CHAPTER 3

Theoretical Considerations and Literature Review

3.1 Theoretical Considerations

3.1.1 Classification of Scheduling Problems

Scheduling problems can be classified as deterministic or stochastic, static or dynamic.

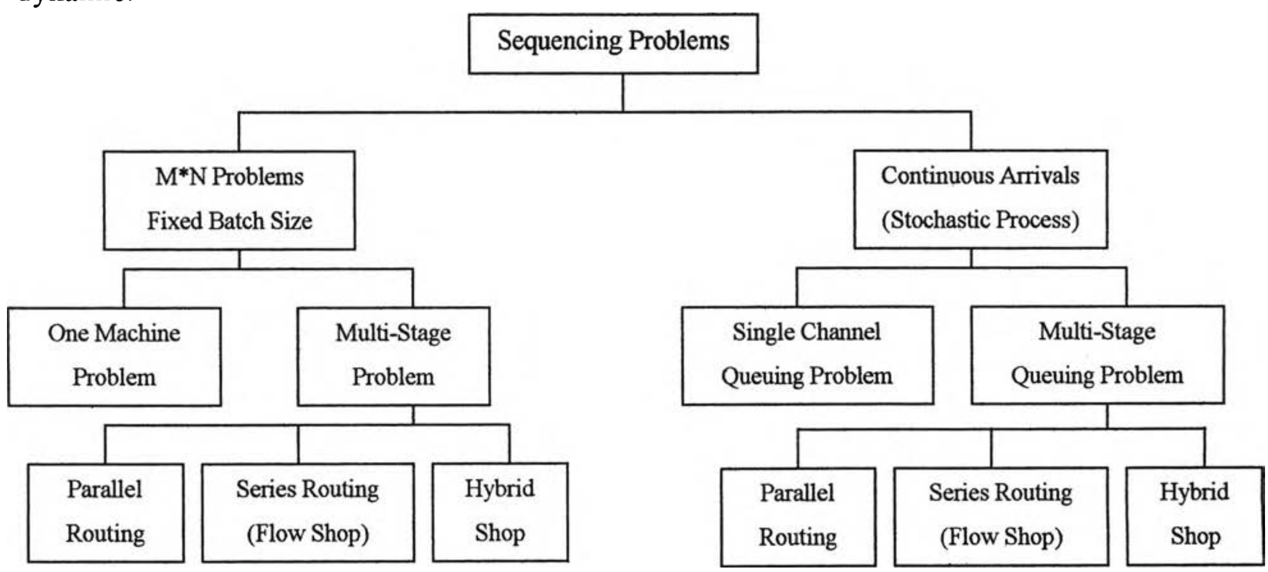

Figure 3.1.1 A classification of Scheduling Problems

3.1.2 The type of machine environments

The number of jobs is denoted by n and the number of machine by m. The subscript j refers to a job, whereas the subscript i refers to a machine.

1.) Single machine

The case of a single machine is the simplest of all possible machine environments and is a special case of all other more complicated machine environment

2.) Identical machines in parallel

There are m identical machines in parallel. Job j requires a single operation and may be processed on any one of the m machines or on any one belonging to a given subset. If job j is not allowed to be processed on just any one, but rather only on any one belonging to a given subset, say subset M_j.

3.) Machines in parallel with different speeds

There are m machines in parallel with different speeds; the speeds of machine i is denoted by v_i. The time p_{ij} job j spends on machine i is, assuming it is processed only on machine i, equal to p_j/v_i. This environment is also referred to as uniform machines. If all machines have the same speed, that is $v_i = 1$ for all i and $p_{ij} = p_j$, then the environment is identical to the previous one.

4.) Unrelated machines in parallel

This environment is a generalization or the previous one. There are m different machines in parallel. Machines i can process job on machine i, equal to p_j/v_{ij}. If the speeds of the machines are independent of the jobs, that is, $v_{ij} = v_i$ for all i and j, then the environment is identical to the previous one.

