“EFFICIENT HEURISTICS FOR SCHEDULING TASKS ON A FLOW SHOP ENVIROMENT TO OPTIMIZE MAKESPAN”

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

FOR THE REQUIREMENT FOR THE DEGREE OF

Master of Technology

In

Production Engineering

By

ATUL KUMAR SAHU

Department of Mechanical Engineering

National Institute of Technology

Rourkela – 8

2009

"EFFICIENT HEURISTICS FOR SCHEDULING TASKS ON A FLOW SHOP ENVIROMENT TO OPTIMIZE MAKESPAN"

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

FOR THE REQUIREMENT FOR THE DEGREE OF

Master of Technology

In

Production Engineering

By

ATUL KUMAR SAHU

Under the guidance of

Dr. B.B. BISWAL

Professor, Department of Mechanical Engineering

Department of Mechanical Engineering

National Institute of Technology

Rourkela – 8

2009

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA – 769008

INDIA

CERTIFICATE

This is to certify that the thesis entitled, **"EFFICIENT HEURISTICS FOR SCHEDULING TASKS ON A FLOW SHOP ENVIROMENT TO OPTIMIZE MAKESPAN"** submitted by **Mr. ATUL KUMAR SAHU in** partial fulfillment of the requirement for the award of **Master of Technology** Degree in **Mechanical Engineering** with specialization in **Production engineering** at the National Institute of Technology, Rourkela (Deemed University) is an authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/ Institute for the award of any degree or diploma.

Date:

Prof.B.B Biswal
Mechanical Engineering Department
National Institute of Technology
Rourkela-769008

ACKNOWLEDGEMENTS

No thesis is created entirely by an individual, many people have helped to create this thesis and each of their contribution has been valuable. I express my sincere gratitude and indebtedness to my thesis supervisor, **Dr. B.B.Biswal**, for proposing this area of research, for his kind and valuable guidance in the completion of the present work. His consistent support and intellectual guidance made me energize and to innovate new ideas.

I am grateful to Dr. R.K Sahu, Professor and Head, Mechanical Engineering Department, NIT, Rourkela for his excellent support during my work. I am also thankful to all the faculty members and staff of Mechanical Engineering Department, for providing me support and advice in preparing my thesis work. Last, but not the least I would like to thank my parents and my friends who are involved directly and indirectly in successful completion of the present work.

Atul Kumar Sahu

Roll no.207me205

M.Tech.(Production Engg.)

Mechanical Engineering Department

NIT Rourkela - 769008

CONTENTS

ABSTRACT

In modern manufacturing the trend is the development of Computer Integrated Manufacturing, CIM technologies which is a computerized integration of manufacturing activities (Design, Planning, Scheduling and Control) produces right products at right time to react quickly to the global competitive market demands. The productivity of CIM is highly depending upon the scheduling of Flexible Manufacturing System (FMS). Shorting the make span leads to decreasing machines idle time which results improvement in CIM productivity. Conventional methods of solving scheduling problems based on priority rules still result schedules, sometimes, with significant idle times. To optimize these, this paper model the problem of a flow shop scheduling with the objective of minimizing the makes pan. The work proposed here deal with the production planning problem of a flexible manufacturing system. This paper model the problem of a flow shop scheduling with the objective of minimizing the makes pan. The objective is to minimize the make span of batch-processing machines in a flow shop. The processing times and the sizes of the jobs are known and non-identical. The machines can process a batch as long as its capacity is not exceeded. The processing time of a batch is the longest processing time among all the jobs in that batch. The problem under study is NP-hard for makespan objective. Consequently, comparison based on Gupta's heuristics, RA heuristic's, Palmer's heuristics, CDS heuristics are proposed in this work. Gantt chart was generated to verify the effectiveness of the proposed approaches.

LIST OF FIGURES

LIST OF TABLES

Chapter-I

INTRODUCTION

CHAPTER-I

INTRODUCTION

1.1 Introduction:

Flexible Manufacturing System (FMS) is an automated manufacturing system which consists of group of automated machine tools, interconnected with an automated material handling and storage system and controlled by computer to produce products according to the right schedule. Manufacturing scheduling theory is concerned with the right allocation of machines to operations over time. FMS scheduling is an activity to select the right future operational program or diagram of an actual time plan for allocating competitive different demands of different products, delivery dates, by sequencing through different machines, operations, and routings for the combination of the high flexibility of job shop type with high productivity of flow-shop type and meeting delivery dates.

