CHAPTER 2
A REVIEW ON FMS SCHEDULING

2.1 INTRODUCTION

Scheduling covers a wide range of problems and concerns all industrial systems. The characteristics of scheduling problems are various and numerous and their modelling is a difficult task. The hardware and scheduling situations of the system influence the scheduling problems. The hardware of a system is represented by machine environment, part (jobs) attributes such as due dates, release dates (arrivals) processing times, precedence constraints, tool constraints and the like. The objective criterion describes the scheduling environment. Various types of scheduling problems have been solved in different FMS environment. Varieties of algorithms have been employed to get optimal or near optimal schedules. Also the scheduling schemes are classified by different dimensions based on the responsiveness of a scheduling methodology that is either a real-time or an off-line or an off-line reschedule. It is the purpose of this chapter to identify the models, the algorithms and the schemes that are needed for scheduling FMS. This chapter concentrates on the scheduling problems, practices and operational issues in FMS. The rest of the chapter is organised around the following four topical areas.

An overview of FMS : Addresses the definitions, the basic types, and the problems encountered with design and operation of FMS

A scheduling framework : Details the different types of scheduling problems, algorithms, and schemes.

FMS scheduling scenario : Presents the literature that deals with the FMS scheduling problems.

Summary : Describes the status of scheduling practice.

2.2 AN OVERVIEW OF FMS

This section addresses the definitions, the basic types, and the problems encountered with design and operation of FMS

2.2.1 Definition

A variety of definitions is given for FMS and a few of them are addressed here:
In FMS, the material handling system permits jobs to follow diverse routings. This can be achieved by a recirculating conveyor linking all stations, by carts, an automatic wire-guided system or by a work transporter (Buzacott 1983).

FMS refers to the integration of several automated work cells, capable of producing a variety of parts. The cells are served by automated material handling equipment, robots, communication system, and a supervisory computer. Each cell typically consists of a universal machine tool equipped with automatic tool changing and material handling (Badiru 1990).

FMS is an integrated, computer-controlled complex automated material handling device and NC/CNC/DNC machine tools that are applicable typically to mid-volume, mid-variety range of production (Dhar 1991).

FMS is an integrated set of computer numerically-controlled machine tools and automatic material handling devices, all controlled by a central computer (Shanker and Agarwal 1991)

### 2.2.2 Types of FMS

FMS are a broad collection of production systems that can take several different structural forms. They are broadly classified into dedicated and random types (Groover 1994). On the basis of the number of computer numerically-controlled machines and their arrangement, FMS are generally divided into five classes as indicated in TABLE 2.1 (Kusiak 1985).

<table>
<thead>
<tr>
<th>Type</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flexible Manufacturing Cell (FMC)</td>
<td>An NC/CNC machine augmented by a part buffer, a tool changer, a pallet changer, etc.</td>
</tr>
<tr>
<td>Flexible Manufacturing System (FMS)</td>
<td>A group of several FMMs that are built according to product or process type.</td>
</tr>
<tr>
<td>Flexible Manufacturing Group (FMG)</td>
<td>A collection of FMCs and FMMs that looks after a specific manufacturing function (machining or assembly or inspection, etc.) and joined by a material handling system.</td>
</tr>
<tr>
<td>Flexible Production System (FPS)</td>
<td>Different types of FMGs connected by a common material handling system.</td>
</tr>
<tr>
<td>Flexible Manufacturing Line (FML)</td>
<td>A set of dedicated machine tools or processing stations arranged according to product type and integrated with an automated material handling system.</td>
</tr>
</tbody>
</table>
2.2.3 FMS problems


At the planning or design stage, the following decisions are required:

- Determination of the type of products that could be produced (Product range problem).
- Selection of a proper process (Process planning problem).
- Selection of type of FMS (System configuration problem).
- Selection of a material transport system (Transport problem).
- Determination of type and number of machines (Machine capacity problem).
- Determination of type and number of fixtures and pallets (Fixturing/Pallet problem).
- Selection of an appropriate computer system (Control software problem).
- Layout and integration of all the above systems (Locational problems).

The operational problems are:

- Selection of parts for a planning horizon (Part-type selection problem).
- Grouping of machines for a set of parts (Machine grouping problem).
- Determination of production ratios of products (Production ratio problem).
- Allocation of material handling equipment, fixtures and pallets (Resources allocation problem).
- Allocation of operations and tools to machines (Machine loading problem).
- Assignment of operations over a span of time (Scheduling problem).
- Determining the operation and control policies for the material handling systems (MHS scheduling problem).

2.3 SCHEDULING FRAMEWORK

This section presents the details of the different types of scheduling problems, algorithms, and schemes.

2.3.1 Scheduling problems

The theory of scheduling is characterised by a virtually unlimited number of problem types. Generally, they are classified into three categories as $\alpha/\beta/\gamma$ where $\alpha$ specifies the
Scheduling FMS Using Heuristic and Search Techniques

machine or system environment, \( \beta \) specifies the job characteristics, and \( \gamma \) denotes the optimality criterion (Bruker 1995, Artiba et al. 1997, Portmann 1997).

The machine (resource) environment is characterised by a set of two parameters. The first parameter is the specification of the type of machines (Dedicated, identical parallel, uniform parallel and unrelated parallel). The second parameter dictates the predominant flow pattern of the jobs which is determined based on the operation precedence relations (General shop, job shop, flow shop, open shop and mixed job). The definitions are given in TABLE 2.2.

<table>
<thead>
<tr>
<th>TYPE</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dedicated</td>
<td>Each job needs processing only on a specified dedicated machines</td>
</tr>
<tr>
<td>Identical parallel</td>
<td>Each job can be processed on any one of the parallel machines, identical in</td>
</tr>
<tr>
<td></td>
<td>nature</td>
</tr>
<tr>
<td>Uniform parallel</td>
<td>Each job can be processed on anyone of the parallel machines having the</td>
</tr>
<tr>
<td></td>
<td>same operational capabilities and operating with different speeds,</td>
</tr>
<tr>
<td></td>
<td>independent of jobs</td>
</tr>
<tr>
<td>Unrelated parallel</td>
<td>Each job can be processed on any one of the parallel machines whose</td>
</tr>
<tr>
<td></td>
<td>operational capabilities are different for each job. (i.e. Operates with</td>
</tr>
<tr>
<td></td>
<td>speeds dependent on job)</td>
</tr>
<tr>
<td>General shop</td>
<td>It is a configuration that represents any one or combination of job, flow,</td>
</tr>
<tr>
<td></td>
<td>open and mixed shops.</td>
</tr>
<tr>
<td>Job shop</td>
<td>Each job requires different set of machines with specific precedence</td>
</tr>
<tr>
<td></td>
<td>ordering of the operations. The resources are shared by a variety of jobs</td>
</tr>
<tr>
<td></td>
<td>at the same time.</td>
</tr>
<tr>
<td>Flow shop</td>
<td>Each job is to be processed on the same set of resources with an identical</td>
</tr>
<tr>
<td></td>
<td>precedence ordering of the operations. The flow shop assumes that all</td>
</tr>
<tr>
<td></td>
<td>parts of the same kind are moving in one direction only.</td>
</tr>
<tr>
<td>Mixed shop</td>
<td>It is a combination of flow and job shop.</td>
</tr>
<tr>
<td>Open shop</td>
<td>The open shop is defined as the flow shop with the exception that there</td>
</tr>
<tr>
<td></td>
<td>are no precedence constraints/relations between the operations</td>
</tr>
</tbody>
</table>

The job characteristics are generally specified with the following parameters: Preemption condition, Precedence relations, Processing time, Number of processors required, Release dates and Due dates. The explanations to the above are given in TABLE 2.3.
### TABLE 2.3 Definition of job descriptors

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-emption condition</td>
<td>Pre-emption of a job or operation means that processing may be interrupted and resumed later, even on other machine. With pre-emption, a job or operation may be interrupted several times. If job or operation splitting is not allowed, then it is the case of without pre-emption.</td>
</tr>
<tr>
<td>Precedence relationship</td>
<td>If the jobs or the operations of a job are to be processed strictly in a specific sequence or order, then the jobs or operations are constrained with precedence relationship.</td>
</tr>
<tr>
<td>Processing time</td>
<td>The processing times of a job or operation may either be deterministic (predictable exactly before the production commences) or stochastic (vary during production run).</td>
</tr>
<tr>
<td>Number of processors</td>
<td>Depending upon the operation requirements, the number of processors may be single or multiple.</td>
</tr>
<tr>
<td>Release dates</td>
<td>It specifies the jobs arrival pattern, whether they are all available at the starting of the scheduling period or arrive in future.</td>
</tr>
<tr>
<td>Due dates</td>
<td>It indicates the time at which the jobs are to be delivered, if there are any deadlines associated with them.</td>
</tr>
</tbody>
</table>