5.) Flow shop

There are m machines in series. Each job has to be processed on each one of the m machines. All jobs have the same routing, that is , they have to be processed first on machine 1, then on machine 2, and so on. After completion on one operate under the first in first out (FIFO) discipline, that is, a job cannot “pass” another while waiting in a queue.

6.) Flexible flow shop

A flexible flow shop is a generalization of the flow shop and the parallel machine environments. Instead of m machines in series there are s stages in series with a number of machines in parallel at each stage. Each job has to be processed first at stage 1, then at stage 2, and so on. A stage functions as a bank of parallel machines; at each stage job j requires only one machine and, usually, any machine can process any job. The queues between the various stages usually operate under the FIFO discipline.

7.) Open shop

There are m machines, Each job has to be processed again on each one of the m machines. However, some of these processing times may be zero. There are no restrictions with regard to the routing of each job through the machine environment. The scheduler is allowed to determine the route for each job, and different fobs may have different routes.

8.) Job shop

In a job shop with m machines, each job has its own route to follow. A distinction is made between job shops where each job visits any machine at most once and job shops where a job may visit a machine more than once.

3.1.3 Constraint-guided heuristic search

In many real-world situations there is not really an objective. Rather, it is required only to generate a feasible schedule that satisfies various constraints and rules. One approach for generating schedules in these situations is referred to in the literature as constraint-guided search. This approach has been very popular among computer scientists and artificial intelligence experts.

Constraint-guided search may be described best through an example. Consider a number of not necessarily identical machines in parallel. A job has to be processed only in one of the machines; for each job there may be a feasible set of machines M_j to choose from. Job j requires a processing $p_j = 1$, $j = 1,\ldots,n$ and has release date r_j and due date d_j. The goal is to find a feasible schedule in which all jobs are processed within their respective time windows. If an optimal schedule has an objective value of 0, then the schedule is feasible in the context described above.

Constraint-guided search may operate according to the following rules. Jobs are scheduled one at a time. When a job is scheduled, it is assigned to a specific time slot on a specific machine, which is still free. At each iteration an unassigned job is selected according to a set of job rules that have between arranged in some priority. The job rules specify whether the particular job actually can be processed on a given machine. The jobs can be ordered according to their criticality or flexibility; the job with the least flexibility is the most critical and has the highest priority. The flexibility of a job can be measured in several ways. First, there may be a flexibility in time, that is, the length of the time window in between the job's release date and due date relative to its processing time. Second, there may be flexibility with regard to the number of machines on which the job may be processed. After a job has been selected for assignment, a set of machines is considered that are arranged according to some priority. The machine also can be ordered in such a way that the machine with the least flexibility has the highest priority; the flexibility of a machine, measured by the number of jobs that can be processed on the machine. With a given job the machines are checked sequentially to see whether an assignment is feasible. When a feasible assignment is discovered, the assignment is made and the procedure repeats itself.

An important concept in constraint-guided search is constraint propagation. The assignment of a particular job to a given time slot on a given machine has implications with regard to the assignment of other jobs on the given machine and on other machines. These implications may point to the violation of hard constraints and may indicate that the associated part of the search space can be disregarded. This is usually referred to as constraint propagation. In constraint-guided heuristic reach, it is important to keep track of constraint propagation.

Constraint-guided search fits very well with the artificial intelligence way of thinking and developing systems. The rules can be programmed easily in languages such as Prolog and LISP. If there are very specific rules, for example, job j is not allowed to be processed on machine I after time t, then such rules can be added very to the list of rules that need to be check

3.1.4 Dispatching Rules

Research in dispatching rules has been active for several decades, and many different rules have been studied in the literature. These rules can be classified in various ways. For example, a distinction can be made between static and dynamic rules. Static rules are not time dependent. They are just a function of the job and/or machine data, for instance, WSPT. Dynamic rules are time dependent. One example of a dynamic rule is the Minimum Slack (MS) first, which orders jobs according to max $(d_j-p_j-t,0)$ and is time dependent. This implies that at some time job j may have a higher priority than job k and at some later time jobs j and k may have the same priority.