FMS Scheduling system is one of the most important information-processing subsystems of CIM system. The productivity of CIM is highly depending upon the quality of FMS scheduling. The basic work of scheduler is to design an optimal FMS schedule according to a certain measure of performance, or scheduling criterion. This work focuses on productivity oriented-make span criteria. Make span is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand.

The inherent efficiency of a flexible manufacturing system (FMS) combined with additional capabilities, can be harnessed by developing a suitable production plan. Machine scheduling problems arises in diverse areas such as flexible manufacturing system, production planning, computer design, logistics, communication etc. A common feature of many of these problems is that no efficient solution algorithm is known yet for solving it to optimality in polynomial time.

The classical flow shop scheduling problem is one of the most well known scheduling problems. Informally the problem can be described as follows:
There are set of jobs and a set of machines. Each job consists of chain of operation, each of which needs to be processed during an uninterrupted time period of a given length on a given machine. Each machine can process at most one operation at a time. A schedule is an allocation of operations to time intervals of the machines. The problem is to find the schedule of minimum length. This work try to minimize the make span of batch-processing machines in a flow shop. The processing times and the sizes of the jobs are known and non-identical. The machines can process a batch as long as its capacity is not exceeded. The processing time of a batch is the longest processing time among all the jobs in that batch. The problem under study is NP-hard for makespan objective. Consequently, comparison based on Gupta's heuristics, RA heuristic's, Palmer's heuristics, CDS heuristics are proposed. Gantt chart was generated to verify the effectiveness of the proposed approaches.

1.2 Sequencing and Scheduling:

Sequencing is a technique to order the jobs in a particular sequence. There are different types of sequencing which are followed in industries such as first in first out basis, priority basis, job size basis and processing time basis etc. In processing time basis sequencing for different sequence, we will achieve different processing time. The sequence is adapted which gives minimum processing time.
By Scheduling, we assign a particular time for completing a particular job. The main objective of scheduling is to arrive at a position where we will get minimum processing time.

1.3 Types of Scheduling:

Basically there are three types of scheduling:

1.3.1 Single Machine Schedule:

Here we arrange the order of jobs in a particular machine. We achieve the best result when the jobs are arranged in the ascending order of their processing time i.e. the job having least processing time is put first in sequence and processed through the machine and the job having maximum processing time is put last in sequence.

1.3.2 Flow Shop Scheduling:

It is a typical combinatorial optimization problem, where each job has to go through the processing in each and every machine on the shop floor. Each machine has same sequence of jobs. The jobs have different processing time for different machines. So in this case we arrange the jobs in a particular order and get many combinations and we choose that combination where we get the minimum make span.

1.3.3 Job Shop Scheduling:

It is also a typical combinatorial optimization problem, but the difference is that, here all the jobs may or may not get processed in all the machines in the shop floor i.e. a job may be processed in only one or two machines or a different job may have to go through the processing in all the machine in order to get completed. Each machine has different sequence of jobs. So it is a complex web structure and here also we choose that combination of arrangements that will be giving the least make span.

1.4 Significance of Work:

Establishing the timing of the use of equipment, facilities and human activities in an organization can:

1. Determine the order in which jobs at a work center will be processed.
2. Results in an ordered list of jobs

3. Sequencing is most beneficial when we have constrained capacity (fixed machine set; cannot buy more) and heavily loaded work centers
4. Lightly loaded work centers = no big deal (excess capacity)
5. Heavily loaded

a) Want to make the best use of available capacity.
b) Want to minimize unused time at each machine as much as possible.

1.5 Parameters of the Work:

1. Average job flow time

a) length of time (from arrival to completion) a job is in the system, on average
b) Lateness
c) average length of time the job will be late (that is, exceed the due date by)
d) Makespan
e) total time to complete all jobs
f) Average number of jobs in the system
g) measure relating to work in process inventory
h) Equals total flow time divided by make span.

1.6 Objective:

1. To deal with the production planning problem of a flexible manufacturing system. I model the problem of a flow shop scheduling with the objective of minimizing the makespan.
2. To provide a schedule for each job and each machine. Schedule provides the order in which jobs are to be done and it projects start time of each job at each work center.
3. To select appropriate heuristics approach for the scheduling problem through a comparative study.
4. To solve FMS scheduling problem in a flow-shop environment considering the comparison based on Gupta's heuristics, RA heuristic's,

Palmer's heuristics, CDS heuristics are proposed. Gantt chart was generated to verify the effectiveness of the proposed approaches.

My objective of scheduling can yield:

1. Efficient utilization …

a) staff
b) equipment
c) facilities

2. Minimization of …

a) customer waiting time
b) Inventories.
c) Processing time.