The optimality criterion falls into any one of the two types: makespan time and due date related (Mellor 1966). The general specification methodology is given in TABLE 2.4

### TABLE 2.4 Scheduling objective criteria

<table>
<thead>
<tr>
<th>Type objectives</th>
<th>Performance criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan minimisation</td>
<td>The following criteria are considered for this class: 1. Finish the last job as soon as possible 2. Finish each job as soon as possible 3. Minimise the in-process inventory costs 4. Maximise machine utilisation</td>
</tr>
<tr>
<td>Due date based cost minimisation</td>
<td>Typical criteria cited for this class are: 1. Minimise the number of late jobs 2. Minimise the total tardiness 3. Minimise the costs due to not meeting due dates exactly 4. Minimise the maximum lateness of any job</td>
</tr>
</tbody>
</table>
2.3.2 Computational complexity

In general, scheduling problems belong to a broader class of combinatorial search problems. A combinatorial search problem $\mathcal{P}$ is a set of pairs $(I,A)$, where $I$ is an instance of a problem with specified values, and $A$ is an answer to the instant (Blazewize et al 1993, Bruker 1995). Complexity theory provides a mathematical framework in which computational problems are studied so that they can be classified as 'easy' and 'hard'. One of the main issues of complexity theory is to measure the performance of algorithms with respect to computational time (time complexity function is given as: $O(p(k))$, where $p$ is some polynomial and $k$ is the input length of an instant). The scheduling problems concerned are generally NP-hard, that is, it is probably impossible to secure optimal solutions using fast algorithms (i.e. algorithms that run in polynomial time in the size of the problem), though some problems are in fact 'easy', in the sense that they are solvable to optimality by fast algorithms (Portmann 1997).

2.3.3 Scheduling algorithms

The automatic generation of scheduling plans for job shops are traditionally addressed using optimisation and approximation approaches (Blazewize 1993, Bruker 1995). Optimisation algorithms include enumerative procedure and mathematical programming techniques (Linear, Integer programming, Goal programming, Transportation, Network and Dynamic programming). The techniques included in approximation techniques are: Implicit enumeration (B-B); Decomposition (LR); Priority rule based; Heuristics. Local search algorithms (ITS, SA, TA, TS); Evolutionary Programs (GA); AI techniques (KB and ES); A classification framework based on the above is formulated and is given in TABLE 2.5.
### TABLE 2.5 Classification of scheduling algorithms

<table>
<thead>
<tr>
<th>Class</th>
<th>Algorithm</th>
<th>Definition/Application Area</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Optimization</strong></td>
<td>Enumerative</td>
<td>Methodology is based on enumerating all combinations, sorting the feasible set and selecting the best solution from the feasible set. This can be applied to problems that are smaller in size since the solution space grows exponentially with problem size.</td>
</tr>
<tr>
<td></td>
<td>Mathematical programs</td>
<td>Mathematical programs are the formulation with a set of equations (linear as well as non-linear) that represent the constraints and the objective criteria. The difficulty with formulation and the need of huge computation effort restrict this approach to large and complex problems.</td>
</tr>
<tr>
<td><strong>Approximation</strong></td>
<td>Branch-Bound</td>
<td>It is based on the idea of intelligently enumerating the feasible solutions with lower and upper bounds.</td>
</tr>
<tr>
<td></td>
<td>Lagrangian Relaxation</td>
<td>The huge mathematical models are decomposed into smaller problems and the solutions are constructed from the sub problems.</td>
</tr>
<tr>
<td></td>
<td>Priority rules</td>
<td>These are simple sequencing rules that specify the queue discipline. The performance of dispatching rules is problem-dependent.</td>
</tr>
<tr>
<td></td>
<td>Heuristics</td>
<td>These algorithms rely on rules of thumb. Any approach without formal guarantee of performance can be considered 'heuristic'. They are considered useful in practical situations when no better methods are available.</td>
</tr>
<tr>
<td></td>
<td>Local search</td>
<td>These algorithms use the concept of neighbourhood search that search for better solutions within their neighbours and move towards optimal. They provide solutions that are not optimal since they converge at their local optima.</td>
</tr>
<tr>
<td></td>
<td>Evolution programs</td>
<td>Evolutionary algorithm (GA) is based on the recognition that evolution, with its principles of mutation and selection, represents an efficient process for solving hard optimisation problems.</td>
</tr>
<tr>
<td></td>
<td>AI programs</td>
<td>These approaches belong to intelligent decision category search that provides an answer to an instance of a decision problem which may take only two values, either 'yes' or 'no'.</td>
</tr>
</tbody>
</table>

### 2.3.4 Scheduling schemes

The FMS scheduling is normally carried out with a real-time or an off-line or an off-line reschedule scheme. Real-time scheduling is an integrated approach in which the decision to assign a job on a particular facility is arrived upon the occurrence of any one of the following events: new job arrival, completion of one operation or job, resource failure, blocking, congestion, and so on. The decision is taken based on many system attributes: resource requirements and their availability, criticality of jobs, objective function, reliability of the components of the system and so on. In an off-line method, scheduling algorithms and
mathematical programs are used to determine the best schedule, the best schedule is then executed strictly regardless of the events occurring after its formation. This is a finite capacity detailed scheduling approach in which the allocations of resources over a span of time (planning horizon) for each activity are planned in advance and a new schedule would not be constructed before the complete execution of the current schedule. The off-line rescheduling is a mix of these two modes and so sometimes called 'hybrid'. In off-line reschedule, a new schedule is generated upon the occurrence of some pre-specified real time events. Further, the off-line rescheduling schemes can be classified into conventional or regeneration and use or edit methods based on the concept of rescheduling. A classification of scheduling schemes has been made by this dimension and given in figure 2.1.

2.4 LITERATURE SURVEY ON FMS SCHEDULING

There are number of approaches and procedures in the FMS scheduling literature. In this section, the literature that addresses the scheduling problems in flexible manufacturing context is reviewed.

2.4.1 Optimisation Approaches

2.4.1.1 Enumerative techniques

Giffler and Thompson (1960) developed an enumerative procedure to generate all active schedules for the general 'n' job 'm' machine problem. They indicated that it is practical, in problems of small size, to enumerate all active schedules and pick the optimal schedule.
They suggested a random generation of active schedules and priority rules for large problems. They illustrated their methodology with a problem of size 3 x 3 with makespan objective criterion. They reported that a problem with 2000 operations took about 20 minutes with an IBM 704 to select 200 random schedules. It indicates that enumerating all feasible combinations is a time-consuming process and is not practicable when the size of the problem (number of jobs and/or processors) increases.

Chang and Sullivan (1990) addressed a scheduling problem of a dynamic job shop (or FMS), where each job incurs a significant transfer time in going between workstations and a particular operation can be performed by different workstations with possibly different operation times on several groups of similar or parallel workstations. They extended Giffler and Thomson's (GT) algorithm to generate all active feasible schedules and pick the optimal schedule. This algorithm needs to create an extremely large number of active schedules, even for small problems, and as a consequence requires excessive computation time and memory. In this context, they developed an alternative method ‘reduced enumerating technique’ that partitions the operations or jobs into small sub-sets using the concept of group technology, finds optimal schedules for the subsets and combines the optimal schedules of the subsets to provide the entire schedule. They showed that the schedules provided by the reduced enumeration technique for the average total flow time minimisation are close to those with extended GT algorithm. Further, it is indicated that it takes less computation time. However, the problems experimented are smaller and static.

2.4.1.2 Mathematical Programming Techniques (Linear, Integer, Transportation, Network, Goal and Dynamic programming)

Shanker and Tzen (1984) considered scheduling problem in a random FMS as a composite of two interdependent tasks: loading and sequencing. They formulated this problem as mixed integer program (MIP) with two objectives: i) minimisation of system workload unbalance, and ii) minimisation of system workload unbalance and the number of late jobs; under the following constraints: limited tool slots, unique job routing, non-splitting of jobs and limited machine capacity. The MIP formulations, although solvable, are in general too large to be computationally feasible.

Lee and Jung (1989) presented a goal programming model for production planning problem in FMS with multiple conflicting objectives that need to achieve the required production level, balancing workload among machines and minimisation of throughput time,
under the constraints of machine capacity and fixture availability. A limitation of this model is that information on the dynamic working of FMS can not be provided.

