NEAR EAST UNIVERSITY

GRADUATE SCHOOL OF APPLIED AND SOCIAL SCIENCES

NEAR EAST UNIVERSITY LIBRARY 1988 - LEFKOŞA

QUALITY CONTROL USING FUZZY LOGIC IN FLEXIBLE MANUFACTURING SYSTEMS

Metin Bilin

Master Thesis

Department of Electrical and Electronic Engineering

Nicosia - 2003

NEU 26 56 19 82 84

Metin Bilin : Quality Control Using Fuzzy Logic in Flexible Manufacturing Systems

NEAR EAST UNIVERSITY LIBRARY 1988 - LEFKOŞA

Approval of the Graduate School of Applied and Social Sciences

Prof. Dr. Fakhraddin Mamedov
Director

NEAR EAST UNIVERSITY 1988

We certify this thesis is satisfactory for the award of the Degree of Master of Science in Electrical and Electronic Engineering

Examining Committee in charge:

Prof. Dr. Fakhraddin Mamedov, Committee Chairman, Dean of Engineering Faculty, NEU

Assoc. Prof. Dr. Adnan Khashman, Committee Member, Chairman of Electrical and Electronic Engineering Department, NEU

Assist. Prof. Dr. Ali Evcil, Committee Member, Mechanical Engineering Department, NEU

Assoc. Prof. Dr. Rahib Abiyev, Supervisor, Computer Engineering Department, NEU

ACKNOWLEDGEMENTS

'' First I would like to thank my supervisor Assoc. Prof. Dr. Rahib Abiyev for his excellent guidance and friendly relationship during the course of my study. I would also like to thank Prof. Dr. Fakhraddin Mamedov for his endless support. Finally I would like also to thank , Assoc. Prof. Dr. Adnan Khashman, Asist. Prof. Dr. Ali Evcil.''

ABSTRACT

Nowdays most of industrial factories are characterized by different types of output products. According to the demand of the clients the type of these output products can be changed by time. In such condition the reconstruction and scheduling of the work of factories are primary task. The flexible manufacturing systems allow to provide real – time planning, scheduling and control of the work of the factory.

The development of FMS including different automated intelligent workstation acquires great importance for some complex plants. In the thesis the analysis and development of FMS including intelligent workstations such as intelligent robots is considered. The evolution of computer controlled manufacturing systems are described, the role and functions of computers in FMS has been clarified. The types of FMS, its main integrating components, their functions are described. Software and hardware structure of FMS, their control functions are given.

The scheduling and control problems of FMS are described. The four scheduling algorithm for FMS are given. In the thesis the development of intellectual robot for FMS is carried out by using fuzzy technology. This robot is used in computer controlled manufacturing system for expanded polystyrene boards production. The computer simulation of cutting process of boards have been carried out.

INTRODUCTION

The emergence of flexible manufacturing systems (FMS) has sparked an increased interest and appreciation for real-time planning, scheduling, and control. A flexible manufacturing system (FMS) is a highly automated group technology machine cell consisting of a group of processing stations interconnected by an automated material handling and storage system and controlled by an integrated computer system. An FMS is defined as a manufacturing system consisting of automatically reprogrammable machines (material processors), automated tool deliveries and changes, automated material handling and transport, and coordinated shop floor control. An FMS is capable of processing a variety of different part styles simultaneously under numerical program control at the different workstations.

The FMS relies on the principles of group technology. No manufacturing system can be completely flexible. It cannot produce an infinite range of products. There are limits to the degree of flexibility that can be incorporated in a FMS. Accordingly, a flexible manufacturing system is designed to produce parts (or products) within a range of styles, sizes, and processes. In other words, An FMS is capable of producing a single part family or a limited range of part families. The concept for a FMS originated in the 1960s [1].

The flexible manufacturing system was first conceptualized for machining, and it required the prior development of numerical control. The credit for the concept is given to David Williamson, a British engineer employed by Mount during the mid 1960s. The concept was called System 24 because it was believed that the group of machine tools comprising the system could operate 24 hr/day, 16 hr of which would be unattended by human workers. The original concept included computer control of the numerical control (NC) machines, a variety of parts being produced, and tool magazines capable of holding various tools for different machining operations.

One of the first, FMS installed in the United States was a machining system at Ingersoll-Rand Company (now Ingersoll-Dresser) in Roanoke, Virginia, in 1967 by Sundstrand. By around 1985, the number of FMSs throughout the world had increased to about 300. About 20% to 25% of these were located in the United States. As the importance of flexibility in manufacturing grows, the number of FMSs is expected to increase. In recent times, there has been an emphasis on smaller, less expensive flexible manufacturing cells.

Real-time control of an FMS is not a trivial task. . Real-time activities primarily refer to daily operations that require efficient, timely, and adaptive responses to short-term planning, scheduling, and execution problems. Pertinent areas of interest include job releases, loading sequences, dead-locks, and response to resource disruptions such as machine or tool failure. Flexible routings, processing, and part mix, as well as the dynamic nature of a shop floor, place tremendous demands upon the control system. A detailed understanding of operational information, beyond mere "data crunching," is required for efficient production. Static schedules and strategies, developed ahead of time, quickly lose validity in a rapidly changing system, and thus cannot be directly applied over long planning horizons. To make matters even more difficult, no two flexible manufacturing systems are identical, and system decision making can vary in both type and complexity. Due to the dynamic nature of FMS and inherent differences between systems, researchers argue that generic, optimal seeking solutions may be too difficult to resolve in real-time.

Recently, there has been increased interest in the use of simulation for real-time planning, scheduling, and control [19]. Traditionally, simulation has been applied to long-term planning and design of manufacturing systems. These models have been termed "throw away models" because they are seldom used after the initial plans or designs are finalized. Recent reported applications of simulation for real-time, operational control include emulation of real-time control systems, adaptive scheduling and planning, real-time displays of system status, performance forecasting, as well as actual implementation into a shop floor controller. The use of simulation has appeared favorable to purely analytical methods, which often fail to capture complex interactions of a particular FMS.

To make process of machining both automatic and flexible the Computer Numerically Controlled (CNC) machine tools are used. CNC machine tools could be programmed, locally, with a method of making a component. One merely had to load a casting onto a fixture, supply an appropriate program and tooling, and the product would be predictably produced time after time. The coordination of the flow of jobs *between* machines also was carried out automatically. A rudimentary system which comprised a group of CNC machine tools connected by an automatic material handling system. A centralized computer-control system oversaw the shop and coordinated and scheduled the flow of jobs between the machines. With the further advances in computer technology, and the stabilization of CNC technology, the early 1980s saw a flurry of installations of systems designed along the lines of Williamson's System [3]. Pioneering companies such as Caterpillar and John Deere in the US started to build large systems which went against traditional manufacturing dogma - systems which combined economies of scope and scale. These large computer-controlled systems had a relatively high aggregate output yet were flexible, since they could produce a number of different products.

Flexible Manufacturing Systems (FMS), became a great focus of attention in industry and in academic research for a number of years[1]. Although the more skeptical might say that behind the rapid growth of publicity and interest in FMS lay a bubble inflated by a sales-hungry machine-tool industry, it was nevertheless clear that the systems demonstrated a significant technical advance in manufacturing practice. The real strength of these FMS lay in the fact that they brought tremendous benefits in inventory reduction (often 85%), quality improvement and lead time. In many installations, the inventory reduction alone was sufficient to justify the investment in hardware, software and system design effort.

Nowdays the development of FMS for some productions including different automated intellectual workstations is in primary concern. This problem acquire great importance for some complex plants. **[For this reason the aim of the thesis is the analysis FMS and development of intelligent workstation such as intelligent robots for FMS].** To solve this problem the followings have been carried out.

In chapter 1 the evolution of computer controlled manufacturing systems are given. The role and functions of computers in FMS has been classified.

In chapter 2 the types of FMS are given. Main integrating components of FMS, their functions are described. Software and hardware structure of some FMS, their control functions are clarified.

In chapter 3 the scheduling and control problems of FMS are described. The four scheduling algorithm for FMS are given.

In chapter 4 the development of intellectual robot for FMS are carried out. The design of the intellectual robot is based on fuzzy technology.

In chapter 5 the description of the expanded polystyrene boards production and the development of computer controlled manufacturing system for this production are given. The computer simulation of cutting process of boards have been carried out.

Conclusion presents the obtained important results and contributions in the thesis.

CHAPTER 1

EVOLUTION OF COMPUTER CONTROLLED MANUFACTURING SYSTEMS

1.1.Overview

Nowadays computers widely used for plant monitoring real time control of technological processes. This allows to automate industrial process and increase the efficiency of the productions.

In chapter 1 the hierarchy of computer control, evolution of computer in industry, and the functions of computer for controlling real time processes are given. Also supervisory computer control of flexible manufacturing system and its functions, types of software used in FMS are described.

1.2. The Hierarchy of Computer Control

One of the early applications of a computer in an industrial facility was for plant monitoring and supervisory control . Figure 1.1 shows an evolution of the use of digital computers in industrial processes.

As illustrated in figure, the next step in evolution after the supervisory control is the computer numerical control (CNC) of the machines followed by direct digital control (DDC) for a group of multiple numerical control (NC) machines.

In the DDC a computer reads and directly processes measurements, calculates the proper control outputs, and sends the control commands to the activation devices. In the initial implementations of DDC, backup analog control systems were used to avoid the ill effects of computer failures. In spite of the early computer hardware reliability problems, DDC demonstrated many advantages over analog control systems. They included the use of complex logic to calculate more accurately the control command values, ease of data logging, data trending, alarming, and so on. It also avoided the common problem of set-point drifts associated with analog devices. Several different system architectures evolved for the DDC systems in the late 1970s.

However, a central computer was a dominant feature of all the variations of these architectures. The single largest disadvantage of these architectures, as pointed out before, was the single-point failure of the central computer, which could shut down the process. It necessitated the expense of a second computer as backup. Another disadvantage was that the software for the central computer was very complex and required a team of software/hardware experts to change and maintain the software. It also had limited expansion capability and, when the expansions were made, they were very expensive.

In the mid 1960s the distributed control system architecture was brought forward as a viable option. The technology to implement the DDC in a cost-effective manner was not available until the early 1970s. The price of computers decreased significantly and personal computers could be used economically on the factory floor. A number of production lines with distributed control systems have begun emerge since the early 1970s.

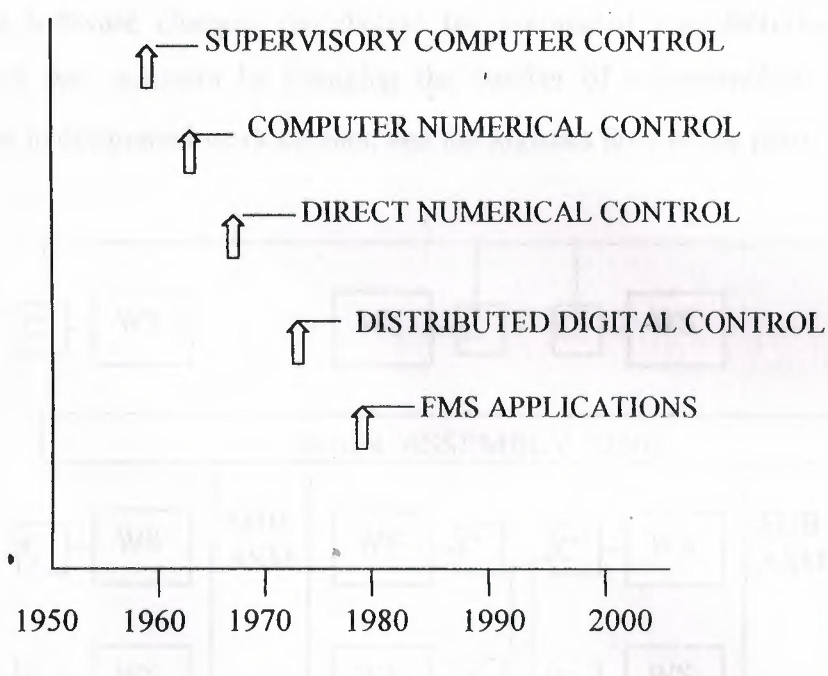


Figure 1.1. Evolution of the use of digital computers

As shown in figure 1.1, every work center has a dedicated control computer. The work center computer is responsible for making production happen in its work center. It communicates with each tool and robot and downloads the programs, resulting in appropriate commands for all equipment. It is also responsible for ascertaining the health of all the components in the work center.

The next level in the hierarchy of computer control is the subassembly/main assembly line computer. In general, this computer coordinates and controls the manufacturing activities within a section of the manufacturing facility. This computer communicates to each work-center computer the type of product to be made and all the appropriate commands. It also serves as a backup to any one of the failed work-center computers in its jurisdiction.

The supervisory computer is at the highest level in the hierarchy of computer control. This computer does overall production planning and scheduling and communicates with the subassembly/main assembly line computers [2]. In the event of the failure of any of the subassembly/main assembly computers, the supervisory computer takes over the tasks of the failed computer.

The key for producing economically different products or part numbers from the automated line shown in Figure 1.2 is the flexibility provided by the computer control. Simple software changes can dictate the automated manufacturing line to produce different part numbers by changing the number of subassemblies, the manufacturing process in designated work centers, and the logistics flow of the parts.

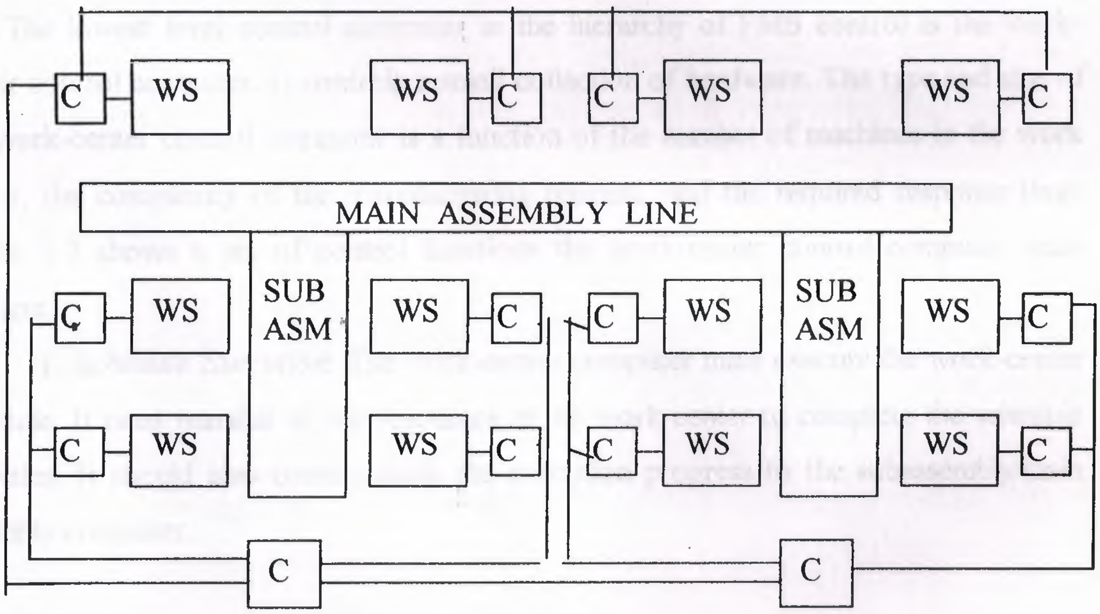


Figure : 1.2 Computer control in manufacturing systems..

1.3. Computer Control in a Work Center

The task of controlling total production in an FMS plant is the management of a complex set of machines interconnected with the automated parts transfer mechanisms. To control a real-time process, the computer must do the following [3]:

Process Control Commands: The control computer must have the software capability to direct the hardware devices to do their tasks. The hardware includes the actual machines that perform the manufacturing operations and the logistics mechanisms.

Process-initiated Interrupts: The control computer must receive and respond to the signals received from the process. Depending upon the importance of the signal, the computer may have to abort its current operation and perform a priority task.

Periodic Time-Initiated Events: The control computer must periodically collect management and status information. This is important for creating a history base and performing trend analyses.

System and Program-Initiated Events:. The work-center computer is connected to subassembly/main assembly control computer. Therefore, it must handle communication and data transfer with the higher level computers.

The lowest level control computer in the hierarchy of FMS control is the work-center control computer. It controls a small collection of hardware. The type and size of the work-center control computer is a function of the number of machines in the work center, the complexity of the manufacturing process, and the required response time. Figure 1.3 shows a set of control functions the work-center control computer must perform.

1. *Schedule Execution.* The work-center computer must execute the work-center schedule. It must marshal all the resources in the work center to complete the schedule activities. It should also communicate the execution progress to the subassembly/main assembly computer.

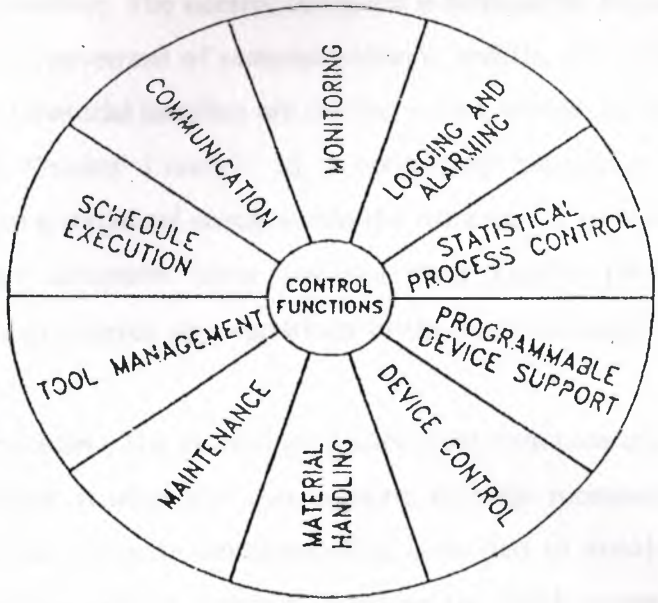


Figure 1.3. Work-center computer control functions

2. *Process Device Control:* The control computer can exercise true device control functions in the open-loop or closed-loop mode. The preplanned control is of the open-loop mode in which a predetermined standard set of commands or sequences of commands with intermediate logical checks are executed. In a closed-loop model, the control computer attains a desired value of a control variable by comparing its actual value with the desired value and uses the error signal as a control input. All devices which do not have their own control computer are controlled by the work-center control computer in this manner.

3. *Programmable Device Support:* the control computer is responsible for downloading the part programs to the devices in the work center that have their own control computers. The downloading is coordinate to the work-center production schedule.

4. *Tool Management:* This control function includes management of all reusable resources such as drills, bits, gauges, and so on, with the work center. It should keep the inventory of all tools and monitor tool wear.

5. *Maintenance:* The work-center control computer must keep track of the health of each piece of equipment within the work center. This includes keeping a maintenance history and creating preventive maintenance schedules. The maintenance function is directly related to the availability of the work center for production.

6. *Material Handling:* The control computer is responsible for the flow of material into the work center, movement of material within it, and the flow of material exiting it. The devices used for material handling are the conveyors, robots, AGVs, and so on.

7. *Statistical Process Control:* In a completely automated FMS plant, it is necessary to perform a statistical check within the work center to a predefined statistical norm. The control computer must perform such checks periodically and take appropriate actions to correct any variations in the manufacturing process within the work center.

8. *Communication:* The control computer must communicate with every device within the work center. It must also communicate with the subassembly/main assembly computer. Orderly and accurate communication is needed to avoid costly waste. The exact product schedule will be communicated to the work-center computer by the subassembly/main assembly computer.

9. *Monitoring:* The control computer serves as an alert and indefatigable supervisor. The most important monitoring task is that of process monitoring. This usually requires that the control computer establishes the status of the instruments and the process variables, the status of the equipment within the work center, and the status of the product itself. The control computer can also monitor indirect measurements. These measurements are a function of several directly monitored process attributes. The computer supervision is an important function in producing a defect-free product in a work center.

10. *Data Logging and Alarming:* The work-center control computer must collect and store the information about all the devices in the work center and the significant events that take place in it. These data will be used to create preventive maintenance schedules and keep the processing capabilities of the devices current. The data can be uploaded to the subassembly/main assembly computer for trend analysis. The data can be used for the alarm management function.

1.4. Computer Control in Subassembly/Main Assembly Lines

The control computer in a manufacturing line communicates with the work-center control computers and ensures scheduled production through its line. The subassembly/main assembly line control computer receives its production schedule

6. *Material Handling:* The control computer is responsible for the flow of material into the work center, movement of material within it, and the flow of material exiting it. The devices used for material handling are the conveyors, robots, AGVs, and so on.

7. *Statistical Process Control:* In a completely automated FMS plant, it is necessary to perform a statistical check within the work center to a predefined statistical norm. The control computer must perform such checks periodically and take appropriate actions to correct any variations in the manufacturing process within the work center.

8. *Communication:* The control computer must communicate with every device within the work center. It must also communicate with the subassembly/main assembly computer. Orderly and accurate communication is needed to avoid costly waste. The exact product schedule will be communicated to the work-center computer by the subassembly/main assembly computer.

9. *Monitoring:* The control computer serves as an alert and indefatigable supervisor. The most important monitoring task is that of process monitoring. This usually requires that the control computer establishes the status of the instruments and the process variables, the status of the equipment within the work center, and the status of the product itself. The control computer can also monitor indirect measurements. These measurements are a function of several directly monitored process attributes. The computer supervision is an important function in producing a defect-free product in a work center.

10. *Data Logging and Alarming:* The work-center control computer must collect and store the information about all the devices in the work center and the significant events that take place in it. These data will be used to create preventive maintenance schedules and keep the processing capabilities of the devices current. The data can be uploaded to the subassembly/main assembly computer for trend analysis. The data can be used for the alarm management function.

1.4. Computer Control in Subassembly/Main Assembly Lines

The control computer in a manufacturing line communicates with the work-center control computers and ensures scheduled production through its line. The subassembly/main assembly line control computer receives its production schedule from the FMS supervisory computer. The primary functions of the subassembly/main assembly line control computers are as follows.

1. *Monitor Production Performance:* The subassembly/main assembly line control computer must monitor the production performance of each work-center control computer in its jurisdiction. If the work-center control computer fails, the subassembly/main assembly line control computer has the responsibility to function as a backup control computer. Therefore, it must be capable of performing the control functions of the work-center control computer.

2. *Database Management:* The subassembly/main assembly line control computer receives and stores all the process and device- related data in its database. It manipulates these data and uploads them to the EMS supervisory control computer for generating production plans.

3. *Production Scheduling:* The subassembly/main assembly line control computer receives primary production and two backup production schedules each day from the FMS supervisory control computer. The subassembly/main assembly control computer utilizes the backup production schedule in the anomalous operating conditions. It communicates such operating conditions to work-center computers as well as to the FMS supervisory control computer.

4. *Alarm Management Section:* The subassembly/main assembly line control computer is responsible for the alarm management system. The control computer must keep log of the reasons for all the alarms in all the work centers in its jurisdiction.

1.5. Flexible Manufacturing System Supervisory Computer Control

As the name implies, the supervisory computer supervises and controls production throughout the entire manufacturing facility. This is the highest and most important level in the hierarchy of computer control. At this level, the demands for various parts are analyzed. Knowing the process involved in the manufacture of each product and the capacity of each of the equipment, a production schedule is created. Simulations are run to understand the capacity pinch-point operations. In [4] a hierarchical production scheduling policy that minimizes the disruptive effects of such disturbances as machine failures is described. A cost analysis of the schedule is also performed to quantify the unused capacity within each subassembly area. Since an FMS facility is very costly, it is important to utilize the equipment as much as possible [5].

