SIMULATION AND ANIMATION OF AN ASSEMBLY SYSTEM

Bahram Keramati, Christine M. Kelly-Sacks
Corporate Research and Development
General Electric Company
Schenectady, NY 12301

Gregory L. Tonkay
Pennsylvania State University
University Park, Pa 16802

A software package was developed for eventual use by manufacturing engineers responsible for the design of an automated assembly loop. The loop consists of a number of cells through which palletized parts flow. A simulation of this system was developed which includes the effects of blocking, machine down times and pallet transport time. In addition to the statistical information usually obtained with any simulation, an animation of the process was also developed which provides the manufacturing engineer with an interactive-graphic environment to examine the dynamic performance of the system.

1. INTRODUCTION

Simulation is a very powerful tool for the detailed design of manufacturing systems. Unfortunately, manufacturing engineers are typically unaware of this fact and tend to rely on vendors for the proper design of a system. A simulation that is properly constructed can not only assist the engineer in the design process, but it also requires the engineer to think about all aspects of the problem in advance. It is not uncommon to find very large manufacturing systems that have been designed and constructed without any comprehensive modeling that do not perform according to the specifications intended for them. Even in such situations, it is instructive to develop appropriate simulation models for the system and use the model to explore fix ideas.

For a simulation to be effective as a design tool, it needs to go beyond providing statistical summary reports. Useful as these are in communicating what happens in a given system, they provide very little information about why things happen. All viable simulation languages contain some type of a trace feature that allow the user to march along time and examine the performance of the system "line by line." In addition to being a tiring and time-consuming task, much of the inter-relationships and the proper view of the overall system is lost.

In this paper, we describe, as an example, a simulation of an assembly operation and the animation of the simulation results to allow the examination of the dynamic performance of the system in an efficient manner. Both the simulation and animation were designed with the end user, i.e., the manufacturing engineer responsible for the total system design, in mind. We will discuss the nature of the assembly system in the next section, followed by a section on the simulation of the system. The animation of the simulation results will be discussed in section 4. Finally, conclusions will be presented in section 5. Since the animation portrays the dynamic performance of the system via computer graphics, a demonstration of it will be given at the Conference. It will only be described in words in section 4 of this paper.

2. DESCRIPTION OF THE ASSEMBLY SYSTEM

The assembly system under investigation has a loop configuration. A number of cells are located around a track and the part to be assembled is placed on a pallet at the first cell. Each cell contains a number of parallel and identical machines which process the palletized parts. As each part is processed, it is sent to the next cell on the loop. There is limited room between the cells on the loop, leading to blocking of the previous cell if the available space fills up with pallets. Due to the nature of this particular problem, the pallet transport time is an important factor since the individual cell process times are typically only a factor of 2 or 3 higher than the time needed to transport a pallet between two cells (if no other pallets are in that queue). The part is removed from the pallet in the last cell and the empty pallet queues up ahead of the first cell.

Each machine in a cell is allowed to have a number of stages, such as a dial-index machine. In this particular case, most machines have a single stage, but some have up to 12 stages. In addition, each machine is subject to down times determined from estimated data on mean time between failures. The planned maintenance does not enter into the normal operation of the assembly loop since all such maintenance is to be performed during a third shift where the loop is not operational. Considering all the factors cited above, each machine in the loop can be in one of four states: starved for a pallet or "idle;" blocked; down for maintenance; and, busy.

At each cell, some type of assembly operation is performed. This could be the addition of new machined or fabricated parts to the palletized part, the addition of incidental purchased parts, such as screws, or, no addition of parts at all. The last case may be a gauging or testing operation.

Prior to construction, the various parameters in the loop need to be quantified. Many questions exist, such as, the number of machines needed in any given cell, the track length needed between any two successive cells, the speed of the pallets, etc. The manufacturing engineers responsible for the design of the loop can obtain data from equipment vendors regarding the processing speeds, maintenance requirements, and other factors regarding each machine. In most cases, the data for unplanned maintenance is not available and some educated guesses need to be made about the frequency of their occurrence for each machine and the expected down time distribution once a failure occurs. In this particular case, such failures are mostly due to part jamming and other feed problems. It is then the responsibility of the engineer to assure that the entire system will perform according to the specifications established by management.

In the next two sections, we describe the simulation and animation tools that have been developed to assist the manufacturing engineer in the design of the assembly loop.

3. SIMULATION OF THE ASSEMBLY LOOP

Prior to any attempt at simulation modeling, consultation with the manufacturing engineers helped establish the type of simulation that would be most beneficial for their purposes. The assembly loop was part of a larger factory containing machining, fabrication and other assembly areas. All the assembly areas were generically similar in that the description given in the last section applied to all of them. What distinguished the different assembly loops were the details of the number of cells, number of machines per cell, pallet queue capacities, and the details of each machine such as cycle time and reliability. It was therefore decided from the outset that what was needed was a simulation package that would capture all the logic of the system in a rather general way and be adaptable to any of the assembly loops that needed to be investigated by data provided by the user. It was essential that the manufacturing engineers be able to run the simulation program themselves and draw their own conclusions. Technical support in experiment design and statistical interpretation of simulation results would be provided for them.