A second way of classifying rules is according to the information they are based upon. A local rule uses only information pertaining to either the queue where the job is resident or to the machine where the job is queued.

Dispatching rules are useful for finding a reasonably good schedule with regard to a dingle objective such as the makespan, the total completion time, or the maximum lateness. However, in real life objectives are often more complicated. For example, a realistic objective may be a combination of several basic objectives and also a function of time or a function of the set of jobs waiting for processing. Prioritizing the jobs on the basis of one or two parameters may not lead to acceptable schedules. More elaborate dispatching rules, which take into account several different parameters, can address more complicated objective functions. Some of these more elaborate rules are basically a combination of a number of the elementary dispatching rules. These more elaborate rules are referred to as composite dispatching rules.

3.1.5 Group Technology

Group Technology (GT) is a technology and philosophy which aims at increasing production efficiency by grouping a variety of parts having similarities in shape, dimension, and/or process routing. Group Technology has been primarily used for improving efficiency in manufacturing. In order to further promote standardization of machine products and parts, it is necessary to apply the concept of GT early at the product design stage. The application of GT at the design stage will allow for a positive use of this technique in the total production process from the design stage through the manufacturing of products. However, the use of GT brings about the decrease in the performance or characteristics of machines. In the design decision process, the conflicting relationships between the advantages and the disadvantages must be evaluated.

3.1.5.1 Group Scheduling

Group Scheduling (GS) has been utilized in job shop situations to obtain benefits, which are normally associated with large-scale production in a glow shop. The introduction of GS in a job shop makes it possible to convert the technical order of the jobs on several machines from the random flow of a job shop-scheduling problem. This results in the creation of group scheduling algorithms; however, the group scheduling algorithms have some limitations in their practical application. They are effective in helping production planners determine an optimal sequence among jobs whose production quantities and assigned machine have been decided by the consideration of due dates and machine capacities. In other words, the arrival dates of all jobs are equal to zero or the same. In reality, however, all jobs do not arrive at the same time. Therefore, a preliminary technique is needed that can organize and assign each job to a certain time period according to its due date and machine capacities.

3.1.6 Production line balancing technique

The problem associated with assemble line balancing are to allocate work elements to work stations in such a way to minimize production cost of the product. In single model assembly line there are two optimization problems:

1) To find a minimum number of workstations, subject to its cycle time not to exceeding a given cycle time, i.e. fixed production volume for optimum number of workstations.

2) To find a minimum amount of idle time, subject to its number of work stations not exceeding a given number of workstations, i.e. fixed number of work stations for minimum cycle time.

In mixed model assembly line balancing problem the same objective is also applicable. Assembly line is related to mass production. For mixed model assembly line there are some variety of work elements of different models. So the setup time and cost are very important in mixed-model assembly line. To minimize the huge investment of variety machine or tools it is important to allocate the same work elements for different models to the same work station. If this cost is low then it may be practical to use the line independently of the models.

Practically perfect balance is seldom achieved. To get the optimal result mathematical models are best method. But the computational cost is so high which for most of the case tun downs the use of the models. It is evident from the review of literature that the most of the proposed assembly line balancing techniques are designed to achieve good approximations to solutions for problem 1, i.e. to find the number of work stations of nearly minimum. It is more suitable to apply them to problem 2 by using a search or heuristic process. This is directly applying to problem by varying a given cycle time and searching for nearly minimum cycle time subject to a given number of workstations.

3.1.7 Performance Measures

Several performance criteria to evaluate the scheduling rules were investigated in 1990. It has been evident that scheduling is a multi-criteria optimization problem. However, most of the research has been concerned with individual measures of performance. The more commonly used criteria may be classified as follows:

1) Time based measures

Mean and variance of flow time per job.

Mean and variance of flow time per operation.

Mean waiting time.

Machine idle time.

2) Work in process measures

Average number of jobs in queue

Value of work in process ($WIP)

3) Due date related measures.

Mean tardiness (MT).