Once the schedule is finalized, material requirements are created. Also, detailed work plans for each subassembly/main assembly work station are developed for each day. In general, a rolling one-week firm schedule with the production outlook for the next four weeks appears to be an adequate scheduling scenario for a complex product. That is, at the end of each day a new day is added to the firm schedule and to the outlook. The detailed work plan for each day is downloaded to each subassembly/main assembly computer at the beginning of each day.

1. *Production Planning:* Production planning necessarily begins with the analysis of Firm demands for each product and the forecast of future sales. The supervisory computer database necessarily contains the description of the production process for each product and the tool capacity in each subassembly/main assembly line. A preliminary production plan for the facility is created for the rolling day that will accommodate the firm and projected demands.

2. *Simulation:* The demands for each product within the production plan should be simulated to understand the pinch-point resources and the intricate subtleties for the product changeover requirements. "What if" scenarios should also be simulated to create alternative production plans. They should include tool failures, part shortages, or quality problems. The data used should be based upon the historical data stored in the supervisory computer. The simulation effort may be very complex since a product may be able to take alternative routes made possible by the FMS. If the preliminary production plan does not appear feasible, the production planning step described above may have to be repeated.

3. *Master Production Schedule:* A detailed production schedule for each day within the rolling week is created for the preliminary production plan, as well as two backup production plans using the "what if" analysis. The details of the master production schedule include the sampling/verification plan for each subassembly and for the final assembly for quality. An overview of the EMS software control functions for variable missions and scheduling procedures is given in [7].The master production schedule must be accurate. It will be downloaded each day to the subassembly/main assembly computers and it dictates the total production operations within the FMS.

4. *Capacity Analysis:* The supervisory computer must determine the unused capacity in each subassembly area. This unused capacity may be utilized to produce

spare parts such as field replaceable units (FRUs). Also, the supervisory computer must schedule preventive maintenance consistent with the utilization of the equipment.

5. Communication: A supervisory computer requires constant communication with the subassembly/main assembly computers to be aware of the status of the equipment as well as the production. It can then compare the status with the simulation results to predict the problem and sound appropriate alarms. In a completely automated FMS, the role of the supervisory computer is very important. It can help to fully utilize FMS hardware and therefore help to manufacture cost-competitive products.

1.6 Types of Software

Software for FMS can be divided into three broad categories: design, extrinsic functions, and intrinsic functions. The design software is used to identify the major components of the system concept, to evaluate design sensitivities and trade-offs for a given concept, and to demonstrate that the system is capable of meeting the planned production requirements. Some of the extrinsic and intrinsic functions that support the FMS are displayed in Figure 1.4. Software for the extrinsic functions is used to plan and control the functions that take place outside the physical boundaries of the FMS. Software for the intrinsic functions is used to load and control the components within the physical boundaries of the FMS.

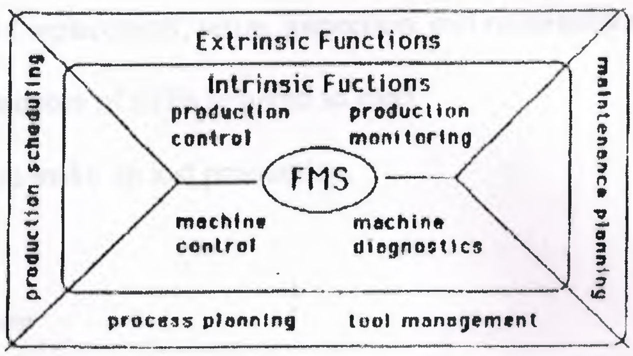


Figure 1.4 Extrinsic and intrinsic functions for the FMS.

1.7. Design functions

1. 7.1.Capacity Planning

Capacity planning is concerned with planning the needed manpower and machine resources to meet the master schedule during a specified time frame. The capacity of a system determines the amount of product that can be produced in a given time period. Capacity planning also is considered to be a refinement procedure wherein the long-range schedule is adjusted to the manufacturing needs of the nearer future. The planning horizon generally is one or several months and the time increments are weeks. The output of capacity planning can be an input to material requirements planning and other scheduling activities. Capacity planning software is used to evaluate the impact of specified production levels on the resources provided through the FMS. It can be used to establish limits for the quantities of the various parts planned for the FMS and the limits on the mix of these quantities for the available machines, tooling and fixturing, and part pallets. It also can be used to evaluate the impact of introducing new part numbers into the FMS and identifying the changes that may be needed to accommodate the additional load. Capacity planning software can be used to evaluate factors that influence the capacity requirements for the FMS—factors such as the following

- Minimum system requirements for
- Effect of change in part mix and demand levels
- Effect of change in planning hori
- Effect on schedule of down machines or tool shortage
- Labor support required for various schedules (e.g., part programming, load/unload, tool replacement, setup, inspection, and maintenance)
- Minimum number of shifts required to meet
- Capability to make up lost production.

1.7.2. Simulation

Simulation has become an important tool when new manufacturing processes are conceived or existing ones are to be altered. A simulation can be described as a software tool that consists of methods, activities, and resources. A simulation is a model of a manufacturing process or system that can be used to emulate it. The model characterizes the layout configuration, the number of pallet shuttles in the system, and the production scheduling rules.

A simulation model for an FMS can be conceived as a four-level hierarchy: Basic components, machine control, scheduling, and order processing and planning [8]. The "basic components" are found at the lowest level of the hierarchy and consist of the machine tools, material handling equipment, inspection systems, and support equipment. The primary action at the basic components level is to implement instructions received from the next level.

At the machine control level, data are gathered and instructions are implemented on the basis of higher level goals and current events. Communication between machines and equipment at this level is limited.

- Data collection (including knowledge from people who know the system and potential problems)
- Simulation system architecture, performance measures, priority rules allowed, and utilization trade-off capability
- Representation of machine and material handling equipment characteristics, including carrier travel speed
- Representation of traffic rules (e.g., load parts on pallet fixture, launch loaded pallets into system, and resolve traffic conflicts)
- Quantitative skills of individuals writing/using simulation
- Reasonable design of simulation experiment(s)
- Reasonable presentation of data (considering performance measurement, correct interpretation of accuracy, and effective use of graphics)
- Documentation of the simulation (including user learning)

1.7.3. Knowledge-Based Systems

Both expert systems and knowledge-based systems are being used to analyze flexible manufacturing systems. An illustration of the architecture for a knowledge-based simulator is given by the Society of Manufacturing Engineers (SME). The expert system incorporates a broad analysis and justification of the decision, while knowledge-based systems employ human knowledge to solve problems that ordinarily require human intelligence [9]. Expert systems (ES) have been applied to both design and operation of FMS. Expert systems can use data and constraints from the problem environment and generate a solution (stand-alone ES) or work jointly with a model and an algorithm (tandem ES).

A blackboard model-based expert system also can be used in the design of an FMS. The main idea is to provide a kind of distributed intelligence for the various FMS design activities. The blackboard data structure serves as a communication interface among the various knowledge sources. To complete the expert system, a user interface and a control mechanism are needed.

Expert systems and knowledge-based systems provide a way to include knowledge sources in the decisions that are made in the FMS design process. Examples of these knowledge sources include characterizing part families, selecting equipment, layout of machines, building FMS models, analyzing output results from FMS simulation, and justifying capital investment.

•Effective user interface (e.g., data entry, initial FMS layout, access existing models and data, constraint analysis)

•Definition and modification of part family

•Appropriate representation of system components and constraints

•Establish machine layout alternatives

•Relationship to other software planning systems (e.g., model definition, simulator input/output, data analysis/interpretation)

•Nature of command structure or control methodology

•Trade-off between degree of accuracy and software complexity

1.8.Exstrinsic Operating functions

1.8.1. Production Scheduling

The master production schedule gives the timetable for end-product deliveries. The schedule is translated into material and component requirements by using material requirements planning (MRP) in conjunction with factory capacity planning. An operation schedule is developed to assign specific jobs to specific machines to meet the daily or even hourly requirements. The end products specified in the master schedule consist of components, each of which is manufactured by a sequence of processing operations. Operation scheduling involves the assignment of start dates and completion dates to the batches of individual components and the designation of machines on which the work is to be or at least can be performed. The scheduling problem is complicated by the fact that many parts may be competing for time on a fixed number of machines. The complications becomes more severe when interruptions and delays occur, for example, machine breakdowns, changes in job priority, and other operational problems.

Applications are also being developed using the expert system methodologies [9]. Other opportunities exist through integration of scheduling algorithms and simulation models in order to investigate trade-offs and do sensitivity analysis.

A variety of computer-based scheduling methods can be used in production. Selection of the best method will depend on the overall production objectives and the availability of resources. An economic analysis of the production run and economical batch sizes should be conducted. Given decisions about the number of parts needed and their due dates, the scheduling software determines the appropriate sequence to process a part on the available machine(s) in accordance with a predetermined set of priority rules. The allocation of parts to machines is referred to as machine loading and will depend on the operations to be performed, the capability of each available machine, and the priority scheduling rules.

Selection of the operation scheduling software will depend on the types of parts produced, the type of FMS, and the scheduling priority system. Specific capabilities of the software to be considered include

- Priority rules allowed (e.g., customer order, job/operation precedence, work-center loading, job dispatch, and time conversion)
- Capability to deal with contingencies (e.g., machine breakdown, tool breakage, labor changes)
- Capability to optimize schedule variables to meet a production objective (e.g., minimum production cost)

1.8.2. Process Planning

Process planning represents the link between engineering design and shop floor manufacturing. It determines the manufacturing operations required to transform a part from a rough state to the finish state specified on the engineering drawing. More specifically, it is "that function within a manufacturing facility that establishes the processes and process parameters to be used in order to convert a piece-part from its original form to a final form that is predetermined on a detailed engineering drawing [10]. For machining, process planning includes determination of the specific sequence of machining operations and the selection of depth of cut, feed rate, and cutting speed for each cut of each operation. The machining parameters are in turn included within the CNC part program for the machine(s) that will cut the part. In some cases, the selection of tooling, fixturing, and inspection equipment also is included in the process planning for a part. Many versions of computer-aided process planning systems are available to assist the process planner, though none have been specifically included in software for FMS.

Process planning software is based on a "table lookup" (or database) structure, empirical calculations, or both. In some cases, specific data on machine tool capabilities and tooling also are included. Ultimately, the software generates a form of process plan or "route sheet." Integrated software that provides for the calculation of optimal machining parameters while the CNC part program is developed is not available. Either default values presented by the part programming software are accepted, or engineers enter parameters from tables or other calculation programs.

1.9. Intrinsic operating functions

1.9.1 Production control

Production (or shop floor) control is concerned with the release of the production parts in to the FMS, controlling the progress of the parts in process through the machine tools and inspection stations that are identified by the operating schedule, and acquiring current information on the status of each part in the system. When provided with the capability to determine optimal routing within the FMS, alternative assignments of jobs to machines can be determined based on the machine capabilities , part assignments, and utilization of the machines at any given time. When appropriate, the part program for the part assigned to a machine must be identified and downloaded to the machine. The status of the part carriers in the system must be monitored and used to determine part load and unload requirements. Each part entering the FMS must be identified and production data associated with it must be accumulated as the part moves through the FMS.

1.9.2. Production Monitoring / Reporting

Production monitoring and reporting is concerned with data collection and management reporting. Records of the number of completed parts (both accepted and rejected), inspection results, tool change data , machine utilization, and other management data are collected and reported by the production monitoring / reporting system.

Production monitoring / reporting software provides standard and custom reports for managing the FMS resources. Problems are reported immediately to minimize delays and improve utilization.

1.9.3 Machine / Process Control

At the lowest level of the communication hierarchy for an FMS is the control of the manufacturing processes, operations, and material handling equipment. Parts are moved into and out of each scheduled machine or workstation. The dedicated material handling equipment picks up and delivers parts on command from the FMS computer. Each machine tool in the system is capable of starting , completing , and monitoring the machining operations for each part routed to it. Appropriate sensors and control algorithms are included to correctly position parts at the workstations and to provide process control.

Machine /process control software provides both control and monitoring capability. Part programs and data are stored at each machine tool. Tool wear and replacement strategies are integrated with the machine control software. Software may contain capabilities to adjust the process in real time for variations in process variables.

1.10 Summary

This chapter presented the computer control in FMS, and in assembly lines. Also design functions, production control, machine process control and operating functions are described in this chapter.

CHAPTER TWO
FLEXIBLE MANUFACTURING SYSTEMS

2.1 Overview

A flexible manufacturing system (FMS) is an arrangement of machines interconnected by a transport system. The transporter carries work to the machines on pallets or other interface units so that work-machine registration is accurate, rapid and automatic. A central computer controls both machines and transport system. The key idea in FMS is that the co-ordination of the flow of work is carried out by a central control computer. This computer performs functions such as:

- Scheduling jobs onto the machine tools
- Downloading part-programs (giving detailed instructions on how to produce a part) to the machines.
- Sending instructions to the automated vehicle system for transportation

In this chapter the layout types of FMS, flexibility of manufacturing systems, hardware and software components of FMS, their characteristics are considered.

2.2 Flexibility and Automated Manufacturing Systems

'Flexible manufacturing systems vary in terms of number of machine tools and level of flexibility. When the system has only a few machines, the term *flexible manufacturing cell* (FMC) is sometimes used. Both cell and system are highly automated and computer controlled. The difference between a FMS and a FMC is not always clear, but is sometimes based on the number of machines (workstations) included. The flexible manufacturing system consists of four or more machines, while a flexible manufacturing cell consists of three or fewer machines [12]. However, this distinction is not universally accepted, and the terminology of this technology is not yet fully sorted out.

Some highly automated manufacturing systems and cells are not flexible, and this leads to confusion in terminology. For example, a transfer line is a highly automated manufacturing system, but it is limited to mass production of one part style, so it is not a flexible system. To develop the concept of flexibility in a manufacturing system, consider a cell consisting of two CNC machine tools that are loaded and unloaded by an industrial robot from a parts carousel, perhaps in the arrangement depicted in Figure 2.1. The cell operates unattended for extended periods of time. Periodically, a worker must unload completed parts from the carousel and replace them with new work-parts. This is truly an automated manufacturing cell, but is it a flexible manufacturing cell? One might argue yes. It is flexible since the cell consists of CNC machine tools that can be programmed to machine different part configurations like any other CNC machine. However, if the cell only operates in a batch mode, in which the same part style is produced in lots of several dozen (or several hundred) units, then this does not qualify as flexible manufacturing.

To qualify as being flexible, a manufacturing system should satisfy several criteria. The tests of flexibility in an automated production system are the capability to (I) process different part styles in a nonbatch mode, (2) accept changes in production schedule, (3) respond gracefully to equipment malfunctions and breakdowns in the system, and (4) accommodate the introduction of new part designs. These capabilities are made possible by the use of a central computer that controls and coordinates the components of the system. The most important criteria are (1) and (2); criteria (3) and (4) are softer and can be implemented at various levels of sophistication.

If the automated system does not meet these four tests, it should not be classified as a flexible manufacturing system or cell. Getting back to our illustration, the robotic work cell would satisfy the criteria if it (1) machined different part configurations in a mix rather than in batches; (2) permitted changes in production schedule and part mix; (3) continued operating even though one machine experienced a breakdown; for example, while repairs are being made on the broken machine, its work is temporarily reassigned to the other machine; and (4) as new part designs are developed, NC part programs are written off-line and then downloaded to the system for execution. This fourth capability also requires that the tooling in the CNC machines as well as the end effector's of the robot be suited to the new part design.

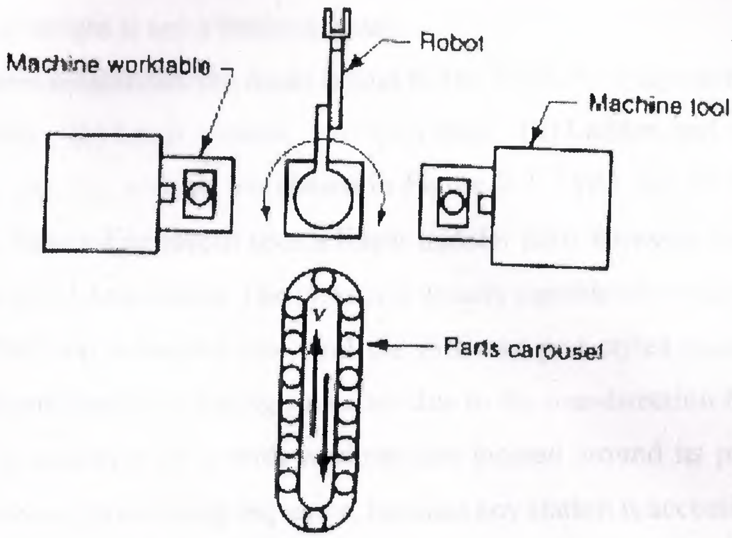


Figure 2.1. Automated manufacturing cell with two machine tools and a robot.

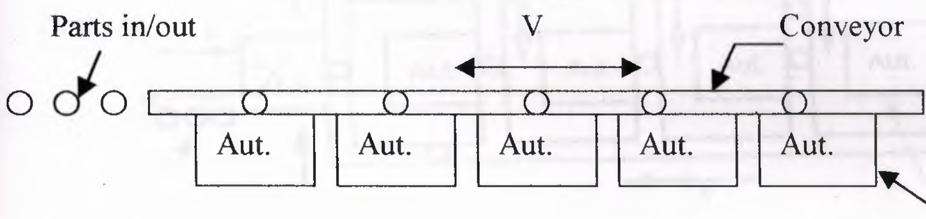
2.3. Integrating the FMS Components

A FMS consists of hardware and software that must be integrated into an efficient and reliable unit. It also includes human personnel

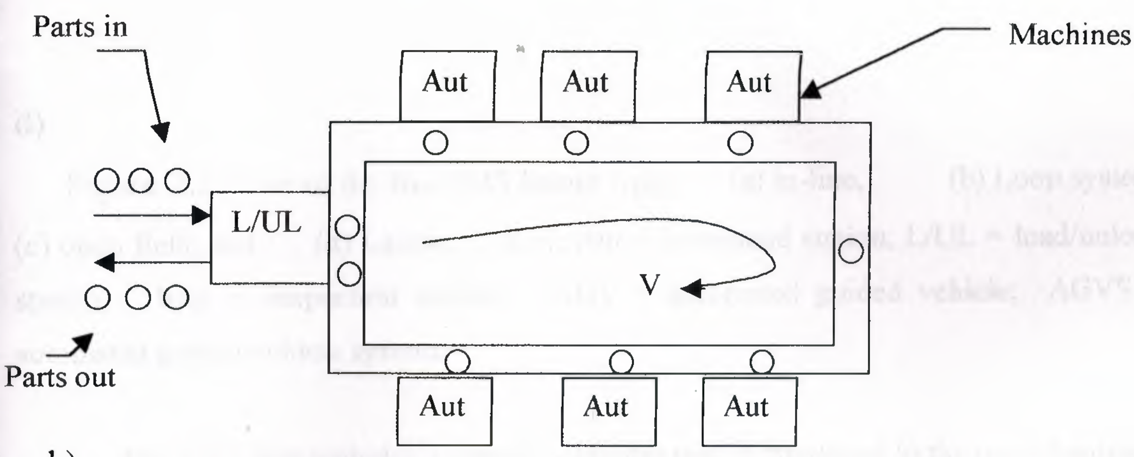
Hardware Components. FMS hardware includes workstations, material handling system, and central control computer. The workstations include CNC machines in a machining-type system, plus inspection stations and parts cleaning and other stations, as needed. For a flexible machining system, a central chip conveyor system is usually included below floor level.

The material handling system is the means by which parts are moved between stations. The material handling system usually includes a limited capability to store parts. Handling systems suitable for automated manufacturing include roller conveyors, in-floor towline carts, automated guided vehicles, and industrial robots. The most appropriate type depends on part size and geometry, as well as factors relating to economics and compatibility with other FMS components. Nonrotational parts are often moved in a FMS on pallet fixtures, so the pallets are designed for the particular handling system, and the fixtures are designed to accommodate the various part geometries in the family. Rotational parts are often handled by robots if weight is not a limiting factor.

The handling system establishes the basic layout of the FMS. Five layout types can be distinguished: (a) in-line, (b) Loop system, (c) Open field , (d) Ladder, and (e) robot centered cell. Types (a), (b), (c), and (d) are shown in Figure 2.2. Type (e) is shown in Figure 2.1, respectively. The in-line layout uses a linear transfer parts between processing stations and load/unload (L/UL) station(s). The system is usually capable of two-directional movement; if not, the FMS op a transfer line, and the different part styles made on the system must follow the same basic processing sequence due to the one-direction flow. The loop layout consists of a conveyor loop with workstations located around its periphery. This configuration permits any processing sequence, because any station is accessible from any other station. This is also true for the ladder layout, in which workstations are located on the rungs of open field layout is the most complex FMS configuration and consists of several loops tied together. Finally, the robot-centered cell consists of a robot whose work volume includes the load/unload positions of the machines in the cell.

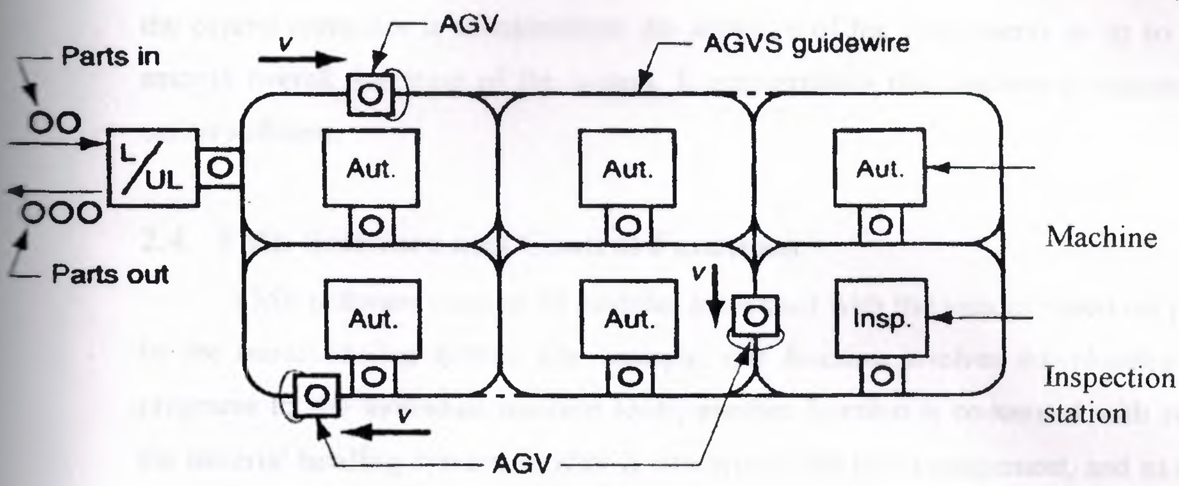


a)

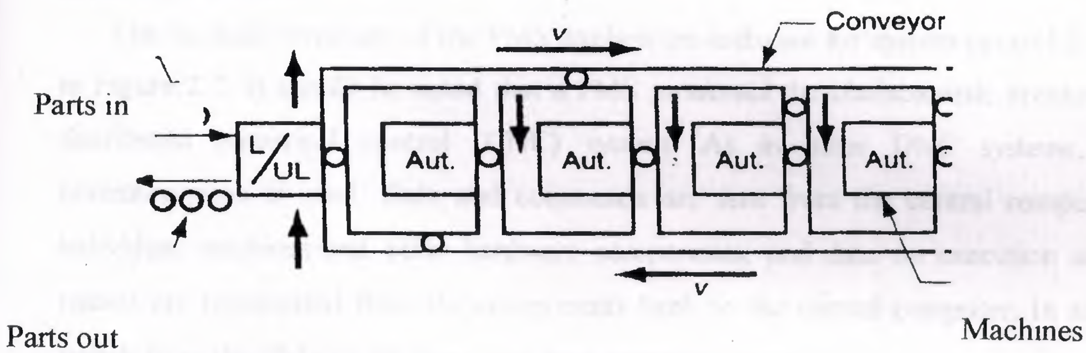


b)

b)



c)



d)

Figure 2.2. Four of the five FMS layout types: (a) in-line, (b) Loop system, (c) open field, and (d) Ladder Key: Aut.= automated station; L/UL = load/unload station; Insp. = inspection station; AGV = automated guided vehicle; AGVS = automated guided vehicle system.