Ghosh and Gaimon (1992) addressed a production planning problem that links Material Requirement Planning (MRP) and scheduling in a FMS. They framed a mathematical model that takes the planned orders in the MRP as the input to solve the following decisions: i) the division of weekly production requirements to daily production requirements and the daily backorder levels, ii) the determination of production quotas for each operation of each part type, and iii) the assignment of these daily production quotas each batch to machine groups, given the flexibility offered by alternate routings. They analysed the model to determine the impact of shop flexibility on total cost, inventory levels, shop utilisation, and the number of set-ups and spilt lot production. The results of experimentation indicate that increasing the routing flexibility of the system without a parallel decrease on set-up costs and times is unlikely to reap significant benefits.

Chisman (1986) proposed an analytical model that is formulated as a Travelling Salesman Problem (TSP) for minimising total set-up time in flow shop production cell, where set-up time is sequence dependent.

Luh et al. (1996) proposed a backward dynamic programming solution technique to a relaxed job shop scheduling problem that incorporated with several major categories of uncertainties: uncertain arrival times, processing times, due dates and part priorities. They showed through simulation that the schedules generated enjoy low expected tardiness and earliness penalties with low tardiness variances. The CPU time is higher and provides only near optimal solution and further, the closeness to optimum is not established.

2.4.2 Approximation Approaches

2.4.2.1 Branch - Bound techniques (Implicit enumeration)

Greenberg (1968) formulated a mixed integer programming model for the general \( n \) job, \( m \) machine problem. They suggested a B-B technique to reduce the formulation as a series of non-integer LP problems. A lower bound technique is applied to resolve the conflicts in GT algorithm for makespan criterion (King 1975). The computation involved with the lower bound calculations is large that restricts its use for large size problems.

Nascinmento and Mario (1993) showed with CPU time and solution quality that the GT algorithm (1960), originally designed for the traditional job shop model, can still provide good results to FMS model. Instead of resolving the conflicts one by one in all possible
means and enumerating all feasible schedules, two B-B algorithms are structured and their behaviour compared on both job shop and FMS model.

Toker et al. (1994) proposed an approximation algorithm for the n job, m machine resource-constrained job shop problem. The algorithm first finds a feasible schedule for the relaxed problem (i.e. without considering resource that constrains the problem) and then adjusts with minimum amount of right shifts to satisfy the resource requirements. The feasible schedule of the relaxed problem is either an optimal or a heuristic schedule. Further, they introduced two lower bounds to adjust the schedules. They evaluated the performance of the proposed algorithm with relative percentage error which is defined as follows.

\[
\text{Relative percentage error} = \frac{(\text{approximate solution} - \text{optimal solution})}{\text{optimal solution}} \times 100
\]

The procedure of right shifting can be used to integrate the material handling system and the tools in the CIM environment scheduling problems.

2.4.2.2 Lagrangian Relaxation techniques

Hoitomt et al. (1993) explored the use of Lagrangian Relaxation technique to schedule job shops that are characterised with multiple non-identical machine types, generic precedence constraints and simple routing considerations. They developed a discrete time-integer programming model with the objective of minimising the weighted quadratic tardiness function under job precedence constraints, capacity constraints, processing time requirements and routing decisions, and described the lagrangian relaxation framework with a decomposed solution methodology to derive a feasible schedule. Their method attempts to bridge the gap between heuristic and optimisation approaches to obtain a near optimal schedule with quantifiable performance for reasonably practical size problems. Numerical examples were taken from a representative industrial job shop. In their implementation, only bottleneck and potential bottleneck machines have been considered for scheduling.

Czierwinski and Luh (1994), in their paper, addressed the issue of scheduling with bills of materials and proposed a Lagrangian Relaxation technique. The method developed is an extension of previous work of Hoitomt et al. (1993) on the scheduling of job shops with the constraints imposed by the bill of materials. The integration of bill of materials and scheduling improves the manufacturer's ability to meet the promised delivery dates and enhances a manufacturer's credibility and competitiveness in the market place.

Chang and Liao (1994) developed an optimisation model for scheduling flexible flow shops with no set-up effects and proposed a lagrangian relaxation decomposition technique.
The proposed methodology consists of three parts: 1) a nominal scheduling algorithm that based on lagrangian relaxation and minimum cost linear network flow, 2) a fast rescheduling algorithm for timely adjusting the nominal schedule to cope with disturbances such as machine break down, failure of transfer devices etc., and 3) periodic rescheduling using nominal algorithm based on the latest system status. This method is quite suitable to handle uncertainties in the production systems since the rescheduling mechanism adjusts itself for small disturbances and schedules are regenerated periodically.

2.4.2.3 Priority dispatching rules

Stecke and Solberg (1982) investigated various operating strategies (sixteen priority rules under five different loading policies) on a caterpillar FMS by means of deterministic simulation with the number of completed assemblies as performance criterion. They reported that, for the caterpillar's system, SPT/TOT (shortest processing time for the operation divided by the total processing time of the job) provides best solutions for the performance measure considered. It is further indicated that the choice of applicable operating strategy depends on many particular system variables and is highly system and performance criterion dependent. They concluded that each system deserves an individual study of various loading and real-time control strategies, through a detailed simulation, to choose the best methods.

O'grady and Harrison proposed a priority index, which is a linear combination of operation times and due dates, to the job shop sequencing problem. The priority index is defined as follows.

\[ P_i = (A_i \times T_i) + (B_i \times S_i) \]

Where,
- \( P_i \): priority index of job 'i' at its current stage
- \( A_i \): a, \( l \times m \) operation times coefficient vector for job 'i'
- \( T_i \): a, \( m \times 1 \) vector which contains the remaining operation times for job 'i' in the processing order
- \( m \): number of remaining operations with job 'i'
- \( B_i \): the due date coefficient vector for job 'i'
- \( S_i \): the due date slack for job 'i'

They applied simulation and modified Hook-Jeeves search technique to find an optimal \( P_i \) (optimal combination of \( A_i \) and \( B_i \)) for the current state. This method, though useful for dynamic situations, involves extensive computation to carry simulation combined with search.
Cenna and Tabucanon (1991) compared five dispatching rules on a bi-criterion (minimisation of flow time and minimisation of maximum tardiness) scheduling problem associated with parallel identical machines through simulation. They considered the following dispatching rules: Shortest Processing Time (SPT), Earliest Due date (EDD), Minimum Slack Time (MST), Longest Processing Time (LPT) and the sequence given by the algorithm of Wassenhove and Gelders (W&G). The following ranking of the dispatching rules for the above bi-criterion objective is made from the simulation result: W&G, MST, LPT, EDD, SPT. An important point to be noted here is that the performance of dispatching rules or heuristics is objective-dependent.

Karsiti et al. (1992) proposed a scheduling technique that works in two stages to determine i) the appropriate machine to perform the required operations and ii) how the jobs at the machines should be sequenced. They conducted a simulation study employing the proposed heuristic routing algorithm known as the minimum job in queue (MINJQ) that assigns the operations to machines in conjunction with four dispatching rules (FCFS: First Come First Served; SPT: Shortest Processing Time; EODT: Earliest Operation Due Time; EFDD: Earliest Final Due Date) which sequence the jobs at the machine queues. The EFDD rule is found superior to others for the two performance measures of minimum number of late jobs and minimum normalised total tardiness.

Kusiak and Ahn (1992) provided a list of non-due date related dispatching rules and due date related dispatching rules, and introduced another dispatching rule 'Most dissimilar Resources (MDR). The MDR rule is found suitable for maximisation of resource utilisation.

Mujanah et al. (1993) investigated the effect of six priority rules (FCFS: First Come First Served; SPT: Shorest Processing Time; LPT: Longest Processing Time; SRPT: Smallest Remaining Processing Time; LPRT: Largest Remaining Processing Time; Random) on various objective criteria (makespan, mean completion time, total waiting times and total idle times) in a flow shop using simulation. The following conclusions are drawn.

1. SPT gives the best performance for average makespan time.
2. SRPT provides best results for mean completion time objective.
3. SPRT rule tends to be the dominant rule concerning total waiting time performance.
4. LPT and LPRT rules give best results for average total time performance at small and large shop sizes respectively.

Their analysis shows that the choice of a priority rule depends on the objective criterion.
Chen and Diamond (1994) considered a hierarchical multiple criteria scheduling problem on parallel-machines. They proposed a generalised SPT algorithm that belongs to list scheduling heuristics where the jobs are ordered by priority first and then by processing time.