Conditional mean tardiness (CMT)

Proportion of jobs tardy (PT)

Mean lateness

Number of jobs tardy

Maximum lateness

4) Cost based measures

Cost of idle machines

Cost of long promises

Cost of carrying work in process inventory

Total cost per job

Average value added in queue.

3.2 Literature Review

WAKILPOOR [38] proposed a model for seeking an optimal solution to N x M job shop scheduling problems in which each job had a number of operations to be performed on the specified machines in an order which might not be the same for all jobs. The model was formulated in the General Purpose Simulation System V (GPSSV) language. FORTRAN subroutines were used to obtain Gantt chart as graphical representation and tables of the results from the model. Furthermore, a case study of a manufacturing firm was conducted in order to illustrate practical applicability of the model.

PLOYPARICHAROEN [32] used Mathematical Programming in the production planning by selecting linear programming as Simplex Method. He found the optimal solution, which had minimized total variable cost in the production. The parameters were about demand, capacity and inventory. Moreover he set the index due to the difficulty of each process and the skill of each worker. He also used Simplex Method and the equation was about assigning the jobs to the suitable skill workers that got the lowest index. These two main information were used to plan the production.

SHUN FUNG [12] focused on Group Technology as applied to the production management to critically evaluate the various models and approach on conventional and group scheduling, thereby highlighting the advantages and limitations of each. And the optimal or near-optimal schedule to perform the n groups of jobs in multistage machines was found.

AHMED ABU CENNA [11] developed an algorithm to consider a problem regarding bicriterion scheduling with parallel machines by the program written in FORTRAN and run on UNIX system. Five different dispatching rules, Shortest Processing Time (SPT-rule), Earliest Due Date (EDD-rule), Minimum Slack Time (MST-rule), Largest Processing Time (LPT-rule) and sequence given by the algorithm of Wassenhove and Gelders has been compared and Wassenhove and Gelders algorithm gave the best result. The developed algorithm was also applied to a real life situation to illustrate and justify the validity of the methodology.

LOUDTRAGULNGAM [25] studied for establishing a production-planning unit, and to set up production planning system for an illustrated Dyeing factory. He used a computer program "Master Production Schedule (MPS)" for the function of this unit, which consist of coordination with others units in the organization, planning systems, and production schedules. The setting up of this unit will enable to devise the organizational chart to fit into modern management and increase the productivity by using information system and arrangement of production schedules with the aid of a microcomputer.

MATTA [26] performed the study initially motivated by the production planning of various types of Tiles by a tile manufacturing company. It uses the process industries plan production which applies the time – phase forward scheduling. The specific scheduling method is the desegregated Capacity – Oriented Production Scheduling. This technique was used where product setups on production line can be scheduled between consecutive period with changeover cost, but without production loss.

SHAHEEN AHMED [1] investigated seven commonly used favorite dispatching rules on a real life job shop environment through simulation. Several practical issues like sequence dependent set-up time, transit time, loading and uploading time, machine breakdown and repair time etc. has been incorporated in the simulation program which performed using the special simulation language SIMAN. Two types of due date assignment rule have also been applied to assess the sensitivity of the rules on due date estimation method. Based on the simulation results it was possible to specify in order of merit the dispatching decision rules most appropriate to BMTF.

ZHAO XIA [40] presented Goal Programming Approach used in developing a production-planning model, which was aimed at the application in a job shop system with bottleneck machines. The model was formulated with two main functions, which were determining the bottleneck machines and finding out a feasible plan. Sequential Linear Goal Program Algorithm through XA package was used to solve the model. The results showed that the proposed model is efficient in such a production system and the solution was very sensitive to the goal level and the goal structures. And the analysis of the result lead to the identification of the bottleneck machine that restricts productivity, for getting the goals, the bottleneck should be properly managed.

HOSSAIN [22] studied an analysis of existing assembly line for manufacturing of different models of motor cycle and redesign the assemble line using Production Line Balancing Techniques. This study is for redesigning the mixed model assembly line keeping the number of workstation constant using the expected processing time concept of different work elements. The existing line was analyzed and redesigned of assembly line was done by applying COMSOAL algorithm. Though analyses of different performance measures, it was found that the existing line was not efficient in all respect and recommended the better assembly line by using the expected time concept.