The FMS also includes a central computer that is interfaced to the other hardware components. In addition to the central computer, the individual machines and other components generally have microcomputers as their individual control units. The function of the central computer is to coordinate the activities of the components so as to achieve a smooth overall operation of the system. It accomplishes this function by means of application software.

2.4. FMS Software and Control Functions

FMS software consists of modules associated with the various functions performed by the manufacturing system. For example, one function involves downloading NC part-programs to the individual machine tools, another function is concerned with controlling the material handling system, another is concerned with tool management, and so on. Table 2.1 presents a listing of the functions included in the operation of a typical FMS. Associated with each function are one or more software modules. Terms other than those in our table may be used in a given installation. The functions and modules are largely application specific.

The modular structure of the FMS application software for system control is illustrated in Figure 2.3. It should be noted that a FMS possesses the characteristic architecture of a distributed numerical control (DNC) system. As in other DNC systems, two-way communication is used. Data and commands are sent from the central computer to the individual machines and other hardware components, and data on execution and performance are transmitted from the components back to the central computer. In addition, an uplink from the EMS to the corporate host computer is provided.

An additional component in the operation of a flexible manufacturing system is human labor. Duties performed by human workers include (1) loading and unloading parts from the system, (2) changing and setting cutting tools, (3) maintenance and repair of equipment, (4) NC part programming, *(5)* programming and operating the computer system, and (6) overall management of the system.

TABLE 2.1. Typical Computer Functions Implemented by Application Software Modules in a Flexible Manufacturing System.

Function:	Description
NC part programming	Development of NC programs for new parts introduced into the system. This includes a language package such as APT.
Production control	Product mix, machine scheduling, and other planning functions.
NC program download	Part program commands must he downloaded to individual stations using DNC.
Machine control	Individual workstations require controls, usually CNC.
Work-part control	Monitor status of each work-part in the system, status of pallet fixtures, orders on loading/unloading pallet fixtures.
Tool management	Functions include tool inventory control, tool status relative to expected tool life, tool changing and resharpening, and transport to and from tool grinding.
Transport control	Scheduling and control of handling system.
System management	Compiles management reports on performance (utilization, piece counts, production rates, and the like); FMS simulation sometimes included.

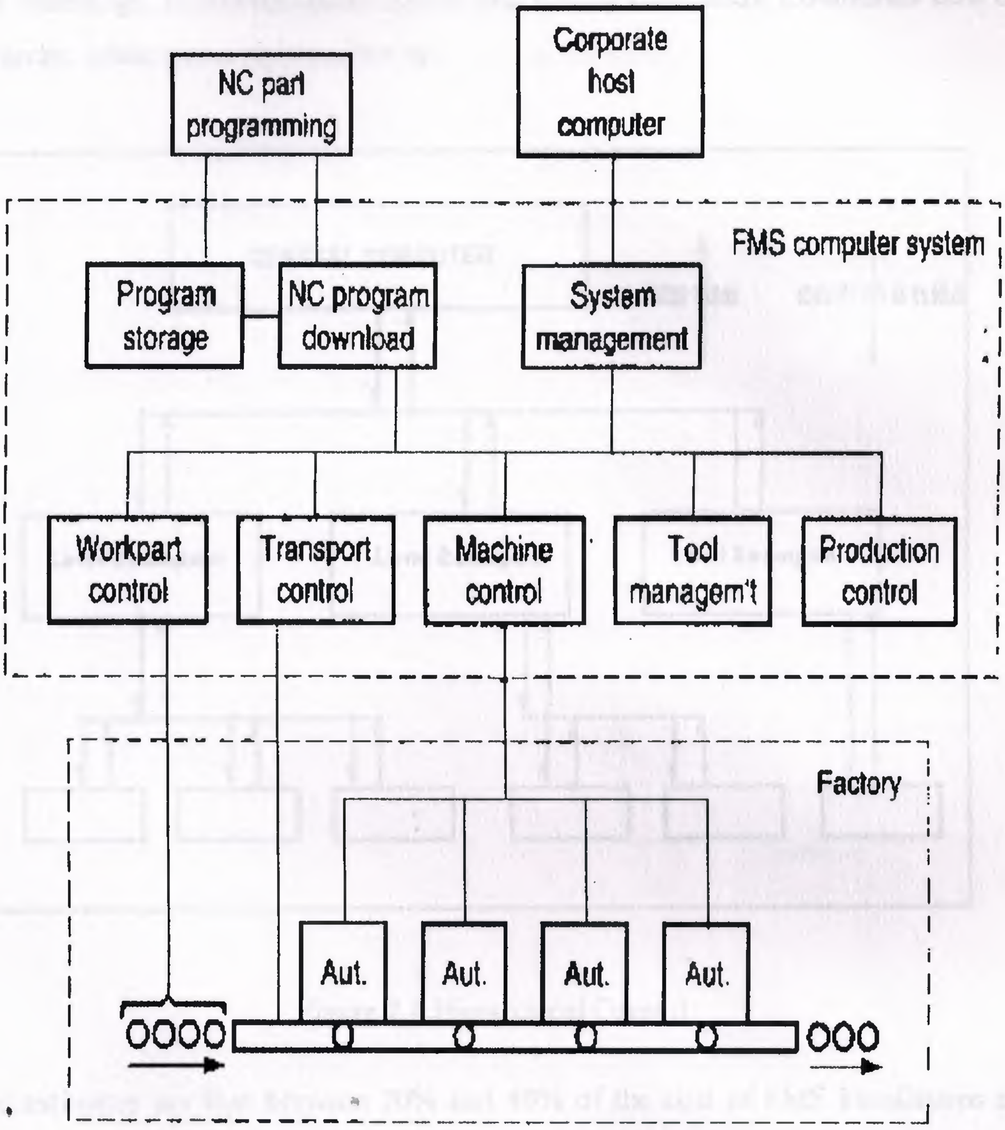


Figure 2.3 Structure of FMS application software system.
Key: Aut.= automated station; NC = numerical control

Writing FMS control software is not a trivial matter. The software is often custom-written, and is not straightforward programming task. There are complex, real-time interactions with remote hardware which require great expertise and experience on the part of the programmer, particularly for larger systems. In order to simplify this problem, many systems use a hierarchical approach to real-time control [12].Each computer controls a team of underlings, collecting status reports and issuing commands. Commands flow down the hierarchy, while status reports flow up.

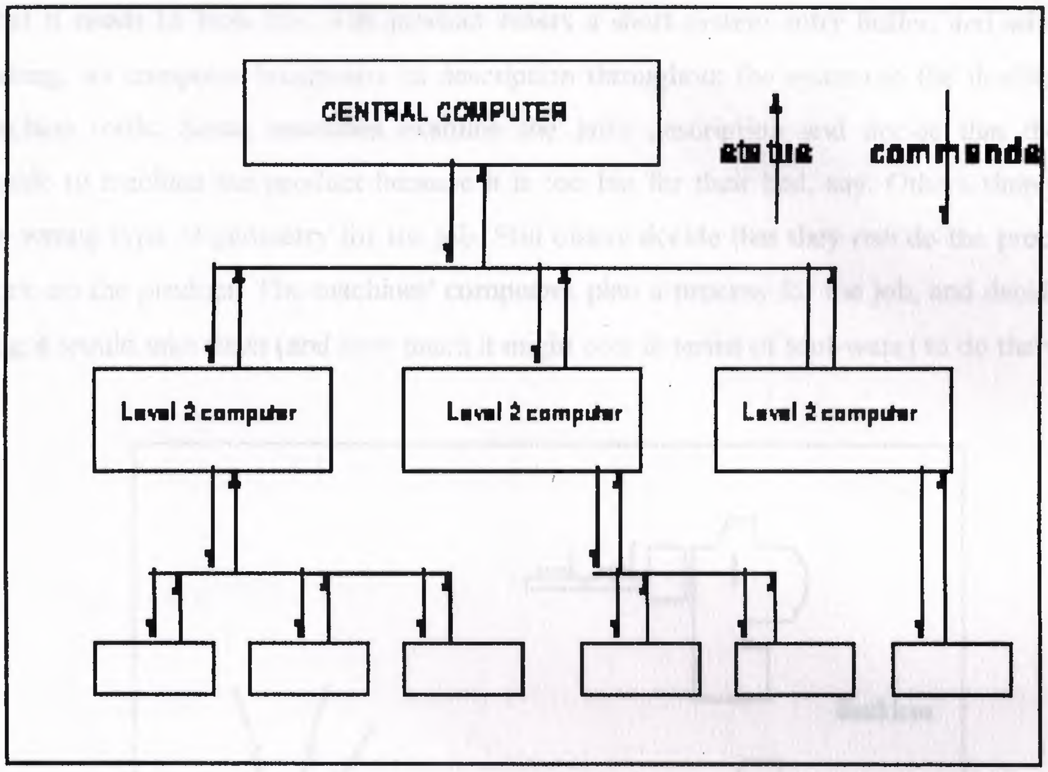


Figure 2.4 Hierarchical Control

Various, estimates are that between 20% and 40% of the cost of FMS installations are in computer software and hardware development.

2.5 The Computerized Product in a Society of Machines

Consider a machine shop with fifty CNC machine tools and automatic vehicles for transportation of materials. This section outlines a new type of control structure for such a system. It is best outlined by describing an example of the production procedure which is followed in order to produce an individual item. A production-control computer requests that a casting which has arrived at the shop be machined. The casting is manually bolted in a flexible fixture on a pallet, and a small computer is fitted to the pallet. This computer contains a processor, some memory and a radio. This assembly will be referred to as "the part". The production control system loads the memory of the part's computer with the processing requirements of the product. In other words, the control system tells the part what it needs to look like. The product enters a short system entry buffer, and while it is waiting, its computer broadcasts its description throughout the system to the flexible CNC machine tools. Some machines examine the job's description and decide that they are unable to machine the product because it is too big for their bed, say. Others simply have the wrong type of geometry for the job. Still others decide that they *can* do the processing work on the product. The machines' computers plan a process for the job, and decide how long it would take them (and how much it might cost in terms of tool-wear) to do the work.

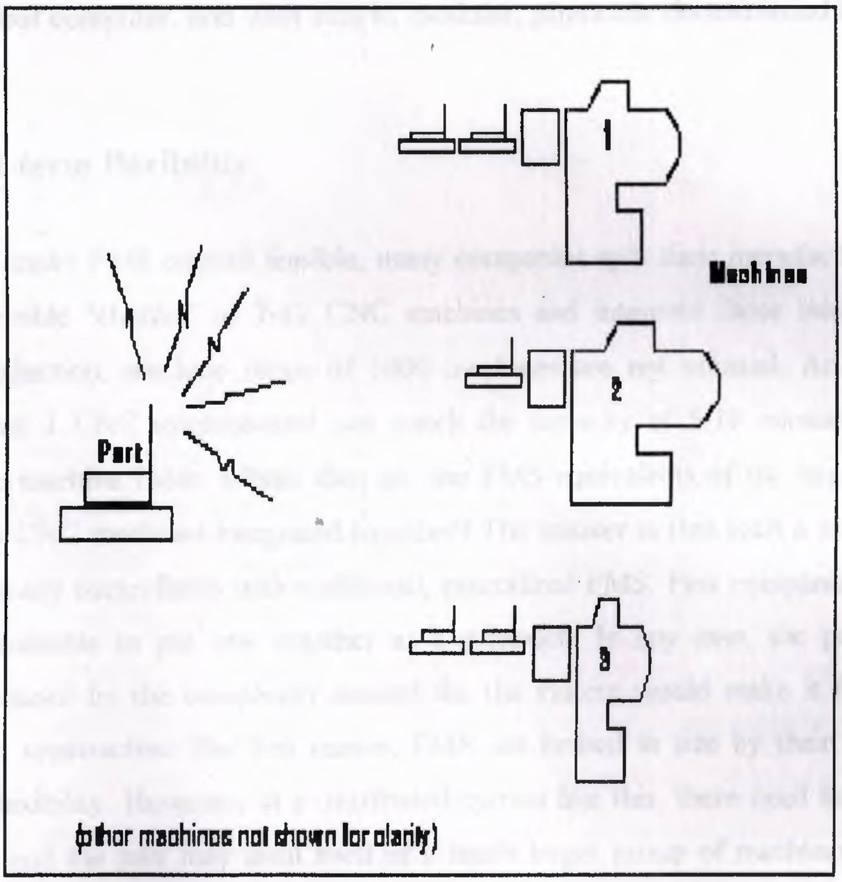


Figure 2.5 Part transmits processing request and data to the system.

Having determined how much processing time is needed, the machine checks its local buffer, determines how many jobs it has waiting for it, and forms a "bid" on the job. It transmits this bid across the network to the waiting part. The part waits until a system-set deadline to receive bids and having collected them, selects a *winner* from the bidding machines. It sends a message to that machine that it has selected it and expects to arrive for processing. The next arrangement the part needs to make is for transportation to the machine. Here there are a number of possibilities but let us say, for now, that the part essentially "calls a cab": It requests transportation from an automated vehicle dispatcher which sends a vehicle to it, and it arrives at the machine. After waiting in line it is machined. While it is waiting it arranges subsequent machining and ultimately leaves the system, once all tasks are complete. Having been processed the part moves out of the system to be assembled into its final product. The part has thus been processed without a central control computer, and with simple, modular, physically decentralized hardware and software.

2.6. Short-term flexibility

In order to make FMS control feasible, many companies split their manufacturing systems into manageable "chunks" of 7-12 CNC machines and integrate these into an FMS. In manual production, machine shops of 1000 machines are not unusual. An often quoted figure is that 1 CNC machine-tool can match the capacity of 5-10 manually controlled, stand alone machine tools. Where then are the FMS equivalents of the large shops, with 100 or 200 CNC machines integrated together? The answer is that such a shop would not be economically controllable with traditional, centralized FMS. Few companies would have the skills available to put one together as a monolith. In any case, the poor long-term flexibility caused by the complexity needed for the system would make it financially and strategically unattractive. For this reason, FMS are limited in size by their controllability and poor flexibility. However, in a distributed system like this, there need be no such size constraints and the part may avail itself of a much larger group of machines, not just the one which it has access to in its traditionally controllable chunk. The fact that the part has access to more machines, and that no group of components is specified in advance for those machines to make, means that many different parts can be serviced appropriately by the system.

The relative advantage here depends to a large extent on the degree of multiple redundancy in the system as a whole. Such redundancy is becoming more prevalent as a result of technological changes in machine tools. Today's CNC machines are considerably more versatile than they were 15 years ago . The move from 3-axis to 5 and 6 axis machines, the use of modular tooling systems and pallet-changers along with the improvement of tool-management systems have dramatically increased the scope of jobs which an individual machine can undertake. This trend is likely to continue as machine tool manufacturers accommodate more and more operations into one workpiece setting and one machine tool. An allied trend is the increase in information processing ability at the machine level. The continuing technological advances in physical versatility and in the distribution of computing in machine tools imply increased multiple redundancy in systems, as well as local, rather than central, information processing.

2.7 Long-term Flexibility

Consider the software changes necessary in order to add a machine to this system. An important question is how long the manufacturing system has to be down in order to implement the software changes. In this system, there is no need to take the system down. A new machine may simply be told the "rules of the game", and be installed in the plant with power and access to tooling. It becomes part of the system with no lost production. Removing a machine from the system is also straightforward. The machine simply stops bidding, and jobs stop coming to it.

Each addition and removal has very limited system-wide ramifications. The machines rise and fall in utilization as they become more or less appropriate for the current product range. This greatly facilitates new product and process introductions, since any peculiar requirements for processing of a new product may be introduced without systemic disruption.

2.8. Programmable Automation

The automated transfer lines were developed well before the computer age. They were actuated using mechanical cams, electrical relays, and similar components. Once built, it was very difficult to make changes in the equipment for new parts or products. The term *fixed automation* applies to these kinds of systems, in which the processing steps and their sequence are fixed by the equipment configuration. The traditional features of this type of automation include high initial investment, high production rates, low unit cost if the product is made in sufficiently high quantities, and inflexibility in accommodating product changes.

An alternative form of automation is *programmable automation,* in which the equipment is designed with the capability to change the processing steps and/or their sequence so that different product styles can be produced. In programmable automation, the process is controlled by a program, which is a set of instructions coded so that the equipment can read and interpret them. Changes in the process are made by changing the program. The features of this form of automation include lower production rates than for fixed automation, low or medium production quantities, and flexibility to accommodate changes in product configuration.

Although the traditional application of programmable automation is for low and medium production, it is of interest to note that modem controllers used on equipment for mass production are almost invariably computer based. Thus transfer lines and similar production systems traditionally classified as fixed automation are today implemented by controllers that can be programmed. in many cases, these systems are designed for mass production of a single product or part, and the production hardware cannot be easily altered for different product styles. Nevertheless, it is advantageous to control the hard- ware by means of a programmable computer controller for the following reasons: (1) it facilitates installation, (2) improvements and upgrades can be made in the control functions, (3) the process and equipment can be readily monitored, (4) data on process performance and product quality can be compiled, and (5) a convenient human-machine interface can be provided.

Three types of programmable automation widely used in manufacturing: numerical control, industrial robotics, and programmable logic controllers. The common functional attribute shared by these technologies is their ability to be programmed and reprogrammed.

2.9 Numerical Control

One of components of FMS is numerical control. Numerical control (NC) is a form of programmable automation in which the mechanical actions of a piece of equipment are controlled by a program containing coded alphanumeric data. The data represent relative positions between a workhead and a workpart. The workhead is a tool or other processing element, and the workpart is the object being processed. The operating principle of NC is to control the motion of the workhead relative to the workpart and to control the sequence in which the motions are carried out. The first application of numerical control was in machining , and this is still an important application area.

2.9.1. Components of a NC System

A numerical control system consists of three basic, components: (1) part program, (2) machine control unit, and (3) processing equipment. The *part program* (the term commonly used in machine tool technology) is the detailed set of commands to be followed by the processing equipment. Each command specifies a position or motion that is to be accomplished by the workhead relative to the processed object. A position is defined by its *x*- y- *z* coordinates. In machine tool applications, additional details in the NC program include spindle rotation speed, spindle direction, feed rate, tool change instructions, and other commands related to the operation. For many years, NC part programs were encoded on 1-in.-wide punched paper tape, using a standard format that could be interpreted by the machine control unit. Today, punched tape has largely been replaced by newer storage technologies in modem machine shops. These technologies include magnetic tape and electronic transfer of NC part programs from central computer.

The *machine control unit* (MCU) in modem NC technology is a microcomputer that stores the program and executes it by :converting each command into actions by the processing equipment, one command at a time. The MCU consists of both hardware and soft ware. The hardware includes the microcomputer, components to interface with the processing equipment, and certain feedback control elements. The MCU may also include a tape reader if the programs are loaded into computer memory from punched tape. The software in the MCU includes control system software, calculation algorithms, and translation soltware to convert the NC part program İnto a usable format for the MCU .It may also İnclude interpolation algorithms to achieve smooth motions of the cutter; however, interpolation is often performed by hard-wired components in the MCU. The MCU also permits the part program to be edited in the event that the program contains errors or changes in cutting conditions are required. Because the MCU is a computer, the term *computer numerical control* (CNC) is used to distinguish this type of NC from its technological predecessors, which were based entirely on hard-wired electronics.

The *processing equipment* accomplishes the sequence of processing steps to transform the starting workpart into a completed part. it operates under the control of the machine control unit according to the set of instructions contained in the part program.

2.9.2 Coordinate System and Motion Control in NC

A standard coordinate axis system is used to specify positions in numerical control. The system consists of the three linear axes *(x, y, z)* of the Cartesian coordinate system, plus three rotational axes *(a, b, c),* as shown in Figure 2.6. The rotational axes are used to rotate the workpart to present different surfaces for machining or to orient the tool or workhead at some angle relative to the part. Most NC systems do not require all six axes to function. The simplest NC systems (for example, plotters, pressworking machines for flat sheet-metal stock, and component insertion machines) are positioning systems whose Iocations can be defined in an *x-y* plane. Programming of these machines involves specifying a sequence of *x-y* coordinates. By contrast, some machine tools have five-axis control to shape complex workpart geometries. These systems typically include three Iinear axes plus two rotational axes.

In many NC systems, the relative movements between the processing element and the workpart are accomplished by fixing the part to a worktable and then controlling the positions and motions of the table relative to a stationary or semistationary workhead. Most machine tools and component insertion machines are based on this method of operation. In other systems, the workpart is held stationary and the workhead is moved along two or three axes. Flame cutters, *x-y* plotters, and coordinate measuring machines operate in this mode.

The coordinates for a rotational NC system are illustrated in Figure 2.6 These systems are associated with turning operations on NC lathes. Although the work rotates, this is not one of the controlled axes. The cutting path of the lathe tool relative to the rotating work piece is defined in the *x-z* plane, as shown in our figure.

Motion control systems based on NC can be divided into two types: (1) point-to-point and (2) continuous path. Point-to-point systems, also called positioning systems. Move the workhead (or workpiece) to a programmed location with no regard for the path taken to get to that Location. Once the move is completed, some processing action is accomplished by the workhead at the location. such as drilling or punching a hole. Thus the program consists of a series of point locations at which operations are performed.

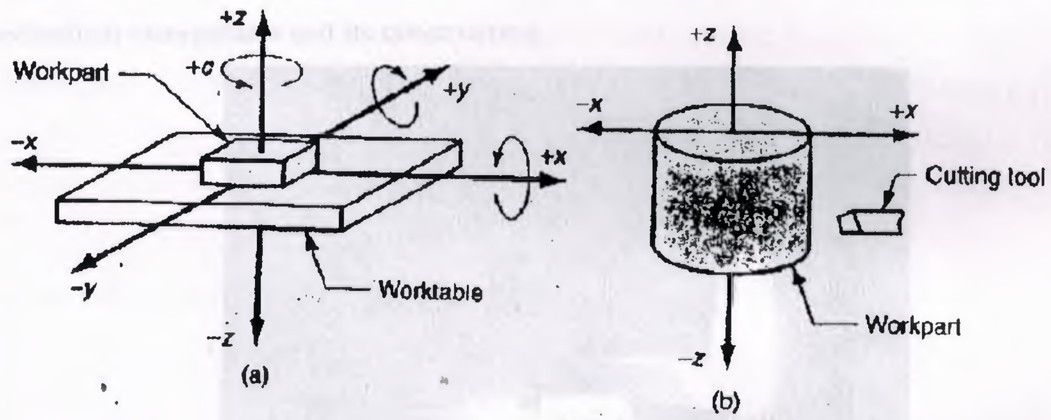


FIGURE 2.6 Coordinate systems used in numerical control (a) for flat and prismatic work and (b) for rotational work.

