modeled with infinite capacity the number of parts which can be transported at one time is unlimited. Thus, a part does not have to wait for the transporter to become idle before it can be moved between stations. Such assumptions can be quite reasonable when material handling is carried out be conveyors, but have to be modified for robotic simulation. However, the basic structure of SINDECS, mainly, the library of flow control procedures, has been retained.

Some of the modeling concepts of SINDECS-R differ from those of SINDECS:

(1) In SINDECS-R the transporter (robot) is no longer considered to be infinitely available, but instead has finite capacity and a logic which defines its assignment to parts requesting service. If all robots are busy, SINDECS-R stores these requests for transportation and processes them later in a logical order as defined by the user.

(2) In multi-robot stations each robot does not need to be able to service each machine. In this case it is possible that a part may require the coordinated effort of two or more robots to make its way through the cell. Of all the robots which can reach a part at a station, a decision must be made to determine which robot to dispatch to that particular part. Consideration of collision between robots with overlapping work areas in multi-robot stations is left to the user for the sake of simplicity.

(3) In robotic cells, it is possible that the processing at a station might require the presence of a robot, for example to hold the part in place in the lieu of a jig or fixture, for instance, in buffing or deburring operations. Moves must be classified by the users as to whether or not the robot remains at the station after part delivery, and if so, the amount of time spent until the robot is freed must also be supplied by the user. To simplify programming and user input it is assumed that these robot processing requirements occur at the beginning of a part's processing at a station.

(4) A new capability to consider quality control has been added in SINDECS-R. It does not pertain strictly to robotic cells but extends the capabilities of SINDECS itself. Quality control is modeled by probabilistically routing some parts back into the cell for reprocessing, some parts out of the cell to be considered scrap, and some parts out of the cell to be considered complete.

4. PROGRAMMING LOGIC

SINDECS-R considers four major types of events, as shown if Figure 1:

1. "Request for robot service"
2. "End of robot service"
3. "Part arrival to a machine"
4. "End of machine service"

To undertake the first extension two additional event types were added; "arrival of a part requesting robot service", and "end of robot service". The arrival event is a conceptual arrival not a physical arrival. The part itself