1.7 Organization of the thesis:

The thesis describing the present research work covers six chapters. Chapter-II describes several diverse streams of literature on the optimization of various scheduling problems regarding single machining, flow shop and job shop. Chapter–III describes the optimization methods of scheduling for the efficient utilization of staff, equipment, facilities utilization with the objective of minimizing customer waiting time, Inventories, Processing time. Chapter-IV describes various proposed methods of optimization for flow shop scheduling problem. Chapter-V deals with the results and discussion for the problem. Chapter-VI concludes with implication and a suggestion for the extension of the study.

1.8 Summary:

Sequencing is a technique to order the jobs in a particular sequence. There are different types of sequencing which are followed in industries such as first in first out basis, priority basis, job size basis and processing time basis etc. In processing time basis sequencing for different sequence, we will achieve different processing time. The sequence is adapted which gives minimum processing time.

By Scheduling, we assign a particular time for completing a particular job. The main objective of scheduling is to arrive at a position where we will get minimum processing time. The various assignment associated with staff, equipment, facilities utilization and customer waiting time, Inventories , Processing time minimization are:

1. Need to assign a job to a machine/resource to process it.
2. Loading.
3. Need to decide how many jobs can be assigned to each machine.
4. Scheduling.
5. Need to decide on a starting time for each job at each workstation.
6. Sequencing.
7. Need to order processing of individual jobs at each workstation.

The work associated with staff, equipment, facilities utilization and customer waiting time, Inventories, Processing time minimization are presented.

Chapter-II

LITERATURE SURVEY

CHAPTER-II

LITERATURE SURVEY

2.1 Introduction:

Another important consideration is the choice of appropriate criteria for scheduling. Although the ultimate objective of any enterprise is to maximize the net present value of the shareholder wealth, this criterion does not easily lend itself to operational decision-making in scheduling. Some researchers are developing methodologies which take revenue and cost effects of schedules into consideration. Researchers and practitioners have so far used operational surrogates that influence costs and revenues. These include: number of parts tardy, average tardiness, weighted tardiness, throughput (this is a revenue-influencing surrogate), as well as average number of parts in the system, machine utilization, and work-in- process inventory, for example. Analyses of these surrogates indicate that a scheduling procedure which does well for one criterion is not necessarily the best for some other. For example, attempts to reduce mean tardiness can lead to an increase in mean flow time. Minimizing make span can result in higher mean flow time.

2.2 Previous Literatures:

Felix T.S. Chan et.al [1] developed optimization models for solving distributed FMS scheduling problems subject to maintenance: [Genetic algorithms approach]. The authors have made an attempt to optimize the following things during the cycle in the work:

a) Allocation of jobs to suitable factories
b) Determination of the corresponding production scheduling in each factory.

Their objective is to maximize the system efficiency by finding an optimal planning for a better collaboration among various processes. They proposed a genetic algorithm with dominant genes (GADG) approach to deal with distributed flexible manufacturing system (FMS) scheduling problems subject to machine maintenance constraint.

KedadSidhoum et.al [2] developed optimization models for lower bounds for the earliness–tardiness scheduling problem on parallel machines with distinct due dates ,considering the parallel machine scheduling problem in which the jobs have distinct due dates with earliness and tardiness. They considered the earliness–tardiness problem in a parallel machine environment. Their objective is related with the parallel machine scheduling problem in which the jobs have distinct due dates with earliness and tardiness costs. The main objective of their model is to optimize tardiness.

Sharafali et.al [3] developed optimization models for production scheduling in a flexible manufacturing system under random demand. They considered the problem of production scheduling in a Flexible Manufacturing System (FMS) with stochastic demand.

Ecker and Gupta [4] developed optimization models for scheduling tasks on a flexible manufacturing machine to minimize tool change delays. They considered the problem of scheduling a given set of precedence constraint tasks on a flexible machine equipped with a tool magazine where each task requires exactly one of the tools during its execution changing from one tool to another requires a certain amount of time that depends on the pair of tools being exchanged. They present a new algorithmic approach for general task precedence relations when it is desired to sequence the tasks in such a way that the total time required for tool changes is minimized. The approaches they used are

a) A heuristic algorithm
b) Simulation.

Lia et.al [5] developed efficient composite heuristics models for total flow time minimization in permutation flow shops considering flow shops with total flow time. Flow shop scheduling is an important manufacturing system widely existing in industrial environments. A flow shop can be described as *n* jobs being processed on *m* machines and each job having the same machine order .Thus they proposed a composite heuristics model to minimize the flow time.