2.10 Industrial Robotics

Industrial robots are more used in FMS. An *industrial robots* is a general-purpose programmable machine possessing certain anthropomorphic features. The most apparent anthropomorphic, or humanlike, feature of an industrial robot is its mechanical arm, or manipulator. The control unit for a modem industrial robot is a computer that can be programmed to execute rather sophisticated subroutines, thus providing the robot with an intelligence that sometimes seems almost human. The robot's manipulator combined with a high-level controller allows an industrial robot to perform a variety of tasks, such as loading and unloading machine tools, spot welding automobile bodies, and spray painting. Robots are typically used as substitutes for human workers in these tasks.

The concept of a robot derives from a play written around 1920 About 40 years later, the first industrial robot was installed in a factory operation.

2.10.1. Robot Anatomy

An industrial robot consists of a mechanical manipulator and a controller to move it and perform other related functions. The *mechanical manipulator* consists of joints and links that can position and orient the end of the manipulator relative to its base. The controller unit consists of electronic hardware and software to operate the joints in a coordinated fashion to execute the programmed work cycle. Robot anatomy is concerned with the mechanical manipulator and its construction.

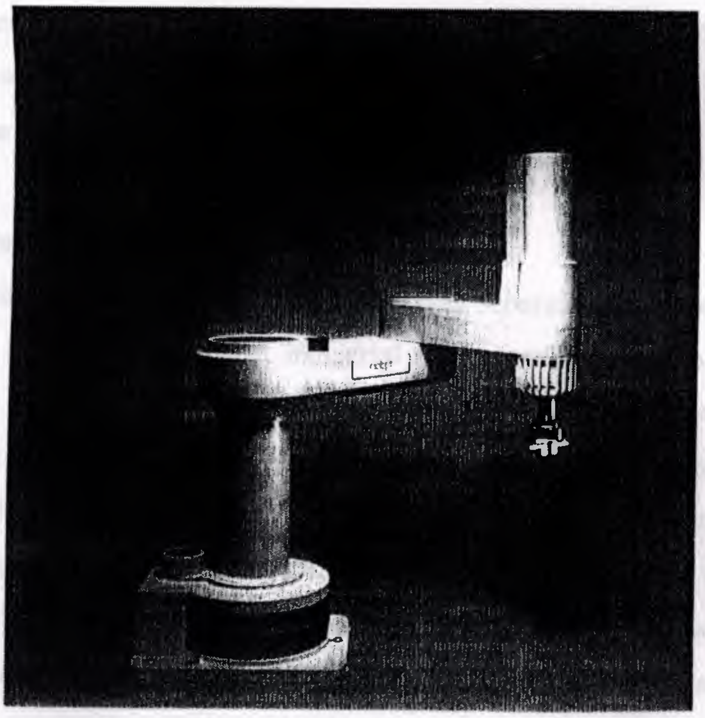


Figure 2.7 shows one of the common industrial robot configurations.

Industrial robotics has its roots in science fiction. The term robot was first used in a Czechoslovakian play written in the early 1920s by K. Capek. The Czech word "robota" means forced worker; when translated into English, the word was reduced to "robot." The play is called R.U.R. (for Rossum's Universal Robots), in which a scientist named Rossum creates a life form that goes out of control and attacks humans-technology run amok.
The first Unimate was installed in a die casting operation around 1961 at the Ford Motor Company. The robot's job was to unload castings from the die casting machine. The first commercial industrial robots were one-armed, hydraulically driven, heavy machines.

2.10.2 Manipulator Joints and Links

A joint in a robot is similar to a joint in a human body. It provides relative movement between two parts of the body. Connected to each' joint are an input link and an output link. Each joint moves its output link relative to its input link. The robot manipulator consists of a series of link-joint-link combinations. The output link of one joint is the input link for the next joint. Typical industrial robots have five or six joints. The coordinated movement of these joints gives the robot its ability to move, position, and orient objects and tools to perform useful work. Manipulator joints can be classified as linear or rotating, indicating the motion of the output link relative to the input link.

2.10.3 Manipulator Design

Using joints of the two basic types, each joint separated from the previous by a link, the manipulator is constructed. Most industrial robots are mounted to the floor. We can identify the base as link 0; this is the input link to joint 1 whose output is link 1, which is the input to joint 2 whose output link is link 2, and so forth, for the number of joints in the manipulator. Robot manipulators can usually be divided into two sections: (1) arm-and-body assembly, and (2) wrist assembly. There are typically three joints associated with the arm- and-body assembly, and two or three joints associated with the wrist. The arm-and-body serves a different function from the wrist. The function of the arm-and-body is to position an object or tool, and the wrist function is to properly orient the object or tool. Positioning is concerned with moving the part or tool from one location to another.

Orientation is concerned with precisely aligning the object relative to some stationary location in the work area.

To accomplish these functions, arm-and-body designs differ from those of the wrist. Positioning requires large spatial movements, while orientation requires twisting and rotating motions to align the part or tool relative to a fixed position in the workplace. The arm and body consists of large links and joints, whereas the wrist consists of short links. The arm-and-body joints often consist of both linear and rotating types, while the wrist joints are almost always rotating types.

Five basic arm-and-body configurations are available in commercial robots. The five types are identified in Figure 2.8 The design shown in part (e) of the figure and in Figure 2.7 is called a SCARA robot, which stands for "selectively compliant assembly robot arm." It is similar to a jointed arm anatomy, except that the shoulder and elbow joints have vertical axes of rotation, thus providing rigidity in the vertical direction but relative compliance in the horizontal direction.

The wrist is assembled to the last link in any of these arm-and-body configurations. The SCARA is sometimes an exception because it is almost always used for simple handling and assembly tasks involving vertical motions. Therefore, a wrist is not usually present at the end of its manipulator. Substituting for the wrist on the SCARA is usually , a gripper to grasp components for movement and/or assembly.

2.11 Work Volume and Precision of Motion

One important technical consideration for an industrial robot is the size of its work volume. Work volume is defined as the envelope within which a robot manipulator can position and orient the end of its wrist. This envelope is determined by the number of joints, as well as their types and ranges, and the sizes of the links. Work volume is important because it plays a significant role in determining which applications a robot can perform.

The definitions of control resolution, accuracy, and repeatability developed for NC positioning systems apply to industrial robots. A robot manipulator is after all, a positioning system. In general, the links and joints of robots are not nearly as rigid as their machine tool counterparts, and so the accuracy and repeatability of their movements are not as good.

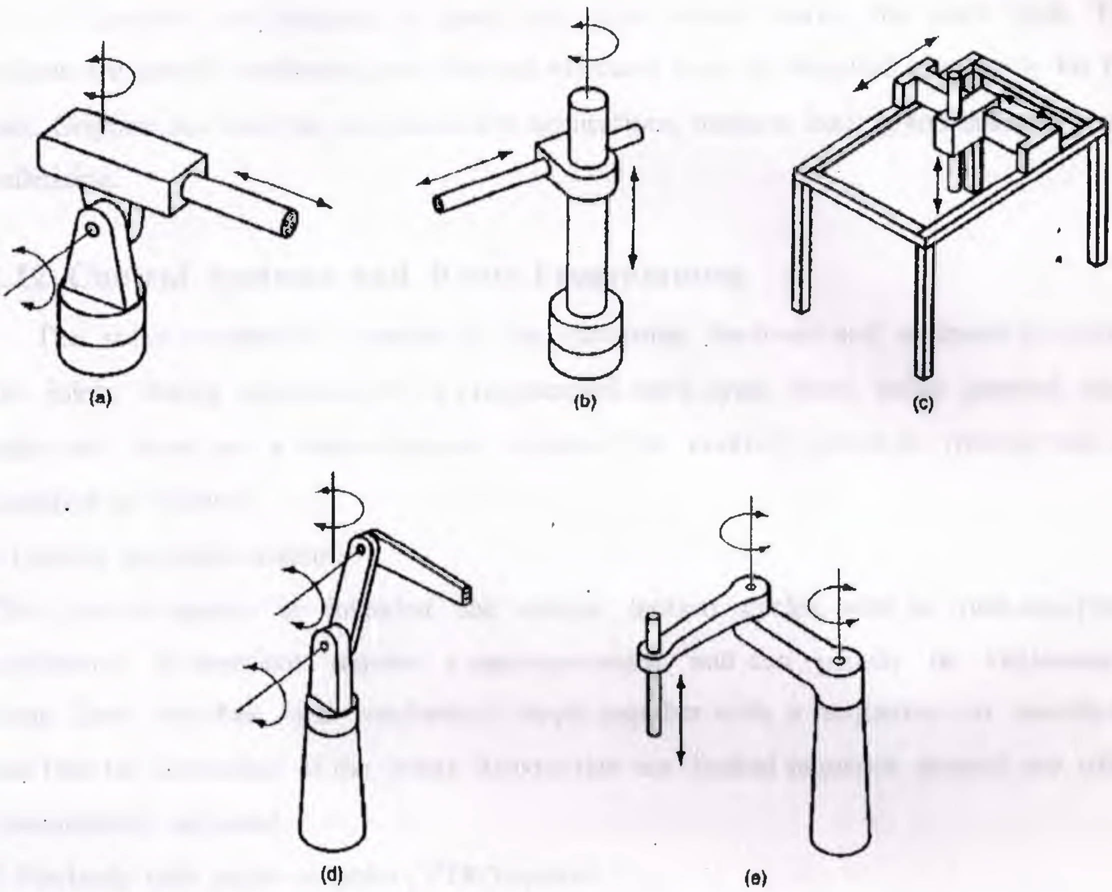


FIGURE 2.8 Five common anatomies of commercial industrial robots: (a) polar, (b) cylindrical, (c) Cartesian coordinate, (d) jointed arm, and (e) Scara, or selectively compliant assembly robot arm.

End Effectors An industrial robot is a general-purpose machine. For a robot to be useful in a particular application, it must be equipped with special tooling designed for the application. An *end effecter* is the special tooling that connects to the robot's wrist end to perform the specific task. There are two general types of end effectors : tools and grippers. A *tool* is used when the robot must perform a processing operation. The special tools include spot welding guns, arc welding tools, spray painting nozzles, rotating spindles, heating torches, and assembly tools (for example, an automatic screwdriver). The robot is programmed to manipulate the tool relative to the workpart being processed.

Grippers are designed to grasp and move objects during the work cycle. The objects are usually workparts, and the end effectors must be designed specifically for the part. Grippers are used for part placement applications, machine loading and unloading, and palletizing.

2.12 Control Systems and Robot Programming

The robot's controller consists of the electronic hardware and software to control the joints during execution of a programmed work cycle. Most robot control units today are based on a microcomputer system. The control system in robotics can be classified as follows.

1-Limited sequence control.

This control system is intended for simple motion cycles, such as pick-and-place application. It does not require a microprocessor and can usually be implemented using limit switches and mechanical stops, together with a sequencer to coordinate and time the actuation of the joints. Robots that use limited sequence control are often pneumatically actuated.

2-Playback with point- to-point (PTP) control.

As in numerical control, robot motion systems can be divided into point-to-point and continuous path. The program for a point-to-point playback robot consists of a series of point location and the sequence in which these points must be visited during the work cycles. During programming, this these points are recorded into memory and then subsequently played back during execution of the program. In point-to-point motions, the path taken to get to the final position is not controlled.

3- Playback with continuous path (CP) control.

Continuous path control is similar to PTP, except motion paths rather than individual points are stored in memory. In certain types of regular CP motions, such as a straight-line path between two points locations , the trajectory required by the manipulator is computed by the controller unit for each move. For irregular continuous motions , such as path followed in spray painting , the path is defined by the a series of closely spaced points that approximate the irregular smooth path. Robots capable of continuous path motions can also execute point-to-point movements.

4-Intelligent control.

Modern industrial robots exhibit characteristics that often make them appear to be acting intelligently. These characteristics include the ability to respond to sophisticated sensors such as machine vision, make decisions when things go wrong during the work cycle, make computations , and communicate with humans. Robot intelligence is implemented using controllers with powerful microprocessors and advanced programming techniques, that move decision on the base of knowledge base. Robots execute a stored program of instructions that defines the sequence of motions and positions in the work cycle, much like a part program in NC. In addition to motion instructions, the program may include instructions for other functions, such as interacting with external equipment, responding to sensors, and processing data.

Two basic methods are used to teach modem robots their programs: (1) lead through programming and (2) computer programming Languages.

Lead through programming involves a teach-by-showing method in which the manipulator is moved by the programmer through the sequence of positions in the work cycle. The controller records each position in memory for subsequent playback. Two procedures for Leading the robot through the motion sequence are available: powered Ieadthrough and manual Ieadthrough. In *powered leadthrough,* a control box is used to drive the manipulator. The control box, called a teach pendant, has toggle switches or press buttons to control the joints. Using the teach pendant, the programmer moves the manipulator to each Location, recording the corresponding joint positions into memory. Powered Ieadthrough is the common method for programming playback robots with point-to-point control. *Manual leadthrough* is typically used for playback robots with continuous path control. In this method, the programmer :physically moves the manipulator wrist through the motion cycle. For spray painting and certain other jobs, this is a more convenient means of pro-gamming the robot.

Computer programming languages for programming robots have evolved from the use of microcomputer controllers. The first commercial language was introduced around 1979 by

Animation, Inc. Computer languages provide a convenient way to integrate certain nonmotion functions into the work cycle, such as computations and data processing, decision logic, interlocking with other equipment, interfacing with sensors, and interrupts.

In chapter 4 the some operation algorithms of Intel lecture robots are described.

2.13 Summary

This chapter presented the layout types of FMS , flexibility of manufacturing systems, hardware and software components of FMS , and their characteristics.

CHAPTER 3

CONTROL AND SCHEDULING OF MANUFACTURING SYSTEM

3.1. Overview

One of the most difficult problems arising in flexible manufacturing systems (FMSs) is the scheduling problem. The scheduling problems encountered in an FMS can be separated into several distinct types which encompass a wide range of resources including parts, robots, machines, and AGVs. The different scheduling problems and apply sets of dispatching rules to each problem in an effort to evaluate the impact of various rules on the system performance is categorized in [3f,4f,5f]. Several researchers have since evaluated different problems under different sets of rules.

Studies have shown that combinations of dispatching rules over a system's production cycle can produce better performance than a single rule alone [6f]. Rapidly changing environments and dynamic flows of jobs can erode the advantages of a particular plan or strategy over time. For example, Yamamoto and Nof (1985) showed that a scheduling/rescheduling approach improved system performance 2-7% compared with fixed dispatching procedures when just machine breakdowns were taken into account[7]. From previous research, three points have become apparent with regard to on-line simulation:

1. There are several scheduling problems and associated dispatching rules to consider in FMS.
2. Both the multi-pass approach to real-time scheduling and planning, and the use of simulation as a combination analysis and control tool appear promising.
3. Separation of planning, scheduling, and execution functions introduces flexibility to a SFCS, and this implies that on-line simulation models should also have a clear separation between decision making and the physical characteristics of the system.

This chapter describes the scheduling and planning algorithms of FMS. The different scheduling algorithms, their graphical representation are given.

3.2. Planning, Scheduling and Control of Flexible Manufacturing Systems

Scheduling may refer to the following subsystems of a flexible manufacturing system: fabrication, machining, and assembly.

In [9] the hierarchical approach linking the machining and assembly system, where the overall FMS scheduling problem was structured as an aggregate scheduling (upper level) problem and real-time scheduling (lower level) problem is presented. At the aggregate level the scheduling problem was modeled as the two-machine flow shop problem and solved by Johnson's algorithm (Johnson, 1954). To solve the real-time scheduling problem, a heuristic algorithm was developed.

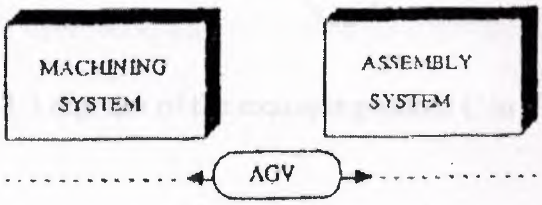


Figure 3.1 Structure of a manufacturing system

Consider a flexible manufacturing system that consists of a machining Subsystem and an assembly subsystem. The two subsystems are linked by a material handling carrier, for example, an automated guided vehicle (AGV), as shown in Figure 3.1

Consider an example product *C* with parts to be machined and then assembled (Figure 3.2). It consists of subassembly A_1, final assembly A_2, and three parts, P_1, P_2, P_3.

Parts P_1 and P_2 are to be machined before the subassembly A_1 is obtained. Assembling P_3 and A_1 results in the product *C* (final assembly A_2 in Figure 3.2).

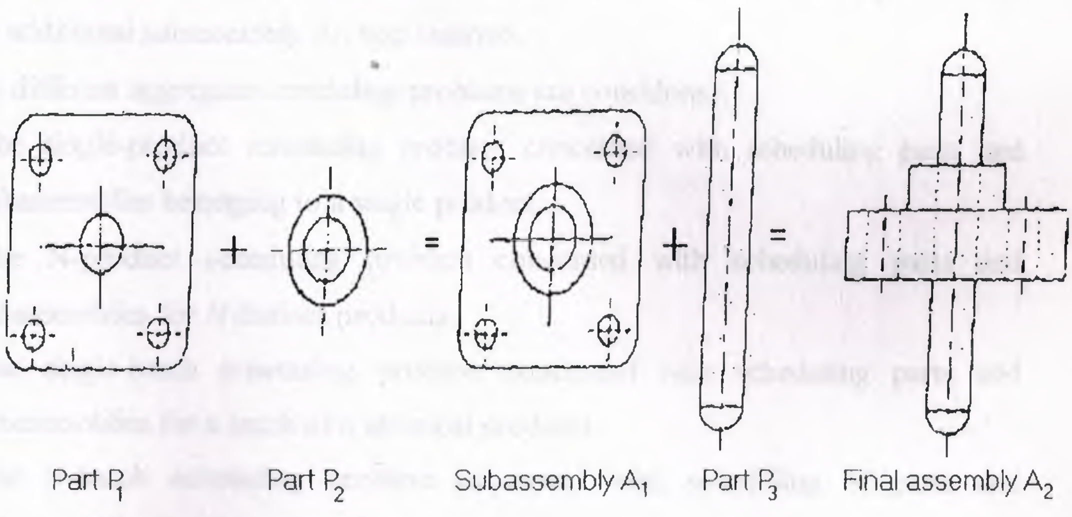


Figure 3.2. An example product C

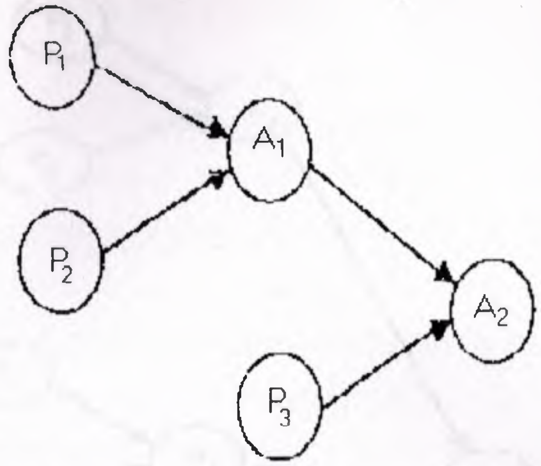


Figure 3.3 digraph of the example product C in Figure 3.2.

The precedence among machining and assembly operations for the product can be represented by a directed graph (digraph) shown in Figure 3.3. In this digraph any node of degree 1, i.e., with the number of edges incident to the node equal to 1, denotes a part; and any node of degree greater than I denotes a subassembly or a final product.

Another example of a digraph is shown in Figure 3.4(a). Without loss of generality, in this chapter, rather than representation of the digraph in Figure 3.4(a), the representation shown in Figure 3.4(b) is used. The latter representation does not allow one to assemble at a particular node more than one subassembly with any number of parts. At node A_3 in Figure 3.4(a), subassemblies A_1, A_2 and parts P_5, P_6 are assembled. The same subassembly A_3 has been obtained using the representation in Figure 3.4(b), where an additional subassembly A_{12} was inserted.

Four different aggregate scheduling problems are considered:

1. The single-product scheduling problem concerned with scheduling parts and subassemblies belonging to a single product.
2. The N-product scheduling problem concerned with scheduling parts and subassemblies for N distinct products.
3. The single-batch scheduling problem concerned with scheduling parts and subassemblies for a batch of n identical products.
4. The N-batch scheduling problem concerned with scheduling of parts and subassemblies for N batches of products.

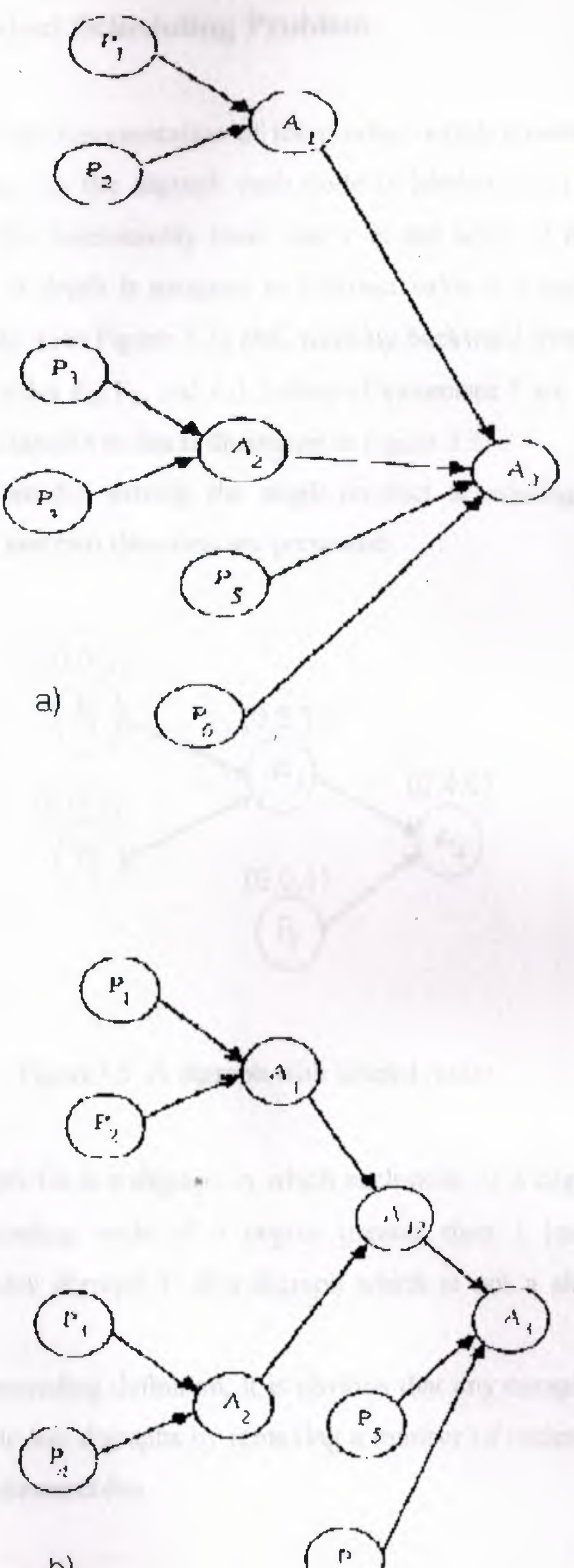


Figure 3.4 .Two different representations of the same product.