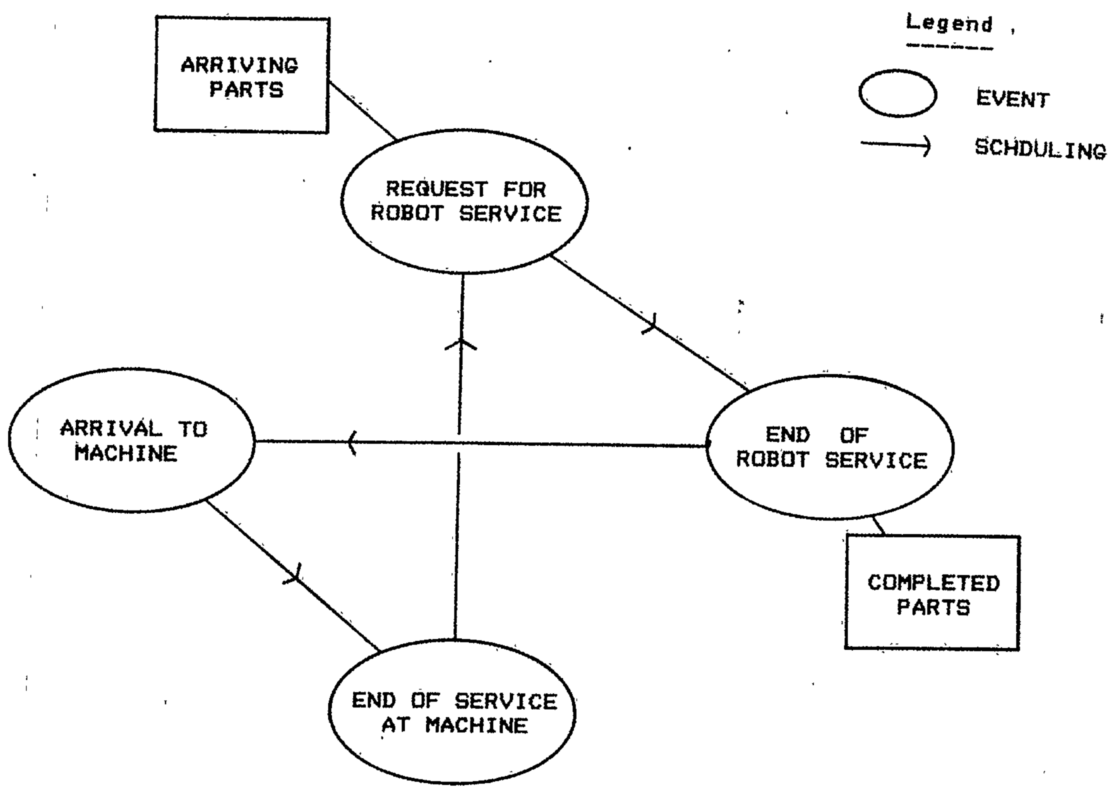

Figure 1: Events and Scheduling

does not queue up for robot service, instead its request arrives to the robot after the part completes service on some previous machine. Upon completion of a parts processing, the controller first checks which robots are physically capable of moving it between its current station and its destination. Of the capable robots, those which are idle are identified. If none is idle, the part remains at the machine, and its request is put in a general requesting part queue; it will be removed and serviced later by a robot which becomes idle. If one robot is idle, an "end of robot service" event is scheduled after the time for the robot to move to the part's current station, unload it, reload the machine if parts are waiting, and move the part to its destination. If more than one capable robots is idle a user chosen decision rule is applied to determine which robot to dispatch. An "end of robot service" event is scheduled as before.

If no single robot is capable of moving the part from its current location to its destination, two robots are requested. One moves the part from its current station to an intermediate position. The second robot moves the part from this position to its destination. The intermediate position is physically capable of holding parts for cases when the second robot is not available. This can be expanded to model the cooperation of any number of robots.

Note that if a queue of parts exists at the machine at which the requesting part has just completed service, then additional logic is required for part to machine selection. A part must be chosen from the queue using a decision rule as already available in SINDECS Since a robot with a double gripper can first unload and then load a part the time to load the new part must be included in scheduling the robot's end of service. Statistics are collected not only on the proportion of time a machine is operating, as in SINDECS, but also the proportion of time it holds a part in its chuck waiting to be unloaded by a robot. Comparison of these statistics can provide insight into the time the parts wait for robot service.

The "end of robot service" event triggers a "part arrival to machine" event (carried over from SINDECS). The part is either queued or processing is started. The robot then examines the queue of parts which have requested service. If the queue does not exist, or the robot cannot service any parts which are in the queue, the robot becomes idle until its service is requested. A decision rule is applied to the parts that the robot can service to choose which particular part will be serviced. An "end of robot service" event is scheduled as before. A graphic illustration of the scheduling of events is presented in Figure 1.

The second extension to SINDECS, the holding of robots while processing, is also handled in the "end of robot service" event. The proportion of processing time a part requires a robot at a machine is supplied by the user for each part type at each machine. If the proportion is greater than zero the release of the robot is scheduled after the time of the proportion multiplied by the service time at this machine.

The final extension, quality control, is easily programmed through user specification of the proportion of parts which require rework and the proportion which must be scrapped for each part type. If the part requires rework it is routed back probabilistically through some user specified portion of its normal processing. Parts determined to be scrap are routed out of the system and an identical part, beginning its processing, is generated.

5. DECISION RULES

SINDECS allows a user to specify the flow logic rules applied to four types of decision rules that are made repeatedly throughout the simulation. Together these rules define the operation logic of the system. These decision rules include:

1. Part type selection for initial entry into the system

2. Part type selection for replacing a part leaving the system

3. Allocation of parts to machines

4. Selection of a process for each particular part

The variety of decision rule choices available for each of these decision is presented in Khator and Nof (1983), and is summarized in Figure 2.

Two new decisions are introduced for robotic simulation:

1. Choice of robot to handle a particular part

2. Choice of part to be handled by a particular robot

The "robot to part" decision determines which of several idle robots should serve a part requesting service. At present, the decision rules available for this decision include choosing 1. the robot which is currently closest to the parts current location, 2. the robot which can perform the move in the shortest time, 3. the robot with the lowest utilization so far, and 4. the robot with the highest priority. Priority is assigned to robots through user input.

The second decision , "part to robot", determines which of a number of parts requesting service should be serviced by a robot which has just become idle. Strategies include choosing 1. the part with the highest user defined priority, 2. the part which has been waiting the longest, 3. the part which has been in the system the longest, 4. the part with the nearest due date, 5. the part which will require the robot for the shortest or longest amount of time, and 6. the part which is closes to the robots current location.

6. INPUT REQUIREMENTS

Input parameters which describe the facility layout for SINDECS include the number of stations, the number of machines at each station, and

No.	Decision	Available in SINDECS	Available in SINDECS-R
1	Part type selection at time of initial entry to system	Yes	Yes
2	Part type selection for replacing finished part	Yes	Yes
3	Allocation of a part to a machine	Yes	Yes
4	Selection of a process	Yes	Yes
5	Choice of robot to transport a part	No	Yes
6	Choice of part to be transported by a robot	No	Yes

Figure 2: Flow Decisions in SINDECS and SINDECS-R

transportation times between stations. Actual parts and processes are defined by the number of part types and the number of different processes for each part type, i.e. the number of different paths through the facility the parts can take to be processed. The processing times at each station are also input. These times are supplied for each production process for all part types.