Jewkes (1994) addressed two priority-based scheduling rules based on the classification of the part types, motivated by the goal of mean flow time in a single stage manufacturing system with repair. The parts (customer) were classified according to the nature of task as Class 1 (Parts visit for both processing and inspection) and Class 2 (Parts visit for rework/repair only). Two priority-based rules, Rule 1 (Priority is given to class 2 over class 1) and Rule 2 (Priority is given to class 1 over class 2) in conjunction with shortest remaining processing time (SRPT) scheduling policy and a discussion on the preferability of each rule by means of queuing analysis are presented.

Mahadevan and Narendran (1994) addressed and analysed a few vdrs for AGV-linked FMS. It is indicated that the Nearest-Idle-Vehicle (NIV) rule performs best in the case of multi-vehicle configuration.

Srinivasan et al. (1994) indicated a few vdrs to operate AGVs in FMS and they are:

- **FCFS**: First-come-first-served
- **MFCFS**: Modified-first-come-first-served
- **FEFS**: First-encountered-first-served
- **MOQS**: Maximum-outgoing-queue-size-first
- **STTF**: Shortest-travel-time (distance) -first
- **MODFCFS**: MODified-first-come-first-served (another modification)
- **CLF**: Closest-load-first
- **FLF**: Furthest-load-first
- **LIV***: Longest-idle-vehicle
- **NV***: Nearest-vehicle
- **RV***: Random vehicle

(*Note: Applicable when number of vehicles are greater than one*)

They pointed out that the performance of the analytical models depends on the vdr adapted for the system and it is difficult to generalise relationship between the models and the vdrs.
Bahouth and Foote (1994) considered an assembly job shop scheduling problem. They viewed it as a case of managing two-bottleneck machines in flow shop and based on that proposed three sequencing rules (JNP: Johnson No Priority; JHP: Johnson Half priority; JFP: Johnson Full Priority). They compared these rules with SPT rule with makespan objective criterion in a simulation model, developed for a machine job shop. No generalisation had been possible to achieve with their results.

Waiker et al (1994) evaluated ten different dispatching rules with respect to six different performance criteria under light, medium and heavy shop loads. They carried out a simulation study to determine the ranking of the dispatching rules for a given shop load and performance criteria and investigated the performance of these rules. The experimental results are conveyed with critical comments on the performance of the dispatching rules under different loading conditions of the shop. It is indicated that the two rules, 'shortest processing time' and 'least work remaining', perform well under criteria related to the processing time. It is also found that the rules that perform well in average and r.m.s. tardiness, perform poorly under percentage-of-jobs-late criterion. Their work indicated that the performance of dispatching rules depends on the shop congestion level and the performance criterion chosen.

Piplani and Talvage (1995) studied the performance of a few launching strategies that select and admit a part type to enter into the closed manufacturing systems periodically, in conjunction with different dispatching rules that select a machine for operations from the alternate route choices through simulation. They indicated that the performance a particular control strategy depends on the simulation experiment setting that reflects the operating environment of the real systems.

Selladurai et al. (1995) developed a computer simulator for dynamic scheduling of FMS and investigated the effects of few scheduling rules viz. SPT, LPT, EDD, SIO (Shortest Immeneninent Operation Time), MST, SLACK/LRO (Slack time divided by Largest Remaining Operation Time), on several performance criteria. Their simulator is a cheap and useful tool to find the appropriate dispatching rules for the system. However, the simulation is an evaluation procedure that predicts the possible performance with each control strategy and needs an exact specification of the operating environment.
2.4.2.3 Heuristics

Shanker and Tzen (1984), as an alternative to MIP they formulated, proposed certain heuristic loading policies for loading and sequencing problems and compared the proposed heuristic methods with the analytical MIP method. Further, they suggested five different loading policies for the problem of minimising system workload unbalance and investigated the system performance with sequential and heuristic loading policies in conjunction with four dispatching rules using simulation. They pointed out that investigation with loading policies in conjunction with dispatching rules is meaningful and stressed the need for further analysis with more dispatching rules and loading policies.

Chan and Pak (1986) proposed two heuristic algorithms for solving the scheduling problem with the goal of minimising the total cost of tardiness in a statically loaded FMS. The proposed heuristics are compared with a few published algorithms that dealt static problems in terms of makespan and average lead-time performance. Modifications required to handle dynamic cases are also addressed in their paper. They showed that both algorithms work satisfactorily under static and dynamic conditions than priority-based rules.

Shaw (1987) addressed the use of local area network (LAN) in cellular systems to distribute efficiently the orders to various cells. He described FMS as a loosely coupled system of co-operating flexible cells in which each cell can be set up to produce items belonging to a range of several part families, but in which a particular cell holds a competitive advantage over other cells on a specialised subset of the jobs. The proposed distributed scheduling approach is based on a network wide-bidding scheme wherein the scheduling decision to assign the job to the appropriate cells is made by collecting the price of each cell for taking on the job. He employed earliest finish time of the job, which is submitted for bidding, as the price of each cell (bidding-EFT). The other bidding scheme that is suggested use bidding-SPT (the cell that can process the part with shortest processing time) as the price. He compared the performance of bidding-EFT and bidding-SPT heuristics of the proposed method with a myopic-SPT that employed a centralised scheduling scheme through a simulation study. The closeness to the optimal is not established.

Nasr and Elsayed (1990) have developed a MIP model for a general job shop type machining system with alternative machine tool routings. On the basis of that formulation, they presented two heuristic algorithms, namely, Bound Algorithm and Shortest Finish Time (SFT) algorithm, to minimise the mean flow time. They claimed that these algorithms solve large
problems in relatively short time. However, the proposed methodologies are not compared with other methods.

Liesten (1990) addressed deterministic flow shop sequencing problems with limited buffer storage. He identified and analysed the heuristics that are employed to such problems. The results indicate that their performance heavily depends on the objective criterion and number of machines.

Mukhopadhyay et al. (1991) proposed a heuristic that uses an analytical hierarchical process to determine an optimal schedule integrating tool allocation, pallet allocation and material handling system. The procedural steps of the proposed heuristic are: assignment of tools to machines (using a sub-heuristic that finds suitable combination of tool sets on machines based on the tool-operation and tool-machine compatibility), determination of part priority and ordering the parts (using a composite priority index, that is calculated with the characteristics of part or machine, for part or machine preference), allocation of pallet, determination of machine priority and selection of a particular compatible machine for the selected part and selection of a AGV to transfer the parts. The performance is not compared with related heuristics.

Mahmood and Dooley (1991) examined the group scheduling heuristics with due date adherence and flow time performances under various experimental conditions. They considered twelve heuristics that resulted on combining three dispatching rules for sequencing the jobs in each sub-family and four heuristics (two exhaustive and two non-exhaustive) for ordering the queue of sub-families. It is pointed out that, in general, the performances of exhaustive heuristics are superior to non-exhaustive heuristics. This indicates that improving the schedules within the sub-family will show further improvements in the overall schedule.

He and Kusiak (1992) addressed three different industrial scheduling problems, developed mathematical formulations and proposed heuristic algorithms for each problem. The first problem is the single machine scheduling problem associated with sequence-dependent set-up times and precedence constraints. The second problem deals with a machine cell that includes five lathes of Okuma type. The third problem is a laser cutter scheduling problem. Their work indicates that the formulation and solution methodology differ with problem environment.

Sabuncuoglu and Hommerertzheim (1991) proposed a dispatching algorithm that considers the availability of both machine and AGV to select a job for loading. They compared
the performance of the proposed methodology with a few simple dispatching rules by using the mean flow time and mean tardiness criterion through a simulation test. It is indicated that the proposed method outperforms the dispatching rules when the mean tardiness is the scheduling criterion. The results are obtained with specific experimental conditions and so there is a need for further analysis under wide experimental conditions.

Fanti et al. (1993) indicated that the routing flexibility is one of the main peculiarities of FMS. They proposed three heuristic methods for real time routing selection, named Total Route Workload (TRW), Route Bottleneck Workload (RBW), and Shortest Clear-Out Time (SCOT), the decision to assume at the operational level in real time scheduling. The first two approaches select the path with minimum total workload in progress or minimum current workload at the bottleneck machine respectively and the third approach performs routing choices measuring a merit index that depends on the near future system behaviour. They assumed two production systems and compared the performance. It is shown that, the SCOT approach exhibits the best throughput values. The SCOT discipline involves detailed information on the system's state and requires short discrete-event simulation runs.