Drstvensek et.al [6] developed a model of data flow in lower CIM levels considering ,models of production automation based on the idea of five levels CIM hierarchy where the technological database (TDB) represents a backbone of the system . Their main objective is to provide a common environment where the evaluation of a given general order and later composition of work orders, and designation of production resources could be done automatically under the operator's supervision.

Ezedeen Kodeekha,(Department of Production, Informatics, Management and Control Faculty of Mechanical Engineering Budapest University of Technology and Economics) [7] developed "A new method of FMS scheduling using optimization and scheduling". Conventional methods of solving scheduling problems such as heuristic methods based on priority rules still result schedules, sometimes, with significant idle times. To optimize these, the author proposes a new high quality scheduling method. He uses multi-objective optimization and simulation method .The method is called **"**Break and Build Method**"**, BBM.

Clarence H Martin [8] developed a hybrid genetic algorithm/mathematical programming approach to the multi-family flow shop scheduling problem with lot streaming. He developed a new aspect of the problem related with sublots, the size of sublots and the interleaving of sublots from different jobs in the processing sequence. His approach allows for quicker movement of items through the manufacturing facility that is a key element of synchronous manufacturing. Of course, lot streaming raises new issues such as determining the number of sublots and their sizes.

Chia and Lee [9] developed the total completion time problem in a permutation flow shop with a learning effect. The concept of learning process plays a key role in production environments. Their objective is to minimize the sum of completion times or flow time .They used the dominance rule and several lower bounds to speed up the search for the optimal solution.

Koulamas and Kyparisis [10] developed single-machine scheduling with waiting-time-dependent due dates in which due dates are linear functions of the job waiting-times. They construct an optimal sequence and assign the optimal due dates analytically in a single-machine setting when due dates are linear functions of the job waiting-times and their objective is to minimize the maximum job lateness.

Das, et.al [11] developed, Optimization of operation and changeover time for production planning and scheduling in a flexible manufacturing system and deals with the production planning problem of a flexible manufacturing system. They specifically addresses issues of machine loading, tool allocation, and part type grouping with the objective of developing an operation sequencing technique capable of optimizing operation time, non-productive tool change times, and orientation change times when processing a group's design features

Chen and Lee [12] developed a model for Logistics scheduling with batching [LSB] and transportation. Their objective is to minimize the sum of weighted job delivery time and total transportation cost. Since their problem involves not only the traditional performance measurement, such as weighted completion time, but also transportation arrangement and cost, key factors in logistics management,.

Poulos and Zografos [13] developed a model for Solving the multi-criteria time-dependent routing and scheduling problem in a multimodal fixed scheduled network. Their objective is to present the formulation and algorithmic solution for the multi-criteria itinerary planning problem that takes into account the aforementioned features. Their main objective are :

1. formulate the itinerary planning problem as a multi-criteria shortest path problem with intermediate stops in a multimodal time dependent network,
2. present a decomposition scheme for handling the constraint of visiting intermediate locations within specified time windows, and
3. Introduce a dynamic programming based algorithm for solving the individual elementary multi-criteria time-dependent shortest path problems between any pair of sequential intermediate stops .

In addition, they proved that the Basic Unit of Concurrency (BUC) is a set of the executed control flows based on the behavioral properties of the net.

Hamania et.al, [14] developed a model for Reactive mode handling of flexible manufacturing systems. They deal with a new modeling approach for mode handling of flexible manufacturing systems (FMS). Based on a review of the modeling methods and the specification formalisms in the existing approaches, they show that the mutual benefit of functional modeling and synchronous languages is very convenient for mode handling problem.

Hsu, et.al. [15] developed a model for cyclic scheduling for F.M.S.: Modelling and evolutionary solving approach. They concern the domain of flexible manufacturing systems (FMS) and focuses on the scheduling problems encountered in these systems. They have chosen the cyclic behavior to study this problem with the objective to reduce its complexity. This cyclic scheduling problem, whose complexity is NP-hard in the general case, aims to minimize the work in process (WIP) to satisfy economic constraints. They study the problem of FMS control by a predictive approach to compute a cyclic and deterministic schedule.

Sadykov [16] developed a branch-and-check algorithm for minimizing the weighted number of late jobs on a single machine with release dates. He consider the scheduling problem of minimizing the weighted number of late jobs on a single machine .He proposed a branch-and-check algorithm , where a relaxed integer programming formulation is solved by branch-and-bound and infeasible solutions are cut off using infeasibility cuts.

Wu and Zhou [17] developed a model for Stochastic scheduling to minimize expected maximum lateness. They concerned with the problems in scheduling a set of jobs associated with random due dates on a single machine so as to minimize the expected maximum lateness in stochastic environment. This is a difficult problem and few efforts have been reported on their solution.