3.3 The Single-Product Scheduling Problem

Consider a digraph representation of the product which consists of a number of parts and subassemblies. In the digraph each node is labeled *(a,b,c)* where a, is the machining time, *b* is the subassembly time, and *c* is the level of depth of the node considered. The level of depth is assigned as follows: value 0 is assigned to the root node (for example, node A_2 in Figure 3.3) and, working backward from the root node to the initial nodes (i.e., nodes F_1, F_2, and F_3), values of increment 1 are assigned. Digraph G from Figure 3.3 with labeled nodes is illustrated in Figure 3.5.

Before an algorithm for solving the single-product scheduling problem will be developed, a definition and two theorems are presented.

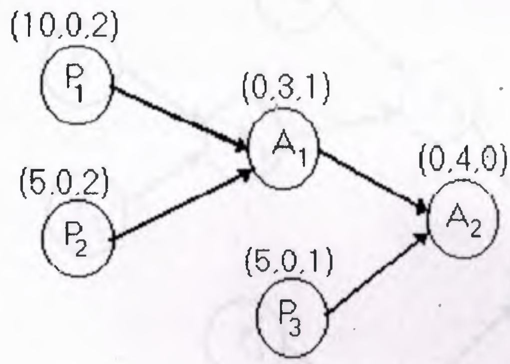


Figure3.5 A digraph with labeled nodes.

A simple digraph Gs is a digraph in which each node of a degree greater than 1 has at most one preceding node of a degree greater than 1 [see Figure 3.6(a)]. Consequently, a *complex digraph* C is a digraph which is not a simple digraph [see Figure 3.6(b)].

Based on the preceding definition, it is obvious that any complex digraph can be decomposed into simple sub digraphs by removing a number of nodes corresponding to the final assembly or subassemblies.

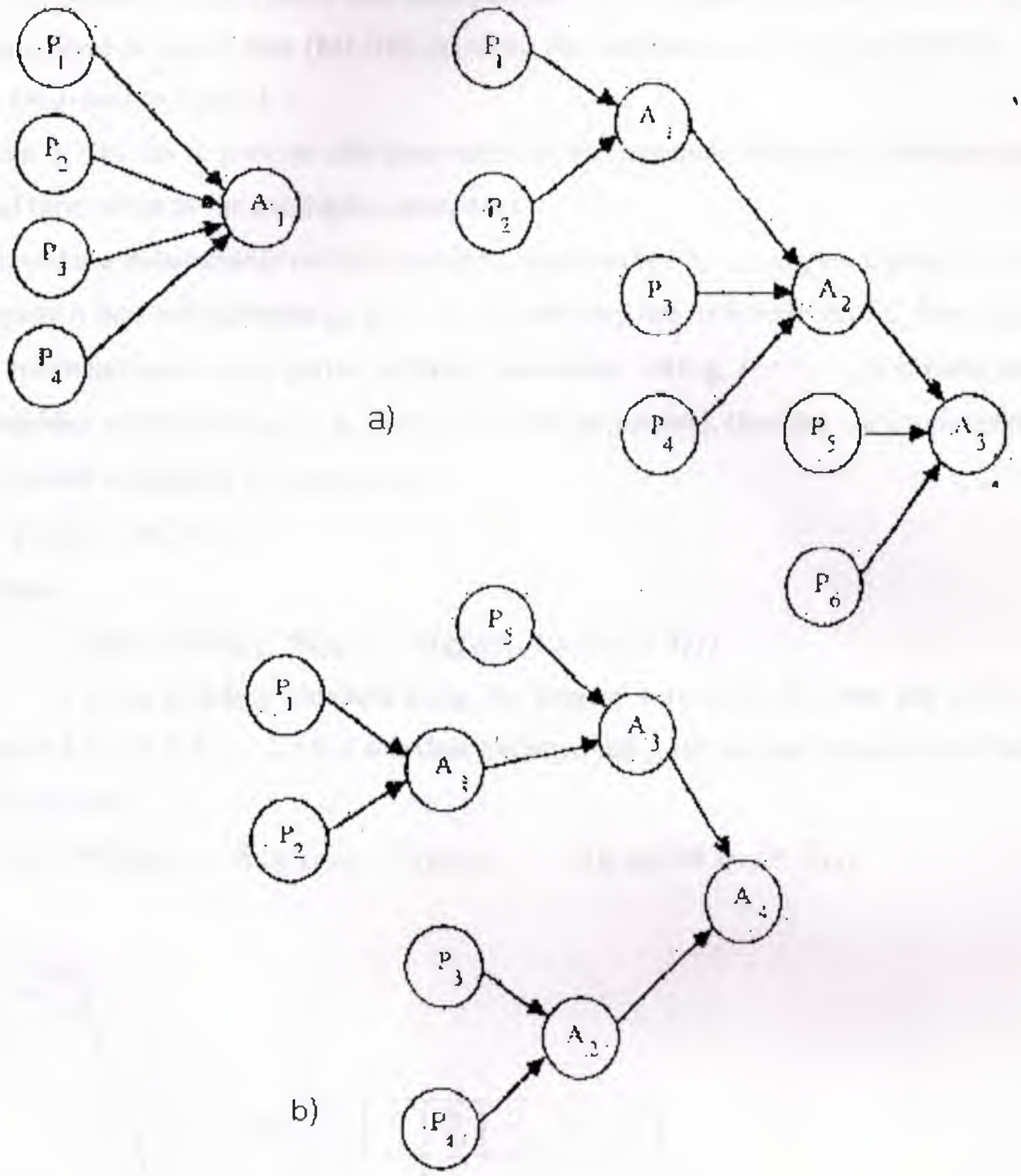


Figure 3.6 Example of two types of digraph: (a) two simple digraphs G_s and (b) complex digraph G.

Scheduling nodes (parts or subassemblies) of a simple digraph G. with the maximum level of depth first (MLDF) provides the minimum make-span schedule. This is illustrated in Figure 3.7

In Figure 3.7(b) the in-process idle time refers to the assembly subsystem, whereas the terminal time refers to the machining subsystem.

Scheduling nodes (parts and subassemblies) of a simple digraph G_s with the maximum level of depth first (MLDF) provides the minimum make-span schedule [ ]. This is illustrated in Figure3.7

In Figure 3.7(b) the in-process idle time refers to the assembly subsystem, whereas the terminal time refers to the machining subsystem.

Consider a subassembly or final product *C* represented by a complex digraph *C* and decompose it into sub digraphs $g_1, g_2, \ldots,$ g, by removing the root node v_0 of *C*. Let $S(g_i)$ be the minimum make-span partial schedule associated with g, $i = 1, \ldots, t$. if parts and subassemblies corresponding to g_i, and g_j, $I \neq j$ are preempted, then the minimum make-span schedule of product *C* is as follows:

$S(C) = \{S_1(G), S_2(C), v_0\}$

Where

$$S_1(G) = [S(g_{[1]}), S(g_{[2]}, \ldots, S(g_{[k]})], \text{ for } I_{[1]} \leq T_{[1]}$$

i = 1, . . . , *k* is a schedule obtained using the longest in-process idle time last (LITL) rule and i= *k*+ 1, k + 2, . . . , t is a schedule obtained using the longest terminal time first (LTTF) rule and

$$S_2(G) = [S(g_{[k+1]}), S(g_{[k+2]}, \ldots, S(g_{[k]}), \ldots, S(g_{[t]}), \text{ for } I_{[1]} > T_{[1]}$$

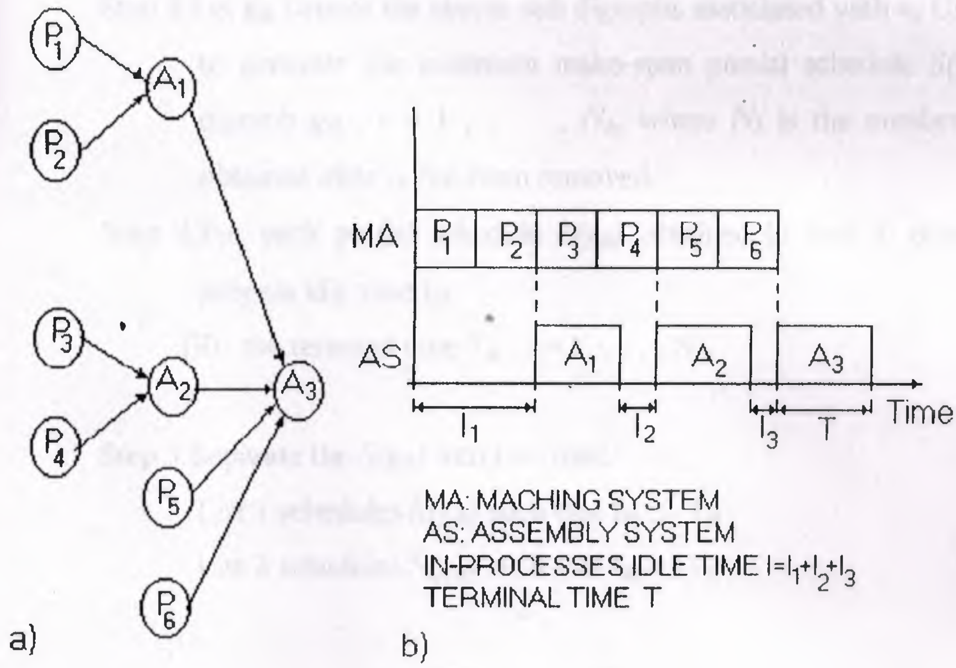


Figure 3.7 Application of the MDLF scheduling rule: (a) simple digraph illustrated Theorem 1 and (b) corresponding minimum make-span schedule.

NEAR EAST UNIVERSITY LIBRARY 1988 - LEFKOŞA

$i = k + 1, k + 2, \ldots, t$ is a schedule obtained using the longest terminal time first (LTTF) rule

Based on above mentioned Algorithm I is developed [3].

Algorithm 1 (The Single-Product Scheduling Problem)

Step 1.Label all nodes of the digraph G representing the structure of the product considered. If G is a simple digraph, use the MLDF rule to generate optimal schedule of product *C,* stop; otherwise, go to step 2.

Step 2.Remove root node v_0 from G and decompose it into sub digraphs g_l, l = 1 , . . , *L*.. If all g~ are simple digraphs, set $k = 0$ and go to step 3; otherwise, decompose each g_l which is not a simple digraph into a simple digraph by removing its root node. Let v_j denote a root node which has been removed, $j = 1, \ldots, J$.(Note that, for convenience, the removed nodes are numbered in the increasing order starting from the root node of *C.)* Set $k = J$ and go to step 3.

Step 3.Let g_{ik} denote the simple sub digraphs associated with v_k Use the MLDF rule to generate the minimum make-span partial schedule $S(g_{ik})$ for each sub digraph g_{ik}, $i = 1, \ldots, N_k$, where N_k is the number of sub digraphs obtained after v_k has been removed.

Step 4.For each partial schedule $S(g_{ik})$ obtained in step 3, determine (i) the in-process idle time I_{ik}

(ii) the terminal time T_{ik}, $i = 1, \ldots, N_k$

Step 5.Separate the $S(g_{ik})$ into two lists:

List 1 schedules $S(g_{ik})$ such that $I_{ik} \leq T_{ik}$

List 2 schedules $S(g_{ik})$ such that $I_{ik} > T_{ik}$,

$$i = 1, \ldots, N_k$$

Step 6.Use the LITL rule to generate

$$S_1(g_k) = [S(g_{[1]k}), S(g_{[2]k}), \ldots, S(g_{[r]k})],$$

for $S(g_{ik})$ in list 1, $i = 1$, . . . , r

and use the LITF rule to generate

$S_2(g_k) = [S(g_{[r+1]k}), S(g)_{[r+2]k}), \ldots, S(g_{[r]k})]$,

for $S(g_{ik})$ in list 2, $i = r + 1, \ldots, t; t = N_k$

Then generate the partial schedule $S(g_k)$ $[S_1(g_k), S_2(g_k), v_k]$

Step 7.If $v_k = v_0$, then $S(C) = S(g_k)$ is the optimal schedule, stop; otherwise, go to step 8.

Step 8.Consider S(Ck) as a simple sub digraph schedule and calculate I_k and T_k. Set $k = k - 1$ and go to step 3.

3.4 The N-Product Scheduling Problem

In this section, the scheduling problem for N distinct products, each in quantity of one, is considered. To solve this problem, a "product-by-product" policy is used. The product-by-product policy assumes that the N product scheduling problem can be decomposed into N single-product scheduling problems. The algorithm for scheduling of N products C_1, C_2 C~ is presented below.

Algorithm 2 (The N-Product Scheduling Problem)

Step I. Using Algorithm 1, determine the optimal schedule 5(Q) for each product $C_i = 1, \ldots, N$.

Step 2. Separate all $S(C,)$ into the following two lists:

List 1: including $S(C_i)$ such that $I_i \leq T_i$, $i = 1, \ldots, k$

List 2: including $S(C_i)$ such that $I_i > T_i$, $i = k + 1, \ldots, N$

Step 3. For the schedules in list 1, develop the LITL schedule:

$S_1(NC) = \{S(C_{[1]}), 5(C_{[2]}), \ldots, S(C_{[k]})\}$

For the schedules in list 2, develop the LTTF schedule;

$$S_2(NC) = \{S(C_{[k+1]}), S(C_{[k+2]}), \ldots, S(C_{[N]})\}$$

Step 4. Generate final schedule $S(NC) = \{S_1(NC), S_2(NC)\}$

The schedule $S(NC) = \{S_1(NC), S_2(NC)\}$ generated by Algorithm 2 is the minimum make-span schedule of the N-product scheduling problem [ ].

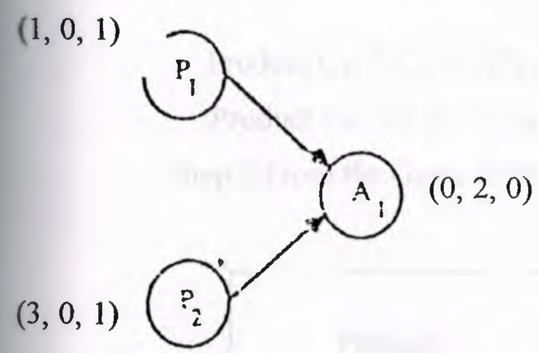
Algorithm 2 is illustrated in the following Example.

Consider an $N = 4$ product-scheduling problem with each product structure shown in Figure 3.8. For simplicity, assume that the structure of each product is represented by a simple digraph.

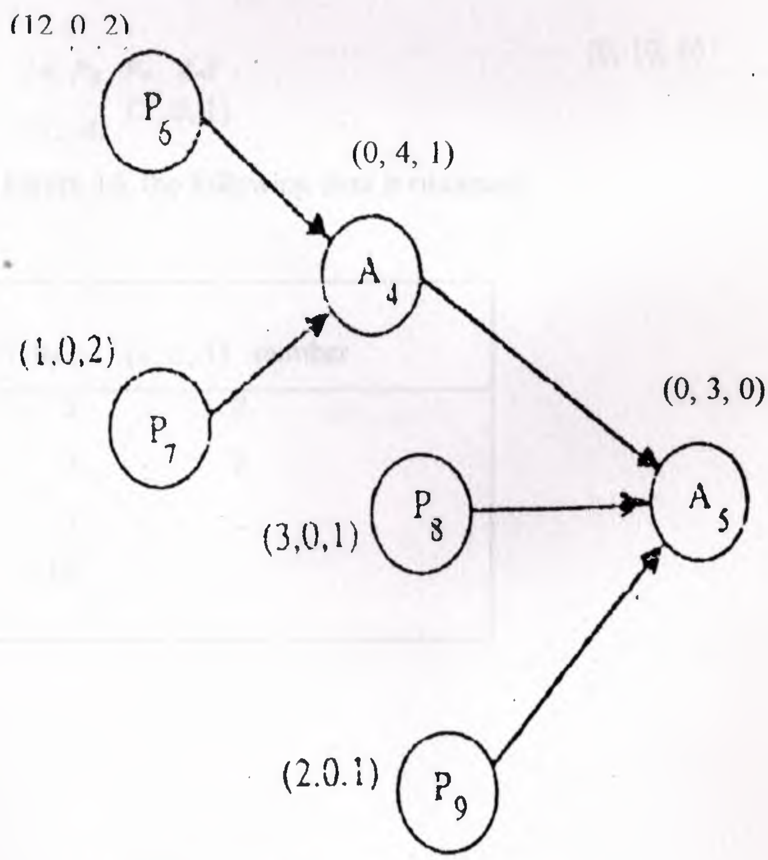
Step 1.Using Algorithm 1, the following schedules (illustrated in Figure 3.9) are obtained:

Product C_1: $S(C_1) = \{P_1, P_2, A_1\}$

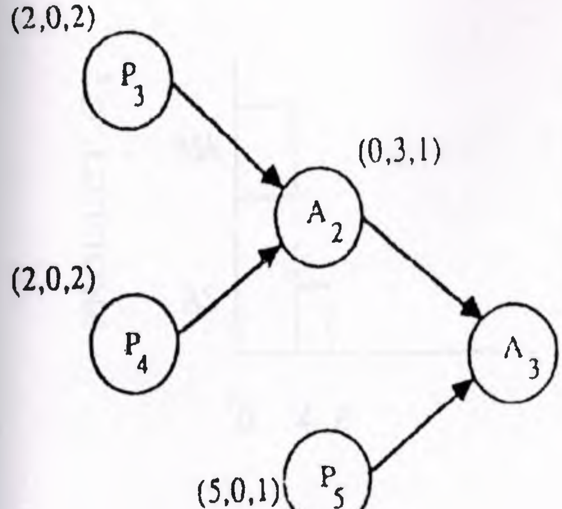
Product C_2: $S(C_2) = \{(P_3, P_4, A_2), P_5, A_3\}$

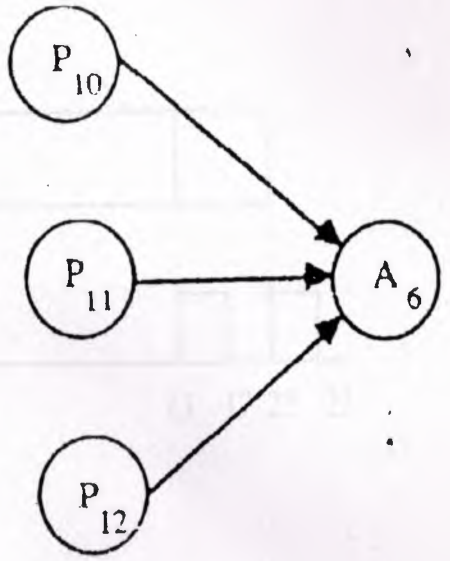


Product C_1



Product C_3





Product C_2 Product C_4

(3, 0, 1)

Figure 3.8 Structure of products C_1, C_2, C_3, and C_4.

(0, 10, 0)

Product C_3: $S(C_3) = \{(P_6, P_7, A4, P_8, P_9, A_5\}$

Product C_4: $5(C_4) = \{P_{10}, P_{11}, P_{12}, A_6$ (2, 0, 1)

Step 2.From the Gantt charts in Figure 16, the following data is obtained:

Product	(0, 3, 0)	(4, 0, 1)	number
C_1	4	2	2
C_2	6	3	2
C_3	18	3	2
C_4	9	10	1

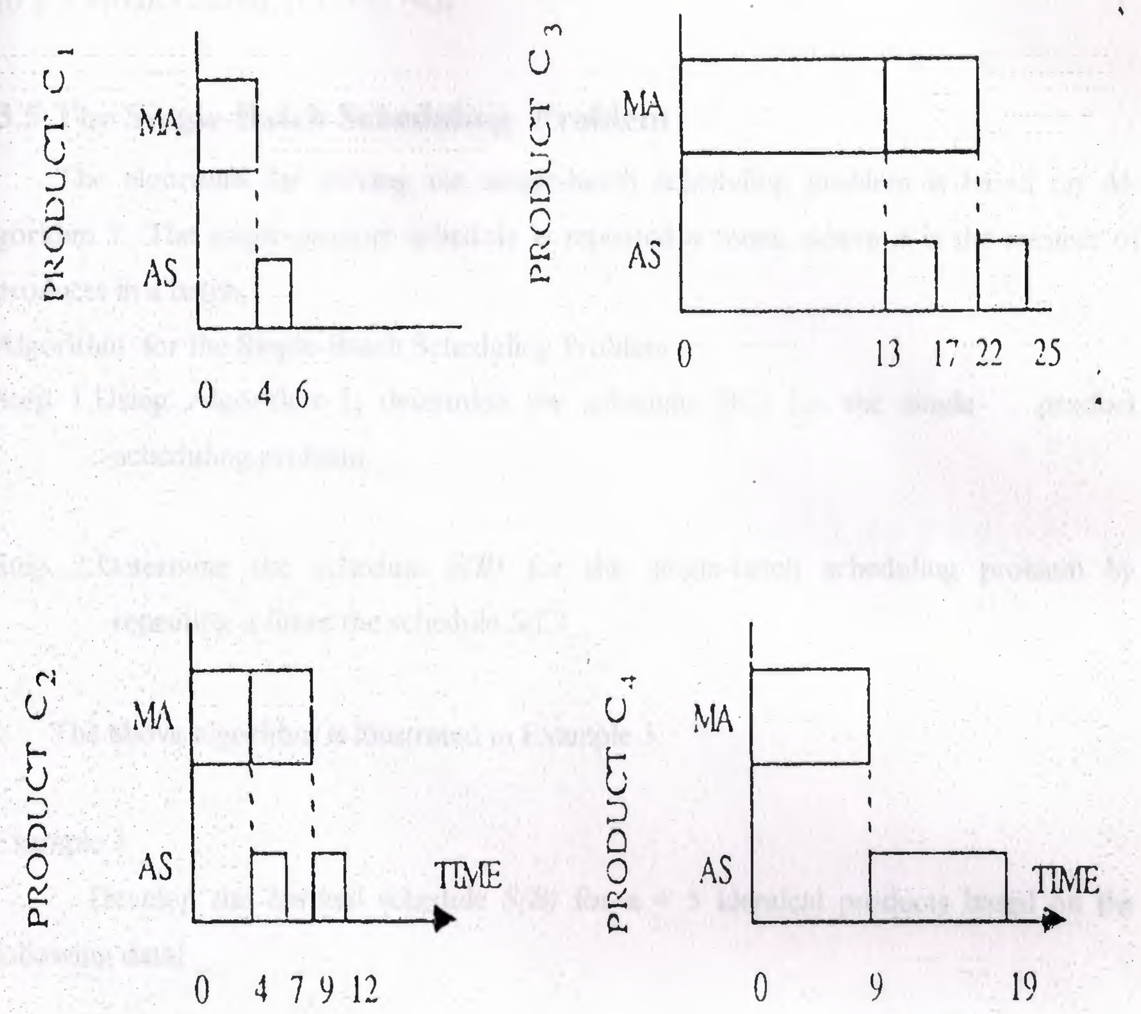


Figure 3.9 Schedules for the products in Figure 3.8

Step 3.Using the LITL and LTTF rules results in the following schedules:

$$S_1(NC) = \{C_4\}$$

$$S_2(NC) = (C_3, C_2, C_4\}$$

Step 4.The optimal schedule is as follows;

$S(NC) = \{[(P_{10}, P_1, P_{12}, A_6)], [(P_6, P_7, A_4), P_8, P_9, A_5],$

$[(P_3, P_4, A_2), P5, A_3], [P_1, P_2, A_1]\}$

3.5 The Single-Batch Scheduling Problem

The algorithm for solving the single-batch scheduling problem is based on Algorithm 1. The single-product schedule is repeated *n* times, where *n* is the number of products in a batch.