Mori et al. (1993) presented a Hierarchical Decentralised Scheduling (HDS) method, which consists of a Strategic Level Scheduler (SLS) to generate manufacturing plan and an Operational Level Scheduler (OLS) to implement SLS output in real-time. Heuristic algorithms for SLS & OLS are addressed. The performance evaluation is carried out by discrete event simulation using timed petri-net (TPN) package. The proposed methodology is useful for large and automated factory that operate under dynamic environments such as failures of machines or AGVs. The weakness of the method is that both the SLS and OLS need to be adjusted each time when the state of the system changes drastically.

Kim and Yano (1993) considered the problem of assigning operations and their associated tools to machines (or group of machines) to maximise the throughput for a specified steady-state mix of orders. They viewed the problem as a two-dimensional bin packing problem in which a bin represents a machine or machine group and the dimensions denote tool slots and processing times. They devised various heuristic algorithms that first allocate tools to operations and then assign the operations, and compared the performance with maximum deviation ratio and relative performance ratio. It is shown that the heuristic algorithms that formulated with time slot ratios perform well. However, no concrete conclusions are drawn.
Ruben et al. (1993) examined the direct and interactive effects of a variety of shop factors on the performance of the best, previously reported, group scheduling and single stage job shop dispatching heuristics through simulation. The heuristics examined in their study are:

**Single stage heuristics**: 1) SPT, 2) FCFS, and 3) Two class SPT (SIX)

**Group scheduling heuristics**: 1) MSSPT (Minimum-set-up-SPT), 2) DDSIX (Due-date-SIX), 3) ESCIX (uses set-up time for queue selection), 4) WOSIX (total work contend based), 5) FCFCFS (the queue with earliest arriving job)

The performance measure used are: average time in system, average tardiness, average percentage of jobs tardy, and average lateness. They ranked the heuristics for various experimental conditions and performance measures. The results indicate that performance comparable to that of the two-stage heuristics can be obtained with the easily implementable single-stage heuristics when factors which lessens the impact of set-up times are in place.

Ulsov and Bilge (1993) pointed out that the majority of the reported work that dealt with the subject of material handling system scheduling is set out as a comparison of various variants in reaction to a pre-specified schedule and on a particular layout. They further indicated that the co-ordination and integration of material handling system with machine scheduling during the scheduling phase of the FMS have not received much attention. Their paper is an attempt to make scheduling of AGVs an integral part of the overall scheduling activity in FMS environment. They proposed an iterative procedure to generate schedules for machines and AGVs simultaneously. Their methodology is based on the following three components:

i) an algorithm that generates machine schedules;

ii) an algorithm that finds a feasible solution to the vehicle scheduling problem given the machine schedule;

iii) an iterative structure that links the two and facilitates the search for a good solution.

They employed Giffler and Thompson's active schedule algorithm (Giffler and Thompson 1960) and Non-delay schedule algorithm (French 1982) using different priority dispatching rules for machine schedule generation. They applied Sliding Time Window (STW heuristic) concept to provide a feasible vehicle schedule. Since the iterative and heuristic procedure requires more CPU time, the authors themselves have suspected whether the iterative procedure can be adapted in real time dynamic scheduling.
Chen (1994) proposed a heuristic algorithm, which selects an appropriate rule (either pull or push rule) based on average moving time between two consecutive events and average machine processing time, to schedule robot activities in FMS. The selected rule is used until new batch of jobs arrives. The performance of the pull-push heuristic is compared with that of three scheduling rules, namely, SFT (Shortest travel time first) FRFS (First request first served), and RND (random) on a manufacturing cell that produces shafts and axles of electric motor. The throughput is used as the measure of performance. The results indicate that the pull-push heuristic performs better than the other classical rules.

Kim and Yano (1994) developed a procedure to a part type selection problem in FMS. The proposed approach, first determines an approximate schedule using list scheduling heuristics and then sets the release priorities for the parts.

Arzi (1995) developed a two-level distributed production control system (DPCS) to (1) assign the parts to one of the cells in the system based on the relative advantage of a cell in terms of workloads (cell lot selector) and (2) control parts within each cell by its own parts flow controller (parts flow controller) that use 'Largest Relative Advantage adjusted for Orders' (LRAO) rule. The operation of the proposed DPCS is governed by several control parameters such as lot selection parameter, limits on storage space, delay correction parameters and so on. These parameters have to be determined to specific operating circumstances and ordering streams. Development of a system with learning capabilities to recalibrate itself when changes occur in the ordering stream is identified as future research area.

Ovacik and Uzsiy (1995) presented a family of rolling horizon (RHP) heuristics for minimising maximum lateness on parallel identical machines in the presence of sequence dependent set-up times and dynamic job arrivals. They compared their performance with two bench marking dispatching rules through simulation and showed that the RHP heuristics yield better solutions than dispatching rules at the expense of acceptable increases in computational effort.

Asseri and Harigu (1995) proposed a heuristic algorithm for scheduling multiple items at a single facility (a bottleneck machine) with least holding cost under the set-up and production capacity constraints. The proposed method is evaluated by comparing the optimal solutions obtained by Luss and Rosenwein's mathematical model.
2.4.2.4 Local search Algorithms (Iterative search, Simulated annealing and Tabu-Search)

Lee and Dicesare (1994) used petri-nets to model the scheduling problems in FMS. They proposed a heuristic search algorithm that generates and searches for an optimal or near optimal feasible schedule with the firing sequence of the transitions of the Perti-net model. Their search mechanism is an iterative procedure that takes much time.

Singh and Bochynek (1997) compared several modern heuristic search methods (GA, SA, TS and Hybrid Search) with respect to a standard cell placement on VLSI design sequencing problem in terms of solution quality (makespan criterion) and computational time. They showed that any method can perform good or bad depending upon its parameter setting and there is a trade-off between run time and solution quality. They concluded that SA proved to be the most suitable for a sequencing problem like standard cell placement. However their conclusions are based with respect to a particular problem. The application of other methodologies can not be neglected for other problem areas.

2.4.2.5 Evolutionary Programs (Genetic Algorithms)

Biegel and Davern (1990) showed the method of applying genetic concepts to scheduling problems. An elementary n-task one processor problem is provided to demonstrate the GA methodology in the job shop scheduling (JSS) problem arena. Further they extended the technique to the n-task on m-processor (serial) case. Their paper is a sort of primer on applying genetic algorithms to the JSS problem. The future research directions indicated are: inclusion of other constraints such as precedence relationship, due date adherence, inventory level, and so on; combining GAs with heuristics; building new schedules based on learned information from prior schedules. GA is addressed as a proven tool that can be employed for any type of scheduling problems.

Dorndorf and Pesch (1993) proposed a GA based on the idea of using a chain of priority rules which fit the needs of a particular problem instant. Within the GA each gene represents a priority rule from the set of priority rules. While decoding a chromosome, to generate a feasible schedule, the $i^{th}$ rule is applied for scheduling the $i^{th}$ conflict in schedule generation procedure. In their GA process, they employed GT algorithm (1960) to generate an active feasible schedule and used the makespan time of the schedule as the fitness parameter.

Sridhar and Rajendran (1994) addressed a GA for part family grouping and scheduling the parts within the part families in a flow line based manufacturing cell. In their contribution, the objectives of makespan and total flow time have been considered independently and
together. The proposed methodology is compared with the existing SA heuristic and indicated that GA fares better than SA in minimising both makespan and total flow times.

Weindal and Garlichs (1994) proposed a GA, for the decentral production scheduling of assembly system. It is shown that, by modelling living beings in the form of their gene (=alternative scheduling) and environment allocated to them (=goal fulfilment), it is possible to computer to simulate the evolution or further development of scheduling plans for enhanced goal attainment. The advantages of the GA compared with classical optimisation technique include shorter computing times without detriment to high quality or goal attainment in the solution found.

Billo et al. (1994) illustrated the usage of GA to solve problems associated with the formation of machine cells. The GA formulation included a fitness function that incorporated two competing objectives: maximise similarity of parts and minimise number of cells.

Stockton and Quinn (1995) proposed a GA for an aggregate production scheduling problem. They explained how GAs can be useful to develop aggregate production plans and illustrated how this type of algorithms provides the means by which the limitations of existing aggregate production plan technique can be overcome.

Chen et al. (1995) proposed a GA based heuristic for flow shop problems with makespan as the criterion and compared the efficiency among the proposed GA heuristic, Widmer and Hertz's heuristic (SPIRIT), and Ho and Chang’s heuristic (HC). Their results reveal that the GA based heuristic is an effective method for flow shop problems and can be applied to any hard optimisation problems.