PINEDO [30, 31] presents method, which originally conceives, for an assembly system used for the insertion of components in printed circuit boards (PCBs). In this assembly system, it is the flow shop manufacturing with a number of stages in series with a number of machines in parallel at each stage. A job, which in this environment is often equivalent to a batch, needs to be processed in each stage on only one machine. The buffers at each stage may have limited capacity. If the buffer is full, then the material handling system must come to a standstill.

Scheduling of flexible flow shop with limited buffers and bypass has two main objectives: the maximization of throughput and the minimization of work in process. With the goal of maximizing the throughput an attempt is made to minimize the makespan. Because the amount of buffer space is limited, it is recommended to minimize the work in process to reduce blocking probabilities. This method consists of three phases:

1. The machine allocation phase.
2. The sequencing phase.
3. The release timing phase.

AMATAYAKUL [2] used a kind of index which was calculated from the amount of order, the inventory and the safety stock (Index = Inventory / Order + Safety Stock) to schedule the production. This is scheduled by hands and shown with Gantt chart.

OUHJITTI [28] established the database management system in the software program as CLIPPER. It classified the data in three Modules.

1. Inventory Management Module
2. Production Planning Information Module
3. Production Planning Module

This software was an interactive processing and shown the production plan in The Gantt chart.

EILION [14] proposed the study, which was related to a chemical plant consisting of four machines and five products were produced. Each machine can produce only one product at a time, but not all are capable of producing the entire product. The purpose of the study is to compare alternative model for determining a production schedule to maximize profit in relation to the total costs incurred. This study compares two production scheduling method, the first is based on linear programming and the second is on a multi-product batch scheduling approach. The simulation technique, that runs the production for the two methods, was tested the results for the comparison.

LIMPAWATTANAPHUM [24] presented the model for scheduling mold machining process which concerned with a job manufacturing system to minimize makespan, improve machine utilization and code computer program for the use of the model. The Giffler and Thomson algorithm with alone SPT or MWKR dispatching rule was considered and modified to solve the problems. For finding the optimal schedule, the comparison between the results from the algorithm and the existing procedure showed that the use of the algorithm was better than that of the existing procedure.

SUWANRUJI [36] focused on selecting the efficient job routing in the flexible manufacturing production. She used an application of Fuzzy Logic for decision Making, so she organized the job routing by Fuzzy Analytical Hierarchy Process (Fuzzy AHP). This program calculated the index of each routing from processing time, the amount of work in process in the machines and the probability to finish the jobs before the machines were broken down. And these indexes were used to be information for selecting the job routing in Fuzzy AHD.

BAKER [5] described various multi-agent architectures, including the heterarchical architecture. It reviewed the claimed advantages for multi-agent heterarchies and also described the type of factories that could use this architecture. It surveyed the three common types of factory control algorithms: dispatching algorithms, scheduling algorithms, and pull algorithms. It discussed how many of the algorithms that were popular in current research could be implemented in a multi-agent heterarchy.

HUNG [23] proposed Tabu search algorithm, which was applied for assembly line balancing and evaluated the results with some existing assembly line balancing models after developing mathematical formulation. A single assembly line balancing problem has been considered that processing time of all tasks/jobs is deterministic and the precedence relationship among tasks was satisfied. It was also studied and evaluated the performance of the motorcycle-engine assembly line of M/S Vietnam

Spare Parts Manufacturing Co., Ltd., and proposed improvements in the design of production line of this company.

COOPER, et al. [13] developed Computer-based Decision Support Systems (DSS) to provide a method for piece dyers to ensure top quality and lower costs in pressure-beck operations. This DSS contained two algorithms that interact with each other during the decision process. Output from this interaction guided decision markers at all steps of the machine scheduling and loading process toward optimal solutions to the total dye-order-processing problem.