Algorithm for the Single-Batch Scheduling Problem

Step 1.Using Algorithm 1; determine the schedule *S(C)* for the single- product scheduling problem.

Step 2.Determine the schedule *S(B)* for the single-batch scheduling problem by repeating *a* times the schedule *S(C)*

The above algorithm is illustrated in Example 3.

Example 3

Develop the optimal schedule *S(B)* for a = 5 identical products based on the following data;

Part or, Subassembly	P_1	P_2	P_3	A_1	A_2
Machining time	10	5	6	-	-
Assembly time	-	-	-	4	3

and precedence constraints:

$P_1, P_2 \longrightarrow A_1$

$P_3, A_1 \longrightarrow A_2$

Step 1.Solving the single-product scheduling problem, the following schedule is obtained:

$S(C) = \{(P_1, P_2, A_1), P_3, A_2\}$

Step 2.The optimal schedule *S(B)* is generated as follows:

$$S(B) = \{[P_1, P_2, A_1), P_3, A_2], [(P_1, P_2, A_1), P_3, A_2], [(P_1, P_2, A_1), P_3, A_2], [(P_1, P_2, A_1), P_3, A_2], [(P_1, P_2, A_1), P_3, A_2]\}$$

3.6 The N-Batch Scheduling Problem

The N–batch Schduling problem can be decomposed in to N single-batch scheduling problems.

Algoritm 4 (The N-Batch Scheduling Problem)

Step 1. Using Algorithm 2, determine the schedule S(NC) for the N product scheduling problem.

Step 2 determine the schedule S(NB) for the N-batch scheduling problem by repeating n_i times each schedule $S(C_i)$ in S(NC), where n_i is a batch size of product C_i.

3.7 Summary

This chapter presented the scheduling and planning algorithms of FMS and the different scheduling algorithms, their graphical representation.

CHAPTER 4

FUZZY INTELLIGENT ROBOTS

4.1. Overview

In FMS we often have problems of control of workstations in the condition of uncertainty of environment, fuzziness of information. One of effective means to solve this problem is the use of fuzzy technology invented by Zadeh [13]. Fuzzy systems provide a rich and meaningful addition to standard logic. In order to construct a fuzzy application a sufficient knowledge on how operates the system that is to be controlled is required. In the fuzzy system design methodology, we are concerned to write down a set of rules on how to make decision, then we incorporate these rules into fuzzy controller that emulates the decision-making process. The performance of the fuzzy controller can be influenced by changing the shape and number of its membership functions, by changing its defuzzification method and its inference mechanism. These operations can be done in relatively easy manner without need for knowledge of all system parameters and without use of mathematical operations of any kind.

In this chapter the development of fuzzy inference system for robot that determine the quality of products is considered.

4.2. Linguistic Variables

The research lately have shown that conventional analysis methods for systems analysis and computer modeling, based on precise processing of numerical data, are not capable of dealing with huge complexity of real technological processes. This leads to the fact that in order to get decisions affecting the behavior of those processes we need to reject of traditional requirements to measurement accuracy, which are necessary for mathematical analysis of precisely defined mechanical systems.

The necessity to sacrifice the precision and determinate is dictated also by the appearance of some classes of control problems that are connected with decision making by operator in the "man-computer" interface. Implementation of the dialog in such interface is impossible without application of languages close to natural ones and capable of describing fuzzy categories near to human notions and imaginations. In this connection, it is valuable to use the notion of linguistic variable first introduced by

L.Zadeh[5]. Such linguistic variables allow an adequate reflection of approximate in-word descriptions of objects and phenomena in the case if there is no any precise deterministic description. It should note as well that many fuzzy categories described linguistically even appear to be more informative than precise descriptions.

To specify rules for the rule base, the expert will use a linguistic description; hence, linguistic expressions are needed for the inputs and the outputs and the characteristics of the inputs and outputs. Here the linguistic variables (constant symbolic descriptions of what are in general time-varying quantities) will be used to describe fuzzy system inputs x_i and outputs y_i.

4.3. Linguistic Values

Linguistic variables x_i and y_i take on linguistic values that are used to describe characteristics of the variables. Let A_i^j denote the j^{th} linguistic value of the linguistic variable x_i. Defined over universe of discourse X_i. If we assume that there exist many linguistic values defined over X_i, then the linguistic variable x_i takes on the elements from the set of linguistic values denoted by

$$A_i = \{A_i^j : j = 1,2,......,N\}$$

(sometimes for convenience we will let the j indices take on negative integer values). Similarly, let B_i^j denote the j^{th} linguistic variable y_i defined over the universe of discourse Y_i. The linguistic variable y_i takes on elements from the set of linguistic values denoted by

$$B_i = \{B_i^p : p = 1,2,.....,M_i\}$$

(sometimes for convenience we will let the p indices take on negative integer values). Linguistic values are generally descriptive terms such as "positive large", "zero" and "negative big". For example, assume that u_1 denotes the linguistic variable "speed", then it is possible to assign A_1^1 = "slow", A_1^2 = "medium", A_1^3 = "fast" so that u_1 has a value from $A_1 = \{A_1^1, A_1^2, A_1^3\}$. Here A_1^1 = "slow", A_1^2 = "medium", A_1^3 = "fast" are called term set.

Another important aspect of the notion of linguistic variable that a linguistic variable is associated with the two rules: the syntactic rule, which can be set as a grammar, generating names for the variable; and the semantic rule, which determines an algorithmic procedure for calculating the meaning of each value. Thus these rules make the essential part of the description of the structure of linguistic variable.

4.4. Rule Base

The mapping of the inputs to the outputs for a fuzzy system is in part characterized by a set of condition $\rightarrow$ action rules, or in *modus ponens* (If-Then) form,

If premise **Then** consequent (2.1)

Usually, the inputs of the fuzzy systems are associated with the premise, and the outputs are associated with the consequence. These If-Then rules can be represented in many forms. Two standard forms, multi-input multi-output (MIMO) and multi-input single-output (MISO), are considered here. The MISO form of a linguistic rule is

If u_1 is A_1^j and u_2 is A_2^k and ,....., and u_n is A_n^l Then y_q is B_q^p (2.2)

It is an entire set of linguistic rules of this form that the expert specifies on how to control the system. Note that if u_1="velocity error" and A_1^j= "positive large", then "u_1 is A_1^j", a single term in the premise of the rule, means "velocity error is positive large". It can be easily shown that the MIMO form for a rule (i.e. one with consequents that have terms MISO rules using simple rules from logic. For instance, the MIMO rule with n inputs and m =2 outputs

If u_1 is A_1^j and u_2 is A_2^k and ,....., and u_n is A_n^l Then y_1 is B_1^r and y_2 is B_2^s

Is linguistically (logically) equivalent to the two rules

If u_1 is A_1^j and u_2 is A_2^k and ,....., and u_n is A_n^l Then y_1 is B_1^r

If u_1 is A_1^j and u_2 is A_2^k and ,....., and u_n is A_n^l Then y_2 is B_2^s

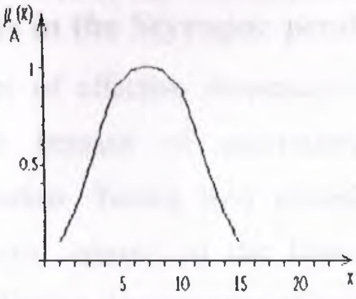


Figure 4.1 Fuzzy set.

The representation of temperature within a range $[T_1, T_2]$ by fuzzy and crisp sets is shown in Figure 2.2. In the first case we use membership function $[T_1, T_2] \rightarrow [0,1]$ for describing linguistic concepts "**cold**", "**normal**", "**warm**". In the second case right - open intervals are used for describing of traditional variable by crisp sets. Fuzzy sets with crisply defined membership functions are called ordinary fuzzy sets.

If membership function of a fuzzy set A assigns to each element of the universal set X a closed interval of real numbers then this type of fuzzy sets are called interval-valued fuzzy sets

$$\mu_A : X \rightarrow \varepsilon[0,1]$$

where $\varepsilon[0,1]$ denotes the family of all closed intervals of real members in $[0,1]$.

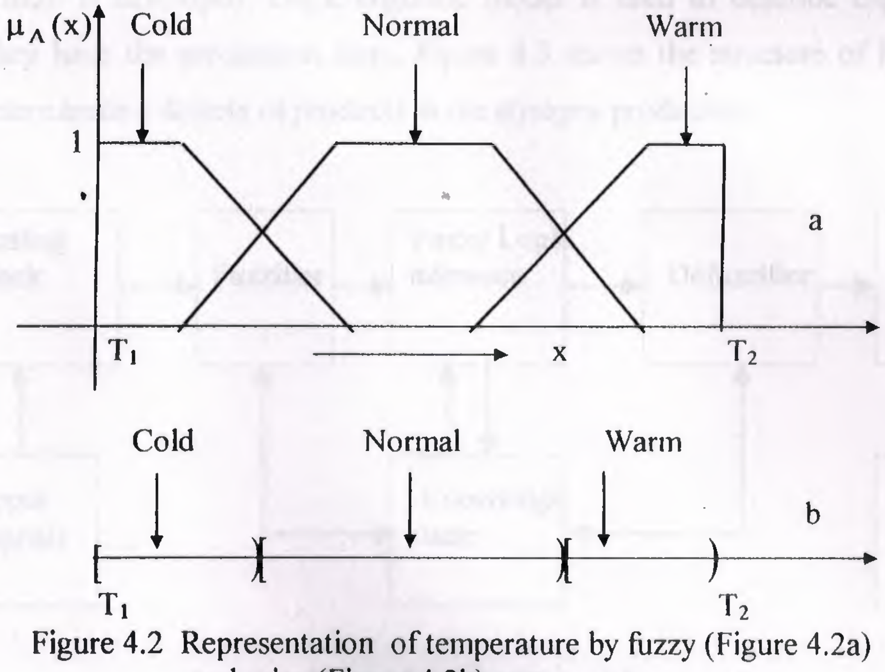


Figure 4.2 Representation of temperature by fuzzy (Figure 4.2a) and crisp (Figure 4.2b) sets.

4.6 Fuzzy Intelligent Robot for the detection and withdrawal of Defect Parts in the Styropor production

The development of effective deterministic models is very difficult in some technological processes because of uncertainty and fuzziness of their works, insufficiency of information. Taking into consideration all these, it is actually to develop intelligent control system on the base of knowledge of the experienced specialists and experts. During development of such system it is necessary to take into account specific characters of technological processes. For such cases one of effective means for information processing is the use of fuzzy logic. Using fuzzy conceptions for estimation of situations and creation of logic rules in control models simplifies problems decision and information processing.

In this chapter the development of fuzzy system for determination defect products in conveyor line of styropor productions is considered.

In present time the definition of quality of EPS sheets is carried out by analyzing characteristics of products. To those parameters curvature, density and sizes of EPS sheets are belonged.

The synthesis processes of fuzzy system including measured and calculated information are considered. Fuzzy system as input signals use curvature and density of EPS sheets. Using above parameters on the base of expert knowledge the fuzzy system is developed. Logic-linguistic model is used to describe expert knowledge. They have the production form. Figure 4.3 shows the structure of fuzzy system for determination defects of products in the styropor production.

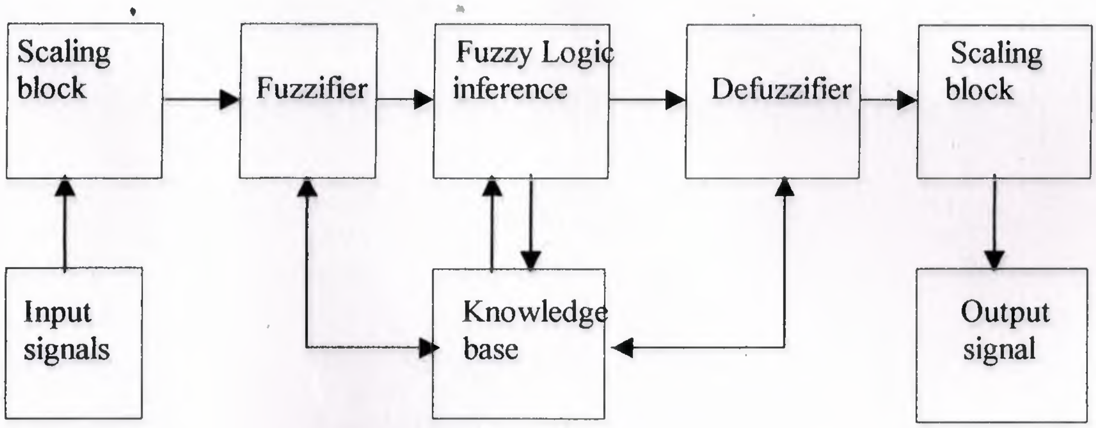


Figure 4.3. Structure of fuzzy system

The input signals by operators are given to the system input. Those signals values after scaling enters to fuzzifier. Fuzzifier, machine logic inference, knowledge base and defuzzifier are main blocks of fuzzy system. Knowledge base (KB) consists of production rules, which have IF...THEN... form. It is constructed on the base of experimental data and knowledge of experts and specialists. In figure 4.4 a fragment of knowledge base is given

IF "X1=N" and "X2=S" THEN "Y=L"
IF "X1=N" and "X2=N" THEN "Y=N"
IF "X1=S" and "X2=S" THEN "Y=L"
IF "X1=B" and "X2=N" THEN "Y=L"
IF "X1=VS" and "X2=B" THEN "Y=H"
IF "X1=B" and "X2=VS" THEN "Y=L"

Figure 4.4. Fragment of knowledge base

Here N is normal, S is small, B is big, VB is very big, VS is very small, L is low, H is high are linguistic value of fuzzy terms. Every fuzzy variable has five terms.

The forms of the terms of fuzzy signals are taken in the triangle form (figure 4.5). Their membership function is determined by the following formula.

$$\mu(x)=\begin{cases}1-\dfrac{\bar{x}-x}{\alpha}, & \bar{x}-\alpha \le x \le \bar{x}\\ 1-\dfrac{x-\bar{x}}{\beta}, & \bar{x} < x \le \bar{x}+\beta\\ 0, & \text{in other case}\end{cases} \quad (1)$$

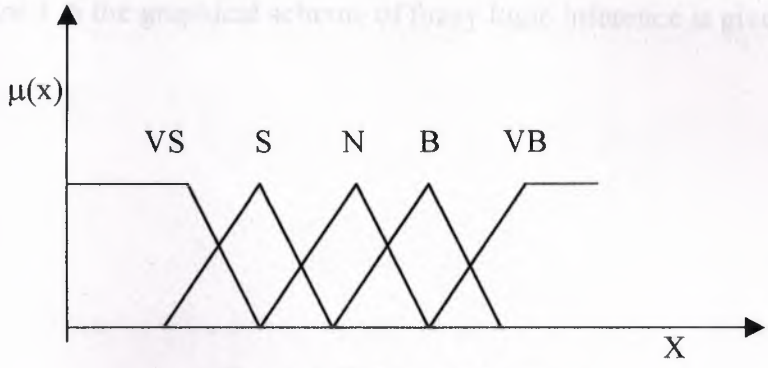


Figure 4. 5. Linguistic terms

Logic machine inference is performed by the following manner.

$$\tilde{X1}, \tilde{X2} \rightarrow \tilde{Y} \quad (2)$$

The membership degrees of the input signals for each active rules in knowledge base are defined. Then the following formula is used.

For left side:

$$\mu(x_i) = 1 - \frac{x_i^m - x_i}{x_i^m - x_i^l}, \qquad i = \overline{1,5} \quad (3)$$

For right side:

$$\mu(x_i) = 1 - \frac{x_i - x_i^m}{x_i^r - x_i^m}, \qquad i = \overline{1,5} \quad (4)$$

Fuzzy logic inference is performed using max-min composition of Zade.

$$\mu(y) = \max_{x1,x2} \min\{\mu_{X1}(x1), \mu_{X2}(x2), \mu_R(x1, x2, y)\} \quad (5)$$

Using the "Center of gravity" method the defuzzification process of fuzzy output signal is performed

.

$$y = \frac{\sum_{i=1}^{n} \mu(y_i) * y_i}{\sum_{i=1}^{n} \mu(y_i)} \quad (6)$$

In figure 4 .6 the graphical scheme of fuzzy logic inference is given.

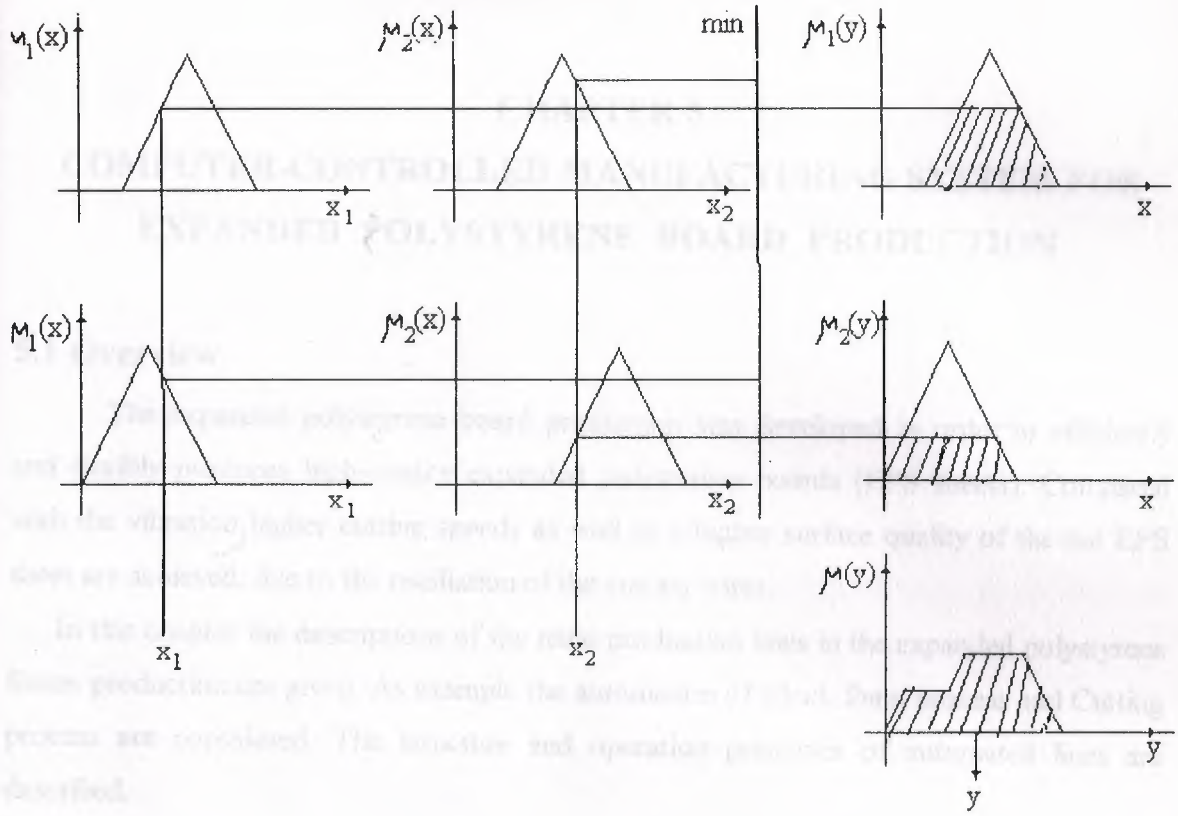


Figure 4.6. Graphical scheme of fuzzy logic inference.

Developed system is applied for quality estimation of EPS sheets in styropor production. For this process interval of input signals varies in the following manner. For curvature of EPS sheets – 1 – 5mm, density – 15 , 20. Using those parameters, on the base of experimental data and expert knowledge, for given processes the fuzzy knowledge base is developed. Using fuzzy logic inference the on-line determination of the qualities of EPS sheets in conveyor lines are carried out. Acquired results have been confirmed in on-line regime.

Suggested system allows using systematical approach and making decision on quality of EPS sheets in on-line working regime.

4.7 Summary

This chapter presented the development of fuzzy inference system for robot that determine the quality of products.

CHAPTER 5

COMPUTER-CONTROLLED MANUFACTURING SYSTEM FOR EXPANDED POLYSTYRENE BOARD PRODUCTION

5.1 Overview

The expanded polystyrene board production was developed in order to efficiently and flexibly produces high-quality expanded polystyrene boards (EPS sheets). Compared with the vibration, higher cutting speeds as well as a higher surface quality of the cut EPS sheet are achieved, due to the oscillation of the cutting wires.

In this chapter the descriptions of the main production lines in the expanded polystyrene foams production are given. As example the automation of Block foam process and Cutting process are considered. The structure and operation principles of automated lines are described.

5.2 Description of the Expanded Polystyrene Foam production

Styropor foams have proven themselves successful in the solution of insulation problems in the building trade for over 40 years. Styropor is an expandable polystyrene , in blocks ,sheet and moldings. Expanded styropor rigid foam boards posses high tensile strength and even at the lowest density are resistant to compression and vibration. Their humidity absorption in still air is practically nil; their water absorption when immersed is very low. Styropor rigid foam boards are resistant to acids, alkalis and ageing and are rotproof. They may be used between temperatures of + 90^0 C and - 200^0 C. They are used as a thermal insulation material, footstep noise abatement, and decorative panels.

The conversion of styropor to foam panels involves three steps of production lines.

1- Pre foaming process
2- Block foam process
3- Cutting process

5.2.1 Prefoaming Process:

There are several methods of carrying this out. It is faster to prefoam with steam than with hot water. In our process we prefoaming with steam. We use a pre-expansion (prefoaming) machine (2) which is shown in processing design. The raw beads (0.6mm - 1.4 mm diameter) are poured in the machine and preexpanded by using steam. During the pre-expansion of styropor beads the cells are filled with a mixture of pentane vapour and water vapour. While the beads are in this condition they are not suitable for molding. The beads should be dried in to the silos (3) by allowing the air. Then the beads are transferred to the silos. If the pre-expanded beads are transferred to the silos by air conveyor much of the moisture can be removed before the beads reach the silo .

5.2.2. Block –foaming process.

After drying in the silos the pre-expanded beads are ready for moulding. The moulds are made from corrosion resistant alloys or from aluminum, Steel, brass or zinc. The moulds are made of 2 or more parts and should be so constructed that the cavity in to which the styropor is filled can be firmly closed but is not air tight. We have two moulds machines in our process (4) One of them has a dimensions of 1000 x 2000 x 420 mm and the other one has 1200 x 2400 x 550 mm . On the walls of the moulds have perforations about 2 mm or perforation plugs which allow the steam to enter in to the mould cavity. The beads are transferred in to the moulds by an air conveyor. After filling the mould with beads the steam injected in to the mould through the perforations. The block product is ready to transfer in to the storage. The blocks should be stored in a place for drying. The blocks are transferred in to the storage by a belt conveyor.