Dorndorf and Pesch (1995) proposed another genetic search process that is encoded with machine numbers. In the proposed GA, the operations are ordered by inserting the optimal sequences of each machine, which is obtained separately with a B-B technique, with respect to the machine order defined in the chromosome and a feasible schedule is generated. This method provides near optimal schedules but requires to perform additional computation to find optimal machine sequences.

Sridhar and Rajendran (1996) developed a GA to the problem of scheduling in flow shops and cellular manufacturing systems with the multiple objectives of minimising makespan, total flow time and machine idleness. They evaluated the proposed GA by comparing it with a heuristic method suggested by Ho and Chang (HC heuristic) and indicated that the proposed
GA yielded much better quality solution than HC heuristic. It is inferred that the GA is an adaptive search method that can be used to tackle hard combinatorial optimisation problems.

Mattfeld (1996), in his book on 'Evolutionary Search and the Job Shop', addressed several GA based algorithms developed for job shop problems. He indicated the followings:

1. Different coding schemes and feasible schedule generation methods are employed to evolve an optimal or near optimal schedule.
2. Most common representation of a chromosome is a genotype coding scheme, which is later decoded into a feasible phenotype (schedule) in the fitness evaluation.
3. Different crossover and mutation operators are tested.
4. The schedule generation technique plays a central role in evolutionary scheduling.
5. The GT is used by many GA based methods to generate active feasible schedules in job shop environment.

Kopfer and Mattfeld (1997) proposed a hybrid GA that applies a local search operator in an evolutionary framework and evaluated with the results obtained for a well-known test suite of particularly hard, but medium-sized job shop scheduling problems (including the famous mt10 problem). The presented results encourage to use GA framework for future research.

Adelsberger et al. (1997) analysed the mutation rates of different mutation operators (Random, Inversion, Insert and Swap) and crossover operators (twenty operators) and studied their effect on the convergence of the GA with a single machine sequencing problem that minimises the total absolute deviation (earliness plus tardiness). They state that as far as sequencing problems are concerned, crossover operators are afflicted with implicit mutation to such a degree that on the basis of a generational GA no adequate convergent optimisation process can be achieved for the scheduling problem under consideration.

Janiak and Portmann (1997) classified the GAs into four categories as:

- **GA1**: GA with specific cross-over and specific mutation techniques
- **GA2**: GA with classical cross-over and specific mutation techniques
- **GA3**: GA with specific cross-over and classical mutation techniques
- **GA4**: GA with classical cross-over and classical mutation techniques

They investigated the performance of them on a single machine scheduling problem where the processing times are dependent on transferable resources. They concluded that the GA2 shows best results in average.
2.4.2.5 Artificial Intelligent (AI) techniques (Knowledge-Based and Expert Systems)

Kusiak (1985) addressed AI and operations research (OR) formulations for scheduling problems in FMS. The features and differences between AI and OR are explored. It is shown that there is a potential for building systems incorporating AI and operations research techniques.

Shaw and Whinston (1986) addressed an artificial intelligent approach to the scheduling of FMS. They proposed a KBsystem that consists of three components: a world model, an action model and an inference engine. The world model comprises the data base that describes the state of the system. In the action model, a knowledge-based framework that constructs a course of action by state changing transformations, represented by operators. The inference engine, which selects a sequence of operators to achieve at the goal state from a given initial state, generates the process plan and schedule. The description of the world model influences the applicability of the proposed methodology in real environment. They laid foundation for integrating intelligent planning, scheduling, and machine learning in FMS.

Wu and Wysk (1988) proposed a Multi-Pass Expert Control System (MPECS) which incorporates an intelligent scheduling module to suggest alternate scheduling rules according to the problem nature and requirement, a generative simulator to evaluate the scheduling rules by simulation and a cell control module to implement the best policy evolved. Three important observations pointed out are: 1) The length of the simulation windows is recognised as an important factor for MPECS. 2) The performance of MPECS is significantly better than the performance of single pass traditional scheduling methods. 3) The performance of MPECS is better than pure multi-pass scheduling that fixes the alternate dispatching rules without any reasoning. The computational time is large since a simulation run is needed each time when the state changes.

Chandra and Talvage (1991) developed an intelligent job dispatching strategy for FMS that takes into account the current state and trends of the system. They used the following four pieces of information to make scheduling decisions in the proposed method: shop congestion level, preference of a part for a machine, criticality of a part and current job objective. The proposed intelligent reasoning procedure addressed as 'EXPERT' has been found to achieve better shop performance than some of popular dispatching rules (SPT: shortest processing time; EDD: earliest due date; LSPO: least slack per operation; LRS: least relative slack), the improved performance being due to the ability to respond to changing
circumstances. This paper motivates the use of logical reasoning with common sense for scheduling FMS.

Nakasuka and Yoshida (1992) proposed a knowledge acquisition learning scheme to predict which rule is the best in a certain shop floor status and based on that, have developed a prototype computer program named 'learning aided dynamic scheduler (LADS)' to select a dispatching rule for real time operations. Their work is briefly outlined as follows: Thirteen attributes characterise the shop floor status. For a given state of the system characterised by their basic attributes, a simulation run finds the best scheduling rule by mapping those attributes. Many such problem instances builds the data base. A large number of data on the basic attribute set Vs the best schedule rules are then input to the inductive learning part that formulates a binary decision tree operator. In the run time module, at each scheduling point during real time operations, the binary decision tree operator selects one scheduling rule.

Kusiak and Ahn (1992) developed a rule-based intelligent scheduling system, which consists of four components: algorithm selector, rule selector, process reactor, and the rule base, to generate schedules that correspond to the manufacturing environment and the objectives of the system. Their methodology is outlined as follows. The algorithm selector selects a particular algorithm based on problem characteristics. The rule selector recommends a suitable dispatching rule to be followed using the acquired knowledge base (rule base) established for various exogenous manufacturing conditions (scheduling objectives, system load levels, resource constrainedness and due date tightness). The process reactor checks the applicability of the selected rule in the endogenous manufacturing conditions (shop status such as inventory level, length of the queue and bottleneck machine number, etc., and preference constraints) and suggests modifications accordingly. The rule base, which is obtained by a discrete simulation assisted knowledge acquisition process, comprises of production rules that relates the processing environments and the algorithms/dispatching rules. The suggested intelligent scheduling scheme is more flexible to handle dynamic scheduling environment. They established production rules with 27 dispatching rules and three load levels for selecting dispatching rule and illustrated their methodology.

Kim and Kim (1994) presented a simulation-based real-time scheduling methodology for an FMS which employs discrete event simulation with information of the current state of the shop floor, estimates system performance with different scheduling rules (thirteen) and selects the best dispatching rule for a given criterion, and the selected dispatching rule is used
until the difference between the actual performance value and the value estimated by simulation exceeds a given limit. The suggested method, though applicable, depends on the speed at which the simulation is carried out and the length of the monitoring period which are important aspects of real-time control.

Li and She (1994) presented the application of knowledge acquisition technologies of artificial intelligence to arrive at the scheduling knowledge in FMS using object-oriented simulation, rule base inferring, clustering and decision tree construction techniques. Their work is the search towards establishment of the inverse relationship between the operating performance and its corresponding decision parameters that arise at the operational level (part-releasing and part dispatching) in the system. The application of machine learning techniques to obtain scheduling knowledge is an effective method in coping with the complexity of FMS. Any how, devising the scheduling rules requires to pre-establish the configuration and the operating procedures of the system.

Chiu and Yih (1995) considered a dynamic scheduling problem in a distributed manufacturing system and proposed a learning-based methodology to extract scheduling knowledge for dispatching parts in such systems. The acquired knowledge is then used to switch intelligently to different priority dispatching rules at various scheduling points, based on the status of the system specified by a set of system attributes. The system attributes such as number of jobs, number of remaining operations, minimum slack time of candidate jobs for a machine and so on, different objective functions, and four simple policies have been considered to define the status of the system.

Wang et al. (1995) presented a system-attribute oriented knowledge-based scheduling system which uses maximise resource utilisation (MRU) algorithm for the generation of training examples and an inductive learning method (continuous iterative dichotomiser 3 'CID3' algorithm) for scheduling knowledge acquisition and rule inferencing. At the various points during a simulation run, the system attributes of the parts and the shop status, which describe the instantaneous characteristics, and the best MRU dispatching rule for that instant have been found and used as training examples.
2.4.3 Other Related Literature

Wilhelm and Shin (1985) investigated the influence of alternate operations (an operation can be assigned to more than one machine) in FMS using simulation. They suggested three schemes for implementing alternate operations and compared with the performance achieved using no alternate operations in terms of makespan time, machine utilisation, production cycle time, required storage space and number of vehicles required. Improvements are shown with schemes that considered alternate operations. It is pointed out that the future research should include features such as machine breakdown, machine buffers, eligibility of several alternate machines to perform an operation to completely understand the impact of alternate operations on the system performance.