The proper level of detail to be included in the simulation was determined jointly with the manufacturing engineers. In a few instances, not enough detail had been included for their purposes and the simulation was expanded to include them. What finally resulted was a simulation with that captured all the relevant details important for the design of the loop.

Due to the constraints outlined above, the simulation was developed in the event scheduling world view of SLAM-II. Although simulation in event scheduling is by no means easier than what can be done with higher level simulation languages, the approach has several advantages for this particular application. First of all, there are no limitations to what can be modeled in event scheduling, since all the logic for the behavior of the system is coded in FORTRAN. Secondly, the manufacturing engineers were to be able to change the design variables, such as the characteristic of the machines and the pallet queue capacities, etc., conveniently, without the need to learn the syntax of a simulation language. Finally, the actual loop definition had to be data-driven since the simulation would be used for designing a number of different assembly loops. Event scheduling affords the programming flexibility to accomplish these objectives conveniently.

Two important concepts in simulations using event scheduling are the concepts of "event" and "attributes." Events are things that occur at an instant of time and change the state of some parameter in the system. Events themselves have no duration but they trigger activities which span a period of time. In event scheduling, one in strictly concerned with events as opposed to activities. Attributes are characteristics of events and entities. This is a very powerful concept since an event attribute can cause a generic event to become applicable to a particular subsystem purely by virtue of the event's attribute. For example, an example of an event is: "a machine just finished processing a part." It is easy to recognize that when such an event takes place, a set of instructions need to be carried out that do not much depend on which machine in the system the event refers to. It is possible, therefore, to write all such instructions generically just once and assure its application to the right machine by having the proper machine number as an event attribute.

Four events are used in the simulation of the assembly loop. These are: 1) a machine just finished processing a part; 2) a machine just finished maintenance and is ready to come back on line; 3) a down time just occurred for a machine; and 4) a pallet just joined the other pallets (if any) waiting to be processed by a machine. Each one of these events carry a number of attributes as follows. Events 1, 2 and 3 have as their first two attributes the cell number and the machine number within that cell. In addition, event 1 has an additional attribute which is the attribute of the pallet engaged in the machine. Event 4 has just two attributes which is the pallet attribute and the queue number in which the pallet is to be filed. The events and their attributes give an indication of the level of detail that has been simulated for this problem.

In order to use the simulation, the manufacturing engineer prepares a data file containing the characteristics of the system he wishes to study. A typical data file will contain the following information: total number of cells; number of machines per each cell; cycle time for each machine; pallet queue capacity feeding each cell; the initial number of pallets in the queue; the breakdown frequency and repair time distribution for each machine and other data specifying the simulation run time and the desired categories of output.

Two types of output result from the simulation. One is the standard "summary report" of SLAM-II, and the other is a file that contains all the information about the system at every instant an event takes place. This latter output contains the status of all the machines and the actual number of pallets present in all of the pallet queues. This output data file can be used for user-specific statistical analysis, plotting of the results, and the animation of the simulated system to be discussed in the next section

4. ANIMATION OF SIMULATED RESULTS

In order to obtain enough information for the detailed design of the assembly system, the manufacturing engineer needs to be able to examine the dynamic performance of the loop conveniently. Animation of simulated results is used for this purpose. An interactive-graphic computer program was written which displays the data contained in the detailed output file created by the simulation program. The program provides three levels of graphics as follows: 1) animation; 2) detailed plot of pallet queue contents on a selective basis as a function of time; and 3) histogram of cumulative machine states on a selective basis. We now briefly discuss each one of these levels in some detail.

At the animation level, the user sees a background of a loop with the proper number of cells and machines in each cell, identified by simple block diagrams. Pallets are represented by character graphics and their proper number in all the pallet queues is shown as a function of time. Time is depicted by a number in appropriate units which is refreshed upon each update. The screen updates are accomplished by selectively erasing only the information that is no longer valid and adding only what needs to be added. In addition to the pallets, machine states are also depicted symbolically, each symbol representing any of the four machine states described earlier.

Animation can be temporarily interrupted at any time for the purpose of entering any of the other two levels of graphical output. For example, the user may wish to examine the contents of several pallet queues as a function of time for a selected time period. This is easily accomplished by supplying the answers to the prompts. Similarly, histograms of machine states can also be observed which depict the fraction of time the selected machine(s) are in any of the four states up to the current time in the animation. Once these two levels of graphics have been sufficiently examined, the user may resume the animation at any time and move forward from there.

It is not possible to provide much more detail about the animation system in a paper of this sort. The authors will present a sample animation at the Conference which will demonstrate the interactive nature of the program.

5. SUMMARY

Simulation and animation provide very powerful tools for the design of manufacturing systems. These tools were applied to an assembly operation as an example. Specifically, the use of the event scheduling world view of simulation for the development of friendly simulations was demonstrated. It was also demonstrated that much information can be provided to manufacturing engineers in the form of graphics and animation, without which the actual examination of system dynamics would be very tedious. The tools have been quite well-received by manufacturing engineers responsible for the design of the system.