5.2.3 Cutting process:

The blocks after drying in to the storage are transferred to cutting department by a belt conveyer. They are ready for cutting in sheets. Expanded Styropor is widely used in the form of board or thin sheet, both of which are commonly made by cutting and slicing large blocks. Cutting with hot wires is one of the most widely used methods of obtaining board from blocks. Mechanical methods of cutting also frequently used: band saws or band-knives, and reciprocating saws, are the most usual tools. In hot – wire cutting an electrically heated wire is forced through the material , which softens in the vicinity of the wire and offers little resistance. High – tensile resistance wire is used , such as Cr – Ni 30 special.

When board is cut from expanded styropor blocks a number of parallel hot wires are used. These are kept under tension by springs so that they do not become slack when heated. The voltage across the wires is adjusted to maintain them at a temperature of 180 – 220 ^{0}C , but should not exceed 42 V. The portions of the wires not in contact with expanded styropor are air – cooled to prevent overheating and widening of the cut at the edges of the material. The speed of cutting , thickness of the cut, and quality of the cut are affected by the density and moisture content of the expanded material, the thickness of the wire, and other factors.

After cutting operations the sheets are ready for delivery,

5.3Applications of expanded foam materials

Expanded styropor is very widely used as an insulating material, in building industry and other applications

1- Refrigeration;

Cold stores, deep freeze containers, refrigerators, insulation for pipes carrying chilled liquids, refrigerator trucks, and refrigerator ships.

2- Building

Thermal insulation of ceilings, walls, roofs etc. ,production of prefabricated units, sound absorption, light weight concrete.

3- Packing

Insulated containers, gift packages, large scale packing of small articles.

4- Decorative articles, toys, and display articles

5- Marine articles

Life belts and life jackets, floats, buoyancy in life boats.

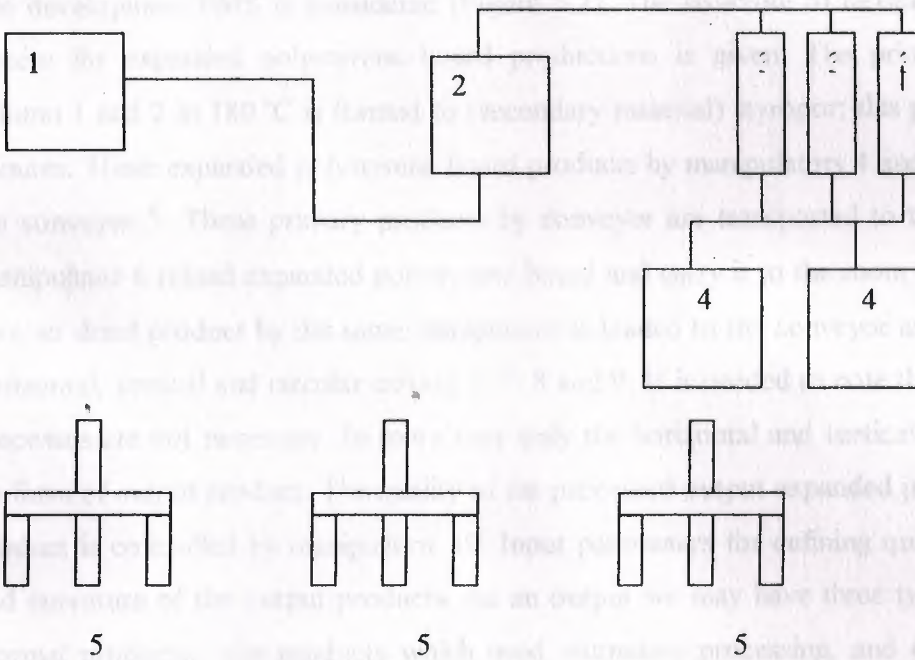


Figure 5.1 The styropor processing plant

5.4 Structure of FMS for EPS sheets production

For the productions, producing different kind of products, the developer flexi able computer-aided control system is in primary concern. The main flexibility in these productions is the reconstruction of the program for control of (machinery) technological process. In this subchapter the structure of FMS for styropor productions is considered.

As described above the styropor productions is characterized by different kind of output products. Each output product is characterized by its own special parameters. To this parameters concern width, height, length, radius, density, curvature etc. To produce different kind of products for the customers it is necessary to set these parameters to certain values, in on-line regime. The choosing these parameters for different kind of products manually needs certain time, which decrease the affectivity of the productions. Taking into account the special characteristics of the operators: such as tiredness and intensive working regime the accurate setting those parameters in some case.

To increase the quality output products and affectivity of the productions in the work. The development FMS is considered (Figure 5.2). The structure of flexible manufacturing system for expanded polystyrene board productions is given. The primary material in column 1 and 2 in 180 °C is formed to (secondary material) styropor; this process takes 20 minutes. These expanded polystyrene board products by manipulators 4 and 3 are loaded to the conveyer 5. These primary products by conveyor are transported to the drying-room. Manipulator 6 reload expanded polystyrene board and carry it to the room for drying. After another dried product by the same manipulator is loaded to the conveyor and is fringing for horizontal, vertical and circular cutting in 7, 8 and 9. It is needed to note that the all cutting processes are not necessary. In more case only the horizontal and vertical cutting depends on form of output product. The quality of the processed output expanded polystyrene board product is controlled by manipulator 10. Input parameters for defining quality are: density and curvature of the output products. As an output we may have three types of products. Normal products, the products which need secondary processing, and defects; they are selected to the tanks 11, 12 and 13 correspondingly.

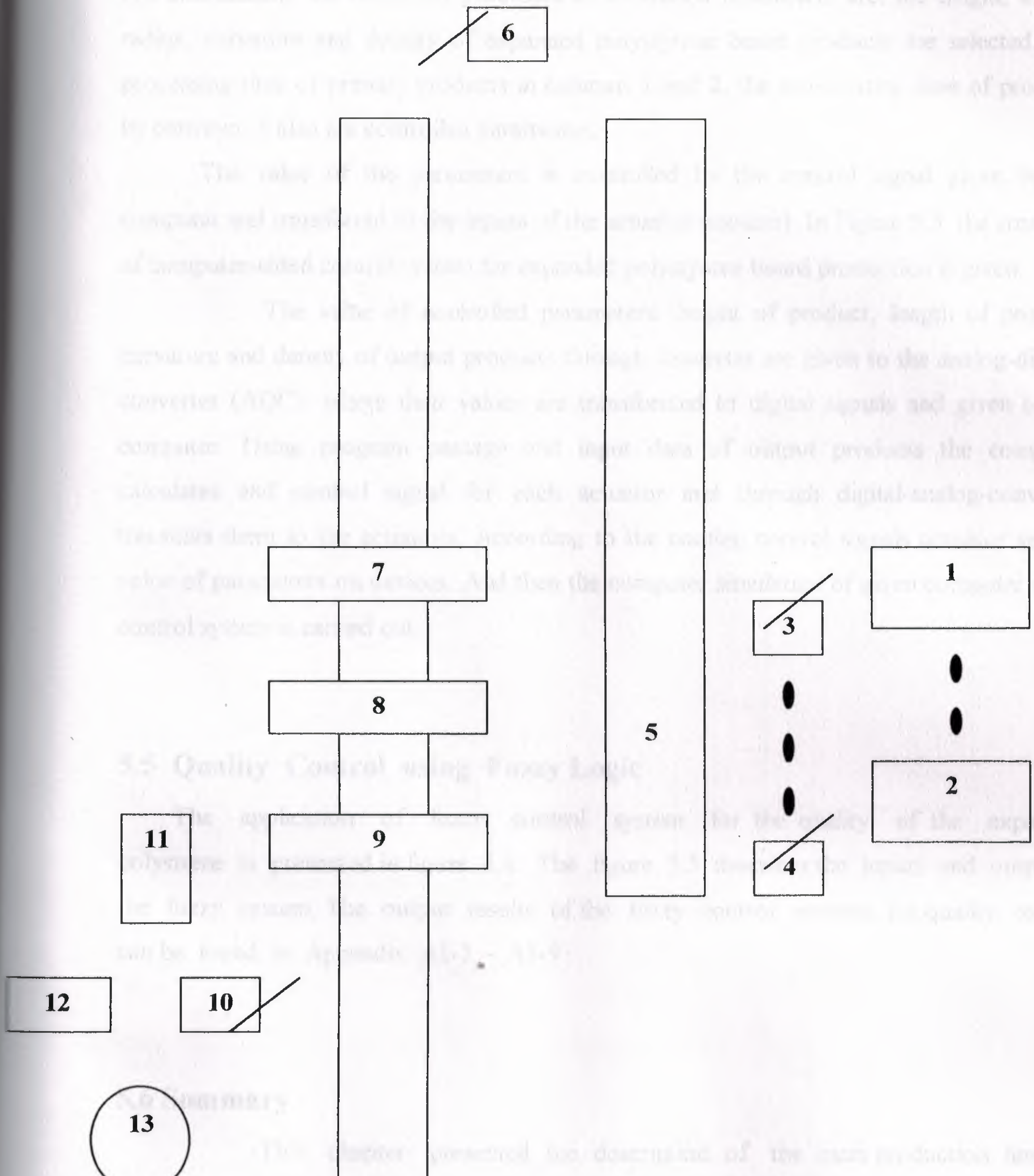


Figure 5.2. The structure of the technological processes of styropor production

For atomization, the described processes as controlled parameters are: the height, length, radius, curvature and density of expanded polystyrene board products are selected. The processing time of primary products in columns 1 and 2, the transporting time of products by conveyor 5 also are controlled parameters.

The value of the parameters is controlled by the control signal given by the computer and transferred to the inputs of the actuator (motors). In Figure 5.3 the structure of computer-aided control system for expanded polystyrene board production is given.

The value of controlled parameters: height of product, length of product, curvature and density of output products through converter are given to the analog-digital-converter (ADC); where their values are transformed to digital signals and given to the computer. Using program passage and input data of output products the computer calculates and control signal for each actuator and through digital-analog-converter transmits them to the actuators. According to the coming control signals actuator set the value of parameters on devices. And then the computer simulation of given computer aided control system is carried out.

5.5 Quality Control using Fuzzy Logic

The application of fuzzy control system for the quality of the expanded polystrene is presented in figure 5.4. The figure 5.5 describes the inputs and output of the fuzzy system. The output results of the fuzzy control systems for quality control can be found in Appendix A1-1 – A1-9

5.6 Summary

This chapter presented the description of the main production lines in the expanded polystyrene foams production , the structure of this production and the application of this product. In addition , the application of fuzzy control system for quality control has been presented.

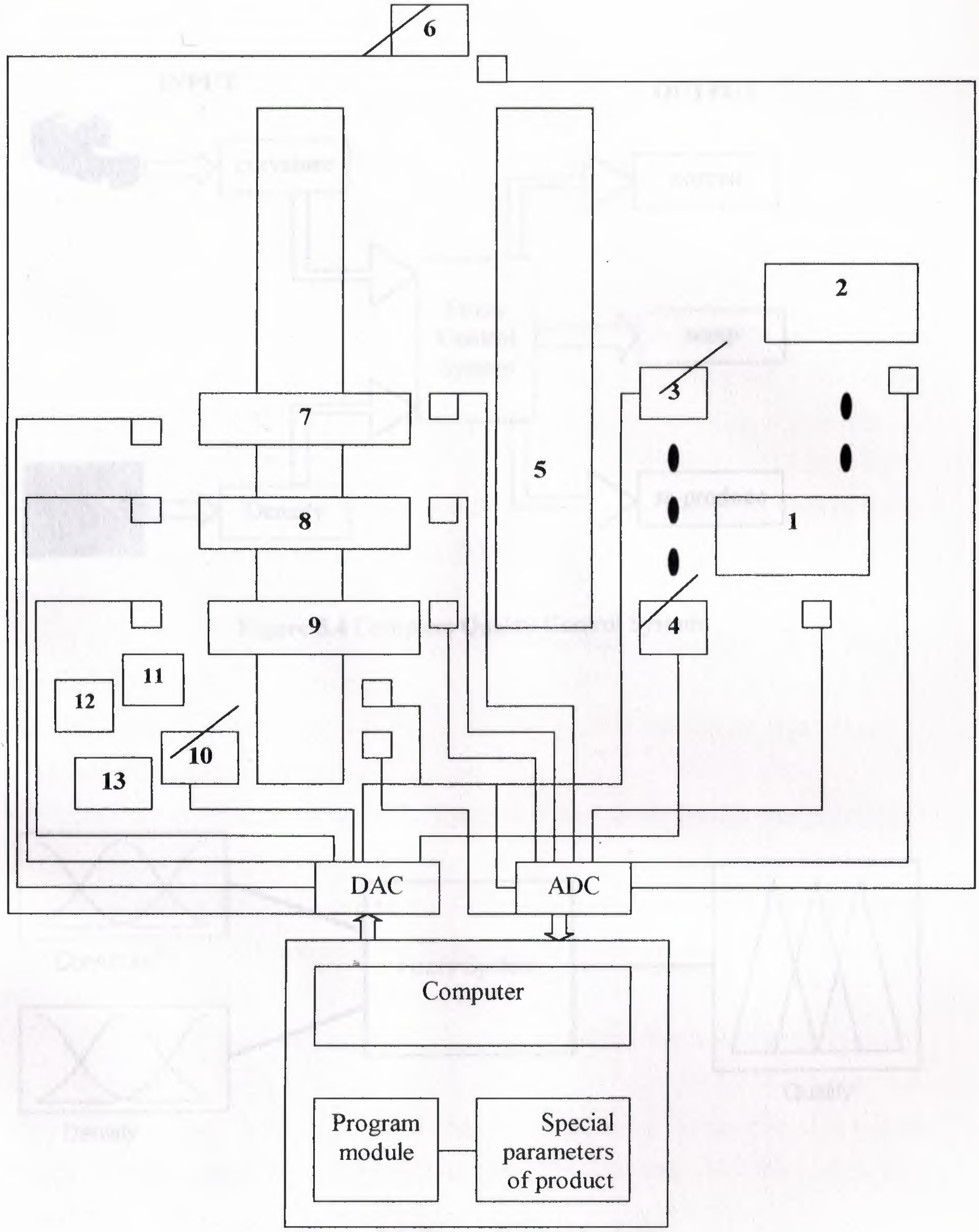


Figure 5.3. The structure of computer-aided control system for styropor productions.

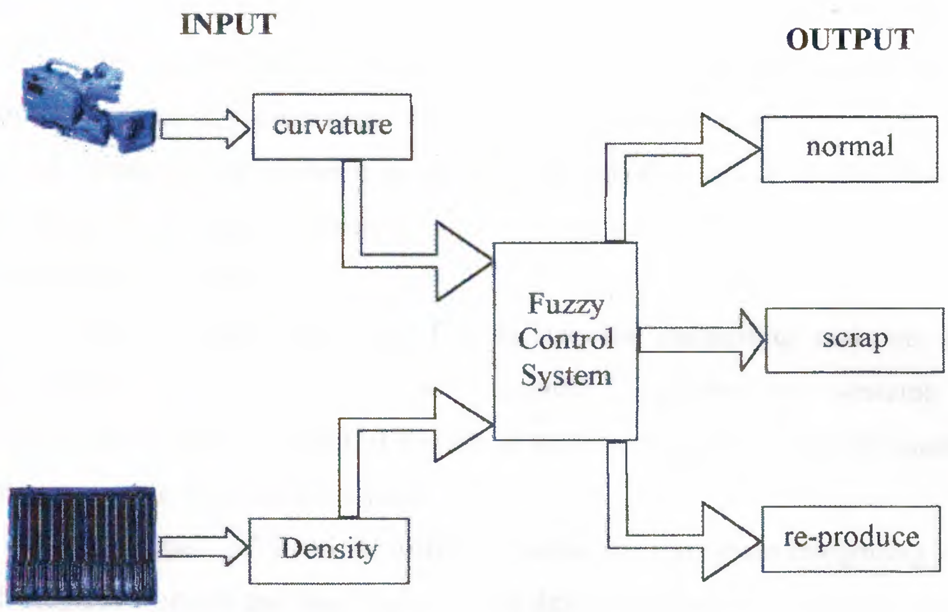


Figure 5.4 Complete Quality Control System

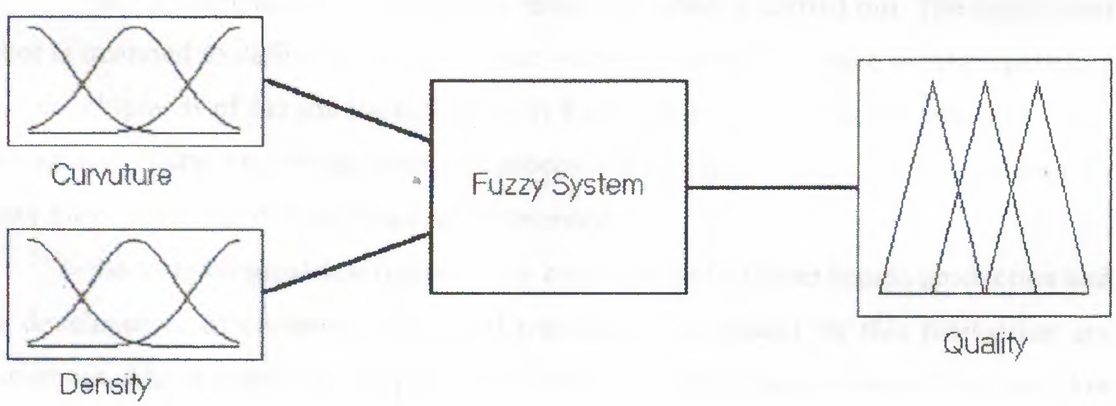


Figure 5.5 Fuzzy Control System

CONCLUSION

Some of the industrial processes are characterized by complicated operational processes, different type output products. Real time planning of these processes requires special hardware and software recourses. The development FMS for these productions including automated intellectual workstations allows to increase the efficiency of productions. In the thesis:

- the role and functions of computers for controlling real time processes in FMS and evolution of computer controlled manufacturing systems are given. Supervisory computer control of flexible manufacturing system and its functions, types of software used in FMS are described.

- the types of FMS for different productions, its main integrating hardware and software components and their functions are described. Software and hardware structure of FMS, their control functions are presented.

- the scheduling and control problems of FMS are described. The four scheduling algorithms for single-product scheduling problem, N-product scheduling problem, single-batch scheduling problem and N-batch scheduling problem are given. The graphical representations of these problems are presented.

- the development of intellectual robot for FMS is carried out. The intellectual robot is intended to define the quality of the output products and make sorting operation. The development of the intellectual robot is based on fuzzy knowledge base and fuzzy technology. Using knowledge base the processing of input information on the base of fuzzy max-min composition principle is presented.

- the technological description of the expanded polystyrene boards production and the development of computer controlled manufacturing system for this production are carried out. The structure of computer controlled manufacturing system and its operation principle for this production has been presented. The computer simulation of cutting process of expanded polystyrene boards has been carried out on IBM PC.

REFERENCES

[1] Fundamental of Modern Manufacturing , Materials, Processes,and Systems. Mikell P. Groover. Prentice Hall International Editions.

[2] Koren, Y (1983) In " Computer Control of Manufacturing Systems" Mc Graw – Hill New York

[3] Williams ; D. J (1988) In ''Manufacturing Systems '' Open University Press, Milton Kaynes, Englang.

[4] Akella R. Choong, Y., and Gershwin, S. (1984) IEEE Trans. Components, Hybirds, Manuf. Technol. CHMT-7

[5] Morton T. E ,and Smut,T. L. (1986). In '' Flexible Manufacturing Systems : Methods and Studies''

[6] Smits, M. ,Ramesh, R. dudek , R., and Blair, E.(1986) . In ''Proceeding of the Second ORSA/TIMS Conference on Flexible Manufacturing Systems.

[7] Fox K. (1982). Soc. Manuf. Eng.- Autofact 4 Conf.

[8] Rembold , V., Blume, C., a
nd Billmann, R. (1985). In '2 Computer-Integrated Manufacturing Technology and systems.

[9] Kusiak, A. (1987) Artificial intelligence and operations research in flexible manufacturing systems

[10] Niebel,B. W., Draper, A.B., and Wysk, R.A.(1989).Modern Manufacturing Process Engineering. McGraw –Hill, New York.

[11] Groover. M.P. Automation Production systems and Computer Integrated Manufacturing Prentice Hall. Englewood Cliffs.

[12] L.A Zadeh, Fuzzy Sets, Information and Control , vol. 8,338-353, 1965

[13]Fuzzy Process Control and Knowledge Engineering in Petrochemical and Robotic Manufacturing. Rafik Aliev, Fuad Aliev,and Mamedgasan Babaev. Verlag TÜV Rheinland Köln 1991.