Carrie and Perera (1985) focused their research on the effect of tool changeover, which arises during the change in product variety and tools wear, in scheduling. The analysis is carried out with an example FMS, for which it is shown that tool capacity constrains the scheduling decisions. It is shown that the number of tool changes due to product variety are minimal. It is further stated that devoting excessive effort to the product variety aspect of the problem is unwise and it would be better to devote attention to efficiently performing tool changes for whichever causes, rather than evolving a scheduling method that assumes they are an immutable constraint.

Chokravarty and Shtub (1987) developed a mathematical program for determining product groups (a set of parts from the list of parts) that minimises the total cost of set-up and holding during the planning horizon under the machine capacity, processing and tool magazine constraints. Since the mathematical program technique involves complex computations, it is suggested that heuristic algorithms would be meaningful to such problems.

Sabuncuoglu and Hommerertzheim (1989) reviewed the developments in simulation and discussed the role of Expert Simulation Systems (ESS) in FMS. It is concluded that simulation and expert systems are the two promising areas which will play an important role in computer integrated manufacturing systems.

Hutchison (1991) in his review on FMS scheduling literature, analysed different types of FMS (Dedicated, Intermediate, Random, Flow shop, Job shop etc.) with different scheduling schemes (Input Sequence, Detailed Schedule, Real time, Off-line etc.). On the basis of the review, it is concluded that off-line schemes with rescheduling aspect are more appropriate for random job shop systems and they can take a leading role in FMS scheduling.
Shanker and Agarwal (1991) reviewed the literature that addressed the loading problems and resource considerations in FMS with 31 references. The summary of the review is:

1. Loading problem is a short term decision which determines the utilisation of capital intensive resources and productivity of the system;
2. The objectives are several and are mainly oriented towards parts, machines and total cost and the selection of suitable objective is a difficult task.;
3. Solution to the loading problem should pave the way for efficient scheduling and dispatching.

Dhar (1991) outlined a framework of interrelationship between FMS planning problems, planning models and functional organisation. The scheduling is indicated as the lowest level planning process in the material requirement planning (MRP) system. Planning and interlinking are the two major problems identified.

McCahon and Lee (1992) modified Campell, Dudek and Smithy (CDS) algorithm to handle the flow shop problem with the processing times given as fuzzy numbers. This paper introduces the concept of applying fuzzy theory to scheduling problems. It is indicated that this fuzzy analysis procedure and subsequent sequence generation calculations are cumbersome if the problem becomes fairly large.

Dines (1993) classified manufacturing enterprises into four fundamentally different sectors, based on complexity/uncertainty grid and identified the reasons for the lack of substantial success of MRP in the process and fashion sectors. They are: lack of responsiveness to accommodate short term changes, unrealistic capacity planning based on infinite capacity and without feedback on shop floor status. He also suggested that Finite Capacity Scheduling (FCS) systems can be an alternative to MRP for production control in process-based manufacturing. They can plan to finite capacity and derive materials and resource requirements to support production schedules. Also, it is pointed out that a logistic integrated system based around finite scheduling has to focus on:

- a quick response sales order processing system,
- an overall capacity planning system for long term planning,
- a finite schedule system for short term planning coupled with rapid reporting module of the shop floor and
- a management information system for tactical strategic decision support.
Gunasekaran et al. (1993) reviewed 196 articles that address FMS to identify the applications of the models in practice and in future research directions. They addressed various problems that are encountered with design and operational issues in FMS. They indicated, with respect to scheduling, the models in FMS lack to provide integration between the various inter-related sub-systems and requires different planning and scheduling methodologies for each class of FMS.

Custodio et al. (1994) addressed the three-level hierarchical structure of production planning system and used Fuzzy Logic Applied to Scheduling (FLAS) at all the three levels to handle the vagueness and lack of precise knowledge in production systems. Simulation results reveal that the proposed system exhibits good performance measure, in terms of a high production percentage, and a low WIP, under resource failure and demand variations. This is a theoretical framework that supports the application of fuzzy logic and fuzzy control in the areas of production planning and scheduling.

Perkins et al. (1994) investigated the stability and performance of generalised Round-Robin scheduling policies in FMS.

Kolen et al. (1994) addressed the possible number of alternate assignments on a set of identical parallel machines when there exists a priori uncertainty with respect to processing time of one job under a particular dispatching rule.

Sevas'janov (1994) reviewed a set of polynomial algorithms (seventy) in relation to such a class of problems as flow shop (FS) open shop (OS) and also the volume calendar problem (VCP).

Iwata et al. (1994) addressed the concept of random manufacturing systems (RMS) that are suitable to realise flexible and adaptive production for dynamically changing orders. The operation of RMS is carried out according to the tender-based decision-making. The concept, architecture, and operation procedures of RMS are well suited for FMS.

Wan (1995) investigated the two different scheduling schemes (off-line and real-time) under stochastic problem settings. They suspected the superiority of the off-line schemes in the following two aspects: the response time to revise the schedule and the assumption of deterministic processing times. They constructed examples in static as well as in dynamic settings with stochastic operation times and showed that real time schemes are capable of providing good results. However, it is believed that the performance of a scheduling method is problem dependent.
Joshi et al. (1995) described the method of formulating FMC models with the physical activities, system actions, and individual machines comprising the manufacturing cell as a first step towards the development of control schemes for implementation. Further, they addressed a hierarchical control architecture to control the components at various levels. The issues addressed provide meaningful guidelines to design a model.

Wu and Li (1995) proposed a scheduling graph, an alternative representation of a Gantt chart, to react quickly and easily to the inevitable rescheduling changes that arise in flexible manufacturing environments. The nodes of the scheduling graph indicate the time elements associated with an operation in a machine and the arrows represent the precedence relationship between the operations. They showed that, with this form of representation, the schedule can be reconstructed with simple logical rules (algorithms) as and when the original schedule needs modification. Anyhow, the proposed methodology has not been evaluated in their contribution and the quality of solution is questionable.

Lane and Evans (1995) viewed the production scheduling as a part of business function that maintains a complete set of plans (master schedule and MRP) and detailed schedules. A decision support software is proposed that has facilities for defining policies to handle numerous minor problems and complete problem-solving strategies to deal with major problems.

Artiba et al. (1997) provided a unified framework and a classification scheme to production systems. On the basis of that, they modelled a hybrid flow shop with the process flow nature, and general and complementary characteristics. Their paper provides the basic terminology associated with modelling production system for scheduling.

Schutlz and Mertens (1997) compared the performance of an ES, a GA and priority rules for production scheduling. Their remarks are as follows:

- The ES approach provides good quality solutions with short run time, but requires detailed knowledge about the relations within the knowledge bases. Additionally, the ES is favourable because the planning process includes local and global shop floor information.

- The quality of the GA depends on run time and needs to speed up the process by incorporating problem specific knowledge.

- The priority rules are best concerning the run time comparison and quite easy to construct, but shows divergent findings.
2.5 SUMMARY

2.5.1 Salient remarks

Scheduling improves productivity

Scheduling, as a part of production planning and control, plays an important role in the entire manufacturing process. A good schedule increases throughput rate, reduces WIP, maximises machine utilisation, meets order due date, satisfies production constraints and further achieves business objectives (refer section 1.2).

Primary resources are machines and material handling system

In general, scheduling is addressed as one that involves the order and timing for assignment of resources to specific operations. Scheduling at CIM level is considered to be a process that relates specific events to specific span of time. Although scheduling refers to the time-phased allocation of all the system resources such as machine, tools and MHS, it is most often applied to the scheduling of jobs on the machine (Sabunuoglu and Hommerstzeim 1991). Since the cost of the machine tools is about 65% of the overall system cost in FMS (Kusiak 1985), most of the scheduling research concentrated on machine scheduling problem. The other schedules are adjusted either prior or later on to the machine scheduling.