Four input segments have been added to this input for SINDECS-R. The added segments include input of:

1. Motion Time Between Stations
2. Unload and Reload Time at a Single Station
3. Delayed Release Operations
4. Part Quality

Input complexity has been kept to a minimum to reduce the risk of input errors.

The input of between station motion times is achieved by specifying initial station, destination station, and motion time. All unspecified between station move times default to zero. Unload and reload times are similarly input, save that only one station is specified. Unload time denotes the time required to only unload the machine before transporting the part (i.e.,for the case where no parts are waiting for processing at the unloaded machine.) Reload time is the time required to unload the finished part and load a waiting part before transporting the unloaded part to its next machine. Unspecified times default to zero.

Delayed release operations require more data from the user. Data required are part type, operation number, robot number, and fraction of run time before release.

Part quality is specified by supplying the proportion of good parts, proportion requiring rework, and proportion scrap. Those parts requiring rework are sent through some number of operations of the process just completed. The number of operations is chosen randomly by the software. The sum of the fractions must equal 1.0.

The input discussed above describes the basic robot cell system to SINDECS-R. Some decision rules, if used, require additional data. For instance, if the robot selection rule is the generalized priority rule, then certain robots are preferred by all part types. The user must input a robot serial number and its priority. Priorities are all relative. The idle robot with the highest priority will be selected to service the part. The default priority value for unspecified robots is zero.

Another robot selection rule also utilizes robot priorities, but with priorities being part specific. For this case the user supplies the above input for each part type. Again, priorities that are unspecified to default to zero.

7. OUTPUT SUMMARY

SINDECS supplies statistics on production rates and time in the system, both overall and by part type. Also supplied is utilization of each server, broken down by machine type.

SINDECS-R offers two statistics on machine utilization; time running statistics and time occupied statistics. Run time statistics reflect the time a machine is actually machining a part. Occupation time statistics reflect the time a machine is occupied with a part, whether the machine is operating or idle. Comparison of these statistics gives insight to the reaction time of the robots to parts' requests for service.

Statistics on robot utilization are also offered. These statistics reflect the percentage of time a robot is busy.

8. EXAMPLES

In this simple example five machines are tended by two robots. Figure 3 shows the machining processes and process times for the two parts introduced to the system. Defaults are used for all decisions rules. The simulation of this system was run with two configurations of robot service. In the first run both robots could service all machines and no cooperation was necessary. In the second neither robot could service all machines and cooperation between the two was required; robot 1 serviced machines 1, 2 and 3, and robot 2 serviced machines 4 and 5.

Results of the simulations showed that the first configuration is superior in all respects to the second. The overall production rate of the first was 25% higher than the second. Machining utilizations for the first configuration averaged 43% higher than utilizations for the second, while machine occupation utilizations for the first configuration averaged only 30% higher than for the second. Robot utilization behaved as expected. Configuration 1 produced utilizations of .75 for robot 1 and .51 for robot 2. Configuration 2 increased the utilization of robot 1 to .98 due to robot 1 servicing the most heavily used machines. Utilization of robot 2 sank to .15 for the same reasoning.

Despite the operational superiority of the first configuration, however, configuration 2 of this example is probably the more realistic of the two. In practice assigning certain machines to be serviced by only one robot is a straightforward way to prevent collision problems. Furthermore, applying robots with a shorter reach range would probably yield significant savings in robot equipment costs. If the segregated cells of configuration 2 were preferred the SINDECS-R user could experiment with different machine assignments for the robots in order to balance their use and increase production.

PART NUMBER	PROCESS NUMBER	MACHINE NUMBER	MACHINING TIME
1	1	1	17.0
		2	33.0
		5	41.0
1	2	1	17.0
		4	10.0
		5	45.0
2	1	3	51.0
		2	25.0
		1	17.0
		3	25.0

Figure 3: Parts and Processes of Example System

ACKNOWLEDGEMENT

SINDECS and SINDECS-R have been developed under an NSF Grant for Modeling Discrete Industrial Flow, the program for Production Research and Technology, and a grant from CIDMAC, Purdue's Computer Integrated Design, Manufacture, and Automation Center.

REFERENCES

Khator, S., Nof, S. Y. (1983), A Flow Controlled Modeling Approach For Computerized Manufacturing Systems, to appear in Material Flow.

Rovira, R., Nof, S., Khator, S. (1983), User's Guide to SINDECS; Modeling of Discrete Industrial Flow, Technical Report School of Industrial Engineering, Purdue University, West Lafayette, Indiana, June.