[14] L. A. Zadeh, Fuzzy set, Theoretic International & Linguistic Hedges, Cybernetics vol.12,4-34.1972

[15] Passino, K. M. and Yurkovich, S. (1998). *Fuzzy Control*, Addison Wesley Longman, Inc, Menlo Park, CA, USA

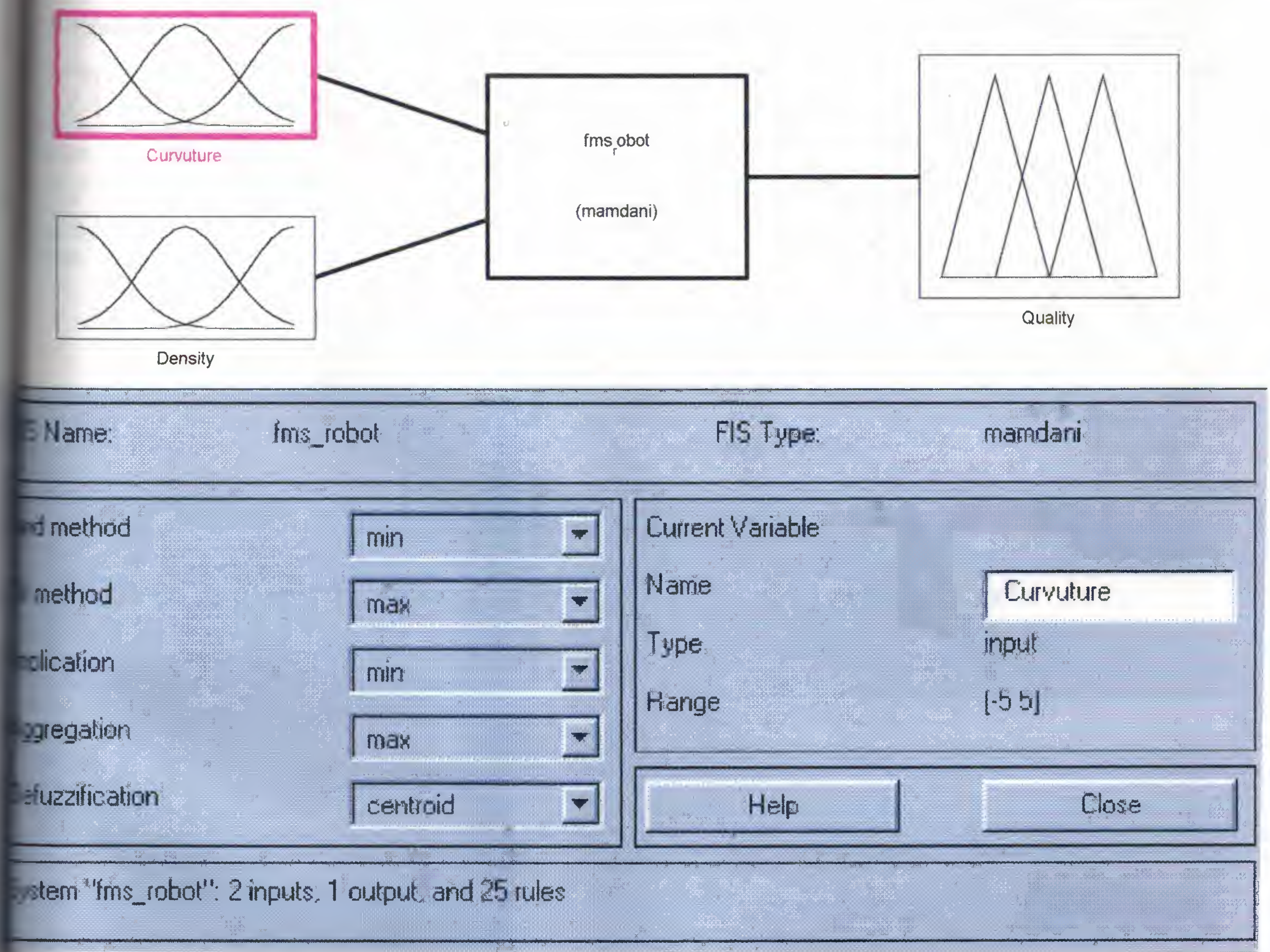
[16] Rolston,L.J. (1984). Modeling Flexible Manufacturing systems with MAP/1 Proc ORSA/TIMS Spec. Interest conf. Flexible Manuf. Syst. 1 st, 1984, p. 199-204

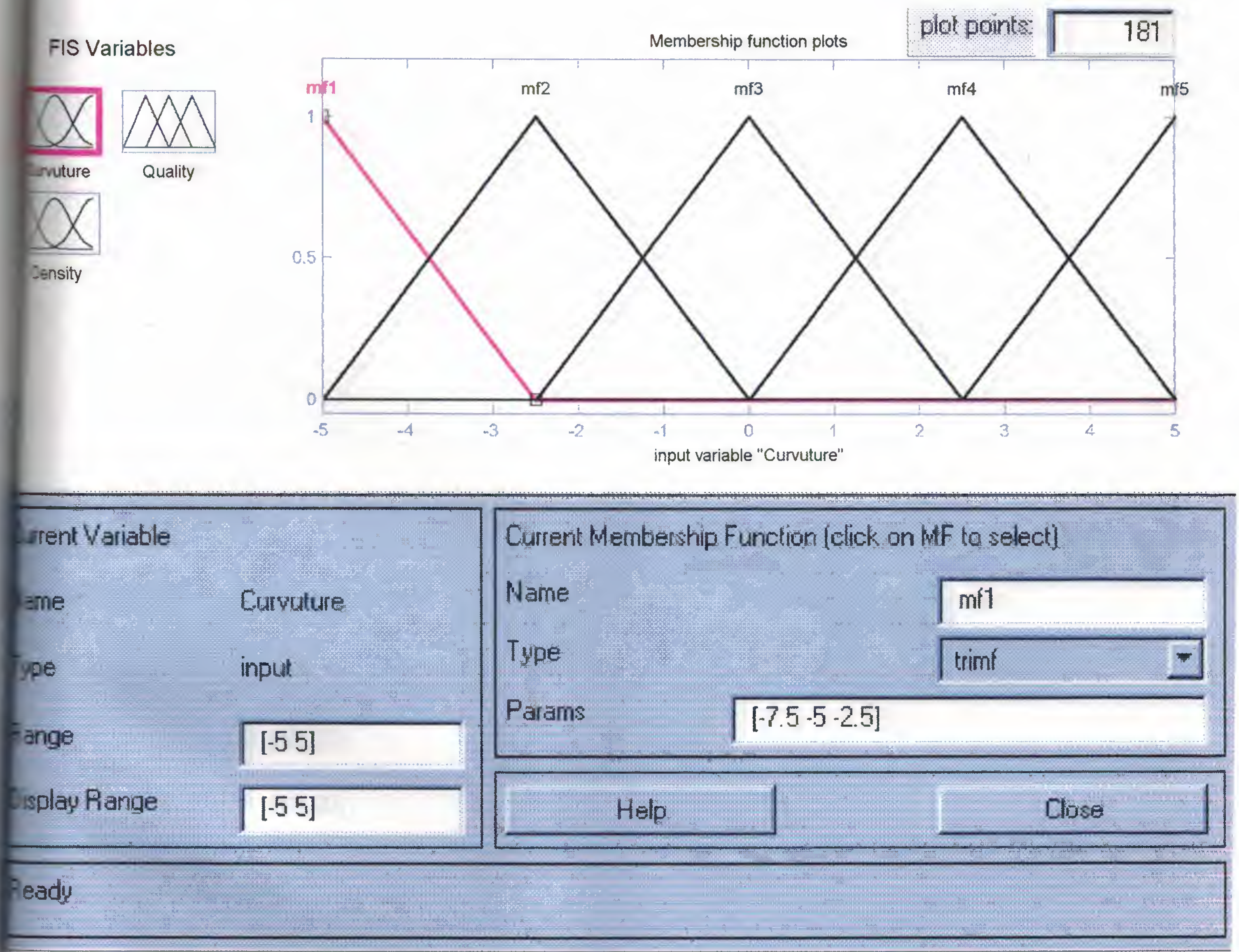
[17] Garett, L , and Mc Henry, J.M (1981) Hydrocarbon processs 60 (12)

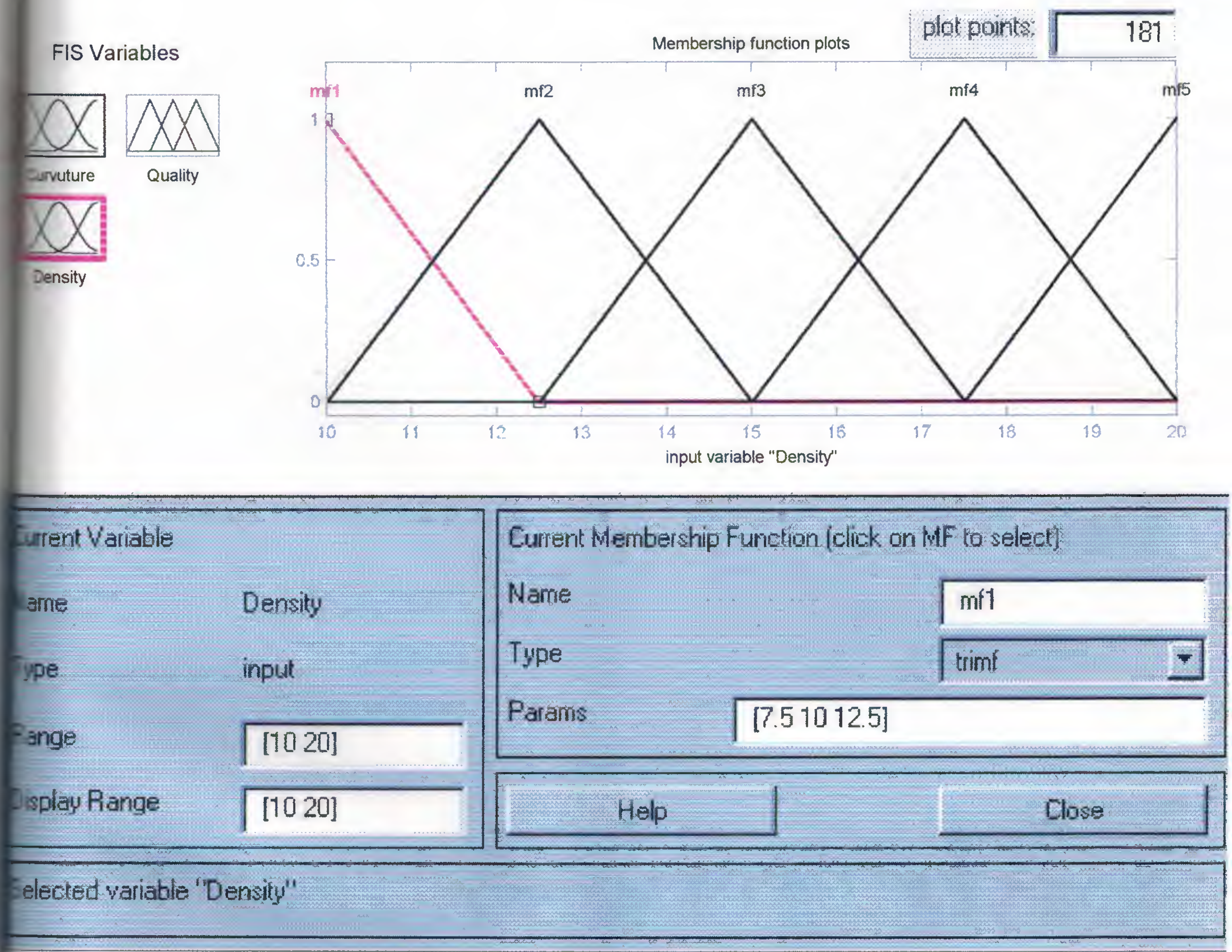
[18] R.H.Abiyev, K.M.Kerimov, T.G.Melikov. Fuzzy system for determination of oil-gas fields productivity/ Geophysics News in Azerbaijan. Baku,N:1,1999.pp.13-15

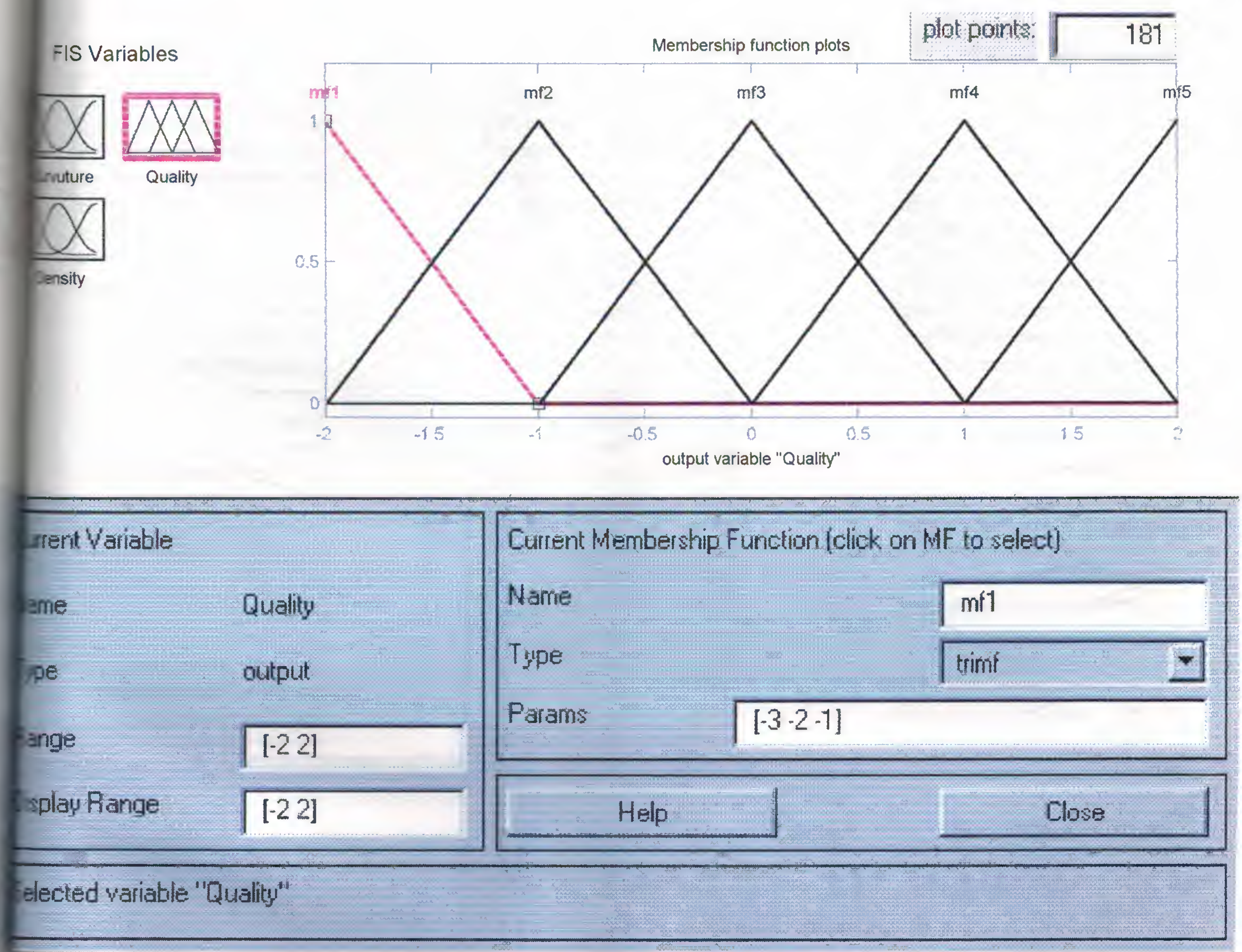
[19] Wu, D.S , and R.A.Wysk. 1989. An application of Discrete-Event Simulation to On – Line Control and Scheduling in Flexible Manufacturing . International Journal of Production Research, Vol. 27 , 1603 – 1624.

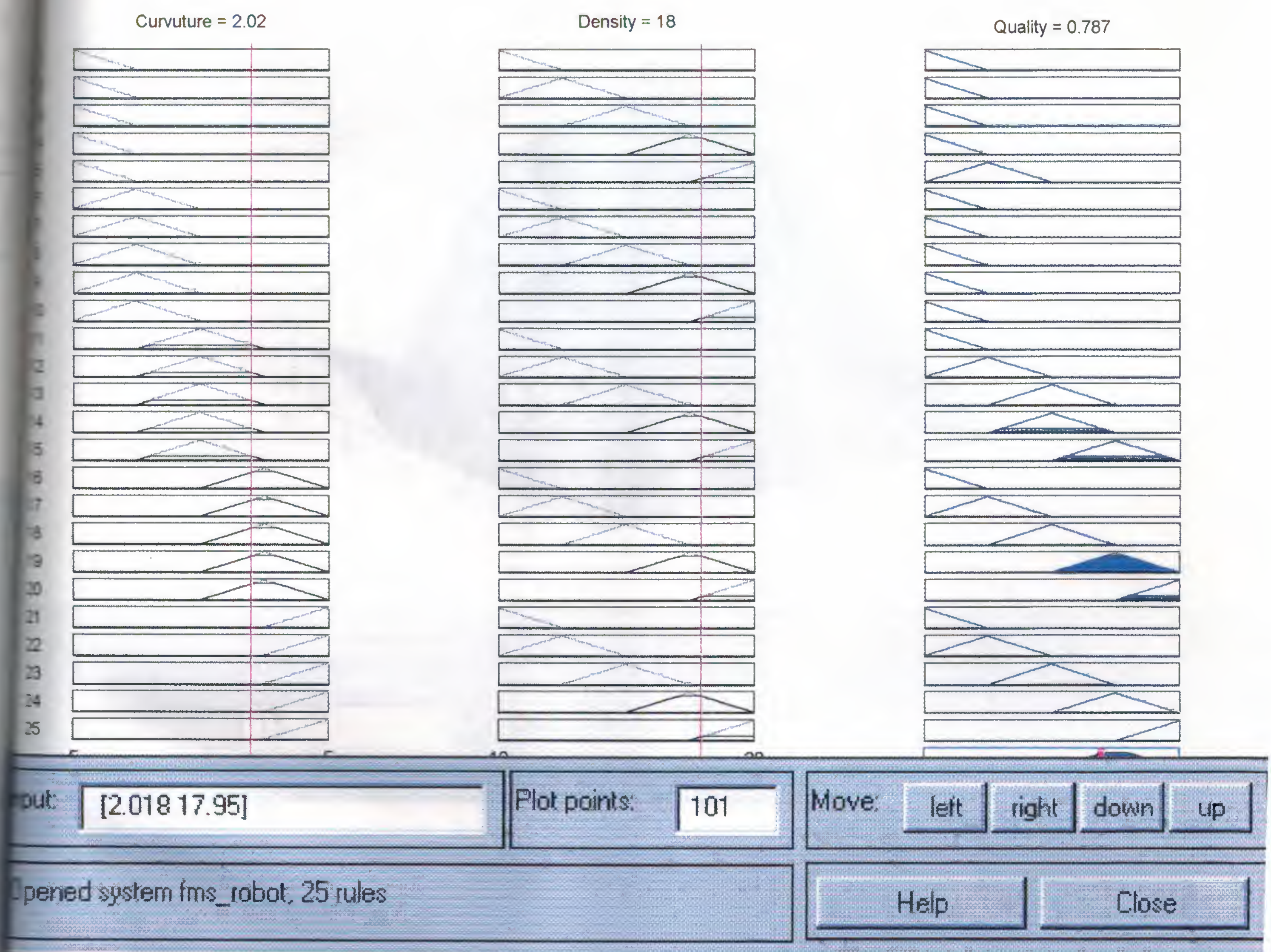
APPENDIX 1


Curvuture
Density
fms_robot
(mamdani)
Quality
Name:
fms_robot
FIS Type:
mamdani
d method
min
method
max
plication
min
ggregation
max
efuzzification
centroid
Current Variable
Name
Curvuture
Type
input
Range
[-5 5]
Help
Close
ystem "fms_robot": 2 inputs, 1 output, and 25 rules

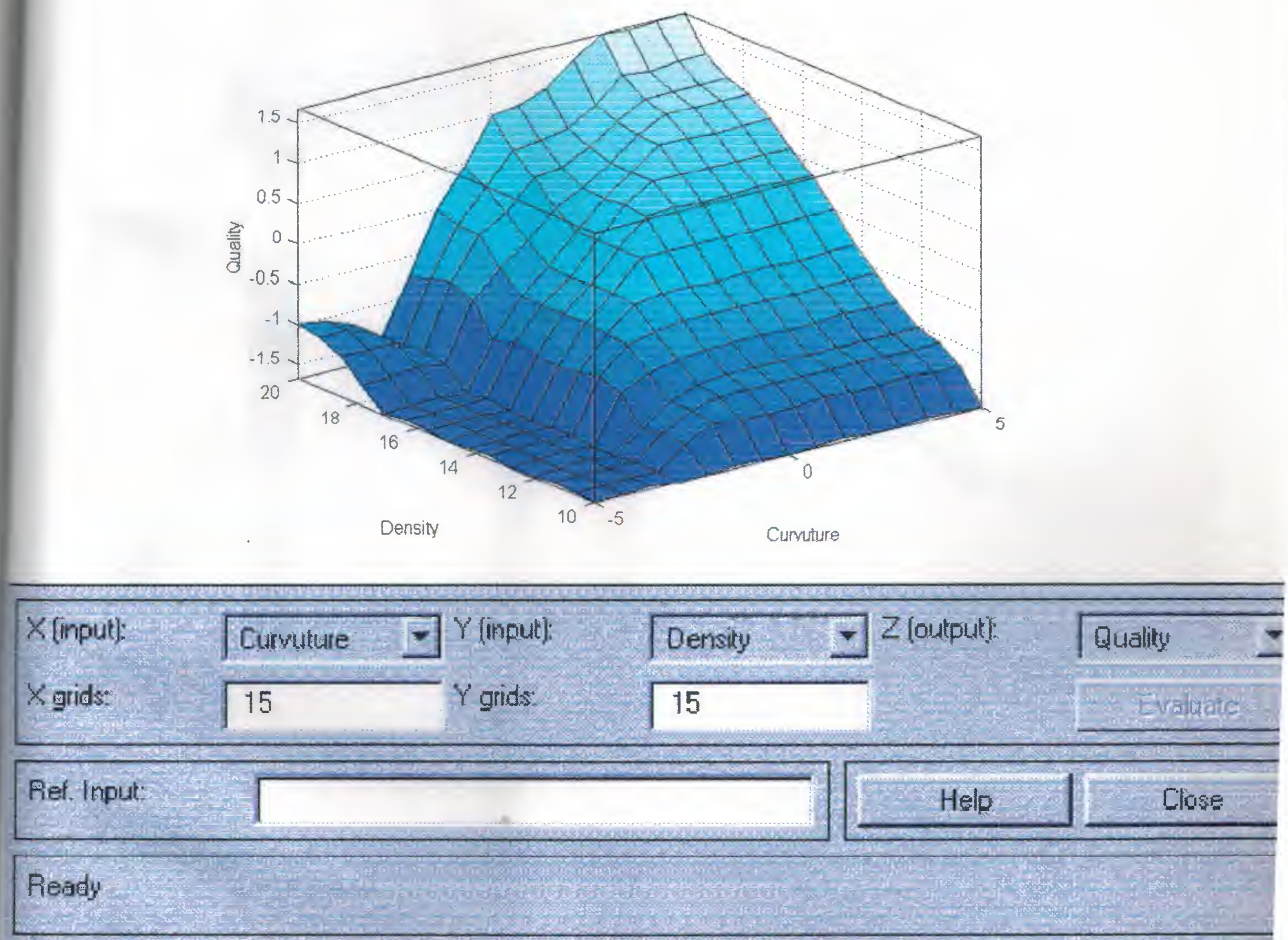

FIS Variables
Curvature
Quality
Density
Membership function plots
plot points:
181
mf1
mf2
mf3
mf4
mf5
1
0.5
0
-5
-4
-3
-2
-1
0
1
2
3
4
5
input variable "Curvuture"
Current Variable
Name
Curvuture
Type
input
Range
[-5 5]
Display Range
[-5 5]
Current Membership Function (click on MF to select)
Name
mf1
Type
trimf
Params
[-7.5 -5 -2.5]
Help
Close
Ready


FIS Variables
Curvuture
Quality
Density
Membership function plots
plot points:
181
mf1
mf2
mf3
mf4
mf5
1
0.5
0
10
11
12
13
14
15
16
17
18
19
20
input variable "Density"
Current Variable
Name
Density
Type
input
Range
[10 20]
Display Range
[10 20]
Current Membership Function (click on MF to select)
Name
mf1
Type
trimf
Params
[7.5 10 12.5]
Help
Close
Selected variable "Density"


FIS Variables
Quality
Membership function plots
plot points:
181
mf1
mf2
mf3
mf4
mf5
1
0.5
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
output variable "Quality"
Quality
output
[-2 2]
[-2 2]
Current Membership Function (click on MF to select)
Name
mf1
Type
trimf
Params
[-3 -2 -1]
Help
Close



Curvuture = 2.02
Density = 18
Quality = 0.787
Input: [2.018 17.95]
Plot points: 101
Move: left right down up
Opened system fms_robot, 25 rules
Help
Close


Quality
1.5
1
0.5
0
-0.5
-1
-1.5
20
18
16
14
12
10
-5
0
5
Density
Curvuture
X (input):
Curvuture
Y (input):
Density
Z (output):
Quality
X grids:
15
Y grids:
15
Ref. Input:
Help
Close
Ready

APPENDIX 2