FMS requires a dynamic scheduling scheme

The scheduling in FMS falls in line with the general scheduling schemes, but it needs to respond for the changes in the state of the system influenced by both internal (resource failures) and external (marketing and technological) disturbances. The uncertainties such as failure of hardware components of the system, urgent jobs and power cut, enforce to modify and revise the plan. This makes necessary a good scheduling scheme to address the dynamic environment. It is mandatory in FMS which is a computerised automated factory. Providing a powerful tool for the scheduler to quickly and easily react to the rescheduling changes is mandatory in to-day's complex and flexible manufacturing systems. The scheduling schemes essentially incorporate the following two factors: the schedule change must fully reflect the real-time information of the production line and the revision must be completed within a short time in such a way that it doesn't delay any real operations.

Inclusion of alternate process plans enhance productivity

Separating process planning and scheduling results in a production schedule that lacks flexibility and adaptability. Due attention should be given to these two functions to achieve realistic and effective production scheduling. Incorporation of alternative manufacturing plans
will enhance the productivity of the system largely, especially during the dynamic changes in the state of the system. Since the inception of CIM, emphasis is given to include alternate route choice in scheduling analysis.

**Objective criteria depends upon the firm's operating policy**

Production systems are designed essentially to meet demand efficiently. Productivity indicates the efficiency of the system effectiveness and depends on many aspects such as hardware utilisation, tardiness, in-process inventory and manufacturing lead time. One or more of the above parameters are used to measure the efficiency of the system. The scheduling objective criterion is selected based on those performance parameters. The performance parameter is decided based upon the business environment. Objectives in scheduling vary from firm to firm and often from time to time. Traditionally the aim would be either to achieve certain contractual target dates or simply to finish all jobs early.

**Optimisation approaches are ruled out in practice**

Optimisation algorithms provide satisfactorily or optimal results if the problems to be solved are not too large. Most existing scheduling methodologies that give optimal solutions apply only to problems of very small size; i.e., a few parts to be scheduled on a few machines. The use of exact enumeration techniques is restricted to problem sizes of a few hundred operations. Since a problem of practical size may involve hundreds of machines and hundreds of products, each comprising thousands of parts to be scheduled over a period varying from several months to a few years, any method generating an optimal schedule will likely require excessive computation time and will not be not be practical for use on a daily basis. With the growing uncertainty and complexity in manufacturing environment, most scheduling problems have been proven to be NP-hard, that is, the computational requirement grows exponentially as a function of the problem size, and this degrades the performance of conventional OR techniques. Also the modelling is a difficult task. The approximation algorithms produce solutions that are guaranteed to be within a fixed percentage of the actual optimum and are considered as urgent and useful tools for solving discrete optimisation problems (Bruker 1996).

**Branch-Bound algorithms involve large computations**

B-B algorithms cut branches from the enumeration tree and therefore reduce number of generated nodes substantially. B-B algorithms rely on a lower bound and an upper bound of the objective function value (Mattfield 1996). In case of B-B technique, there is a branching
tree corresponding to the decision set, and the number of nodes of this tree increases exponentially with the size of the problem (Portmann 1997). The quality of solutions depends on the bounds, a good bound requires a substantial amount of computation.

**Priority dispatching rules are questionable for stability and are environment dependent**

The priority rules are best concerning the run time comparison and quite easy to construct, but shows divergent findings. In practice production schedules are usually generated either by priority dispatching rules, which may be questionable in optimality and stability, or by a time-consuming computer simulation, which is evaluative rather than generative. Various factors such as configuration of the system, shop load levels and objective criteria, govern the performance of priority rules. Hence any generalisation to use a particular rule is difficult to establish.

**Heuristics yield good results and are robust to specific problems**

The performance of heuristics is satisfactory as long as the operating characteristics and objectives of the system remain the same. The use of specific unvarying heuristic rules, however, leads to practical problems in implementing any sequencing system in a manufacturing concern since an unvarying heuristic is then being applied in what is generally a dynamically changing environment (O'grady and Harrison 1985). However, heuristic off-line procedures in systems with large amounts of routing flexibility have been found to provide good, fast performance (Hutchison 1991).

**Local search methods stuck at local minima**

Local Search algorithms share the basic idea of neighbourhoods. A neighbouring solution, which is derived from its originator solution by a pre-defined partial modification, is called MOVE. A MOVE result in a neighbouring solution which differs from its originator solution slightly. Since the chance of finding an improved solution within a neighbourhood is much higher than in less correlated areas of the search space, local search methods are capable of providing near optimal solution (Kopfer and Mattfield 1997). However, local search based heuristics are known to produce excellent results in short run times, but they are susceptible of getting stuck in local entrapments. These methods strongly rely on its parameter setting and there is always a trade-off between run time and solution quality (Schultz and Mertens 1997).

**Evolutionary programs are useful for any hard optimisation problems**

The genetic algorithm is based on the recognition that evolution, with its principles of mutation and selection, represents an efficient process for solving optimisation problems. By
modelling living beings in the form of their gene (=alternative scheduling) and environment allocated to them (=goal fulfilment) it is possible to computer to simulate the evolution or further development of scheduling plans for enhanced goal attainment. The advantages of the GA compared with classical optimisation technique include shorter computing times without detriment to high quality or goal attainment in the solution found. The formulation decides the quality of solutions. Evolutionary algorithms can be formulated almost independently on the details of the problems under consideration. GA applies a local search operator 'crossover' and a global search operator 'mutation' in an evolutionary framework. These algorithms, combine the advantages of both worlds, the efficiency of local search and the robustness of evolution (Kopfer and Mattfield 1997). Local search offers further improvements of solutions resulting from schedule generation heuristics (Michaelwitz 1994). There will be a question regarding the time required for computation. The present day speed of the computer makes this process successful. The increasing performance of modern computer systems allows the application of new algorithms such as GA, SA, for the production scheduling problem that consumed too much computing time in the past (Schultz and Mertens 1997). In GAs, generally, the scheduling problems have been encoded with genotype decision rules and then decoded to feasible phenotype that generates a feasible schedule at the evaluation stage. Thus the schedule generation technique plays a central role in evolutionary search. Also, the success of a GA strongly depends on how well the coding respects the underlying problem. GA is a stochastic algorithm that produces different results in different runs. The quality of the GA depends on run time and needs to speed up the process by incorporating problem specific knowledge. This random search process is regarded better than simulation in the sense that it guarantees near optimal solutions in actual cases. The scope of applying GAs to scheduling problems is high because they are capable of handling any NP hard problems including JSS problem and the present day speed of the computers is high.

AI techniques depend on its capability to combine miscellaneous and situation-specific pieces of information in a short time

The strength of ES approach is its capability to combine miscellaneous, situation-specific pieces of information that need different decisions. The ES approach provides good quality solutions with short run time, but requires detailed knowledge about the relations within the knowledge bases. Additionally, the ES is favourable because the planning process includes local and global shop floor information.
2.5.2 **State of the art**

1. Majority of the research is confined to specific system configurations.

2. Incorporation of alternate route flexibility will enhance productivity. Separating process planning and scheduling results in a production schedule that lacks flexibility and adaptability, and due attention should be drawn to these two functions to achieve realistic and effective production scheduling.

3. The performance parameter depends on the operating policy of the firm and is based on that scheduling objective.

4. Approximation algorithms have been identified as useful tools for handling FMS scheduling problems.

5. The performance of the priority rules has been analysed with simulation experiment, which is evaluative rather than generative and hence does not guarantee optimal performance in real environment.

6. Many heuristics have been proposed and analysed. They are robust to the system characteristics.

7. Recently, Local search algorithms, GA based heuristics and AI algorithms have been evolving to a great extent.

8. The scheduling schemes essentially incorporate the following two factors: the schedule change must fully reflect the real-time information of the production line and the revision must be completed within a short time in such a way that it doesn't delay any real operations.

9. Majority of the off-line scheduling research deals either with part and machine scheduling or Automated Guided Vehicle (AGV) routing separately. In such cases, the real time operations are affected drastically. Integration of the sub systems is needed to improve the overall efficiency of the system.

10. Off-line schemes have been considered more useful and appropriate in view of the following points: (i) they form a basis for MRP, (ii) the objectives can be met more closely and (iii) the follow up is easy. But their weakness lies in their need for revision and control of schedules due to the changes in priority and large computational time.
11. Real time schemes have been considered suitable for dynamic systems where the computational time is the prime concern. But they do not foresee the future courses of actions and result in sub optimal performance.

The above discussions indicate that there is lot of avenues open for doing research in this area. A few of them are:

- Application of new tools such as local search, genetic search, knowledge-based search and the like.
- Development of off-line schemes with integration.
- Methods to enhance the conventional OR technique by hybridisation with heuristic and heuristic search techniques.