Rossi and Dini [18] developed Flexible job-shop scheduling with routing flexibility and separable setup times using ant colony optimization method. They propose an ant colony optimization-based software system. Their main objective is to solve FMS scheduling problem in a job-shop environment considering routing flexibility, sequence-dependent setup and transportation time. Routing flexibility leads to the problem of flexible (or multiprocessor) job-shop scheduling (FJS) which extends the classic problem of job-shop scheduling where no alternative machine is present for processing an operation. They concerns two sub-problems:

1. assignment of each operation to one of the alternative machines (assignment sub-problem);
2. Ordering of the operations on each assigned machine (sequencing sub problem), with the aim of optimizing them.

Wang et.al [19] developed FBS-enhanced agent-based dynamic scheduling in FMS. The main objective is to show the feasibility of the approach and to evaluate the approach via computational experiments. They propose a multi-agent approach integrated with a filtered-beam- search (FBS)-based heuristic algorithm to study the dynamic scheduling problem in a FMS shop floor consisting of multiple manufacturing cells.

Goncalves, et.al [20] developed a genetic algorithm for the resource constrained multi-project scheduling problem. They presents a genetic algorithm for the resource constrained multi-project scheduling problem. The chromosome representation of the problem is based on random keys. They constructed schedules using a heuristic that builds parameterized active schedules based on priorities, delay times, and release dates defined by the genetic algorithm with the objective to optimize the resource constrained multi-project scheduling problem.

Cheng et.al [21] developed a model for Single-machine scheduling of multi-operation jobs without missing operations to minimize the total completion time. They consider the problem of scheduling multi-operation jobs on a single machine to minimize the total completion time. Each job consists of several operations that belong to different families. In a schedule each family of job operations may be processed as batches with each batch incurring a set-up time. A job is completed when all of its operations have been processed. Their objective is to minimize the total completion time.

Teunter et.al [22] developed a model for Multi-product economic lot scheduling problem with separate production lines for manufacturing and remanufacturing. They study the economic lot scheduling problem with two production sources, manufacturing and remanufacturing, for which operations are performed on separate, dedicated lines. Their objective is to develop an exact algorithm for finding the optimal common- cycle-time policy. Their algorithm combines a search for the optimal cycle time with a mixed integer programming (MIP) formulation of the problem given a fixed cycle time.

Tang and Gong [23] developed a hybrid two-stage transportation and batch scheduling problem They study the coordinated scheduling problem of hybrid batch production on a single batching machine and two-stage transportation connecting the production, where there is a crane available in the first-stage transportation that transports jobs from the warehouse to the machine and there is a vehicle available in the second-stage transportation to deliver jobs from the machine to the customer. Their objective is to minimize the sum of the make span and the total setup cost.

Tseng and Liao [24] developed a discrete particle swarm optimization for lot-streaming flow shop scheduling problem. They consider an n-job, m-machine lot-streaming problem in a flow shop with equal-size sublots where their objective is to minimize the total weighted earliness and tardiness.

Chang et.al [25] developed a hybrid genetic algorithm with dominance properties for single machine scheduling with dependent penalties. They developed a hybrid genetic algorithm to solve the single machine scheduling problem with the objective to minimize the weighted sum of earliness and tardiness costs.

Cheng and Lin [26] developed Johnson's rule, composite jobs and the relocation problem. They consider two-machine flow shop scheduling with the objective to minimize make span. Johnson's rule for solving this problem has been widely cited in their work. They introduce the concept of composite job, which is an artificially constructed job with processing times such that it will incur the same amount of idle time on the second machine as that incurred by a chain of jobs in a given processing sequence.

Seong-Jong Joo et.al [27] developed a model for Scheduling preventive maintenance for modular designed components: A dynamic approach. Their objective is to develop a dynamic approach for scheduling preventive maintenance at a depot with the limited availability of spare modules and other constraints. They proposed a backward allocation algorithm and applied it to scheduling the preventive maintenance of an engine module installed in T-59 advanced jet trainers in the Republic of Korea Air Force.

He and Hui [28] developed a rule-based genetic algorithm for the scheduling of single-stage multi-product batch plants with parallel units. They present a genetic algorithm-based on heuristic rules for large-size SMSP. In their work, the size of the problems was enlarged, and the problems are first solved by MILP methods and then a random search (RS) based on heuristic rules has been proposed.

Chen and Askin [29] developed a model for Project selection, scheduling and resource allocation with time dependent returns. They formulate and analyze the joint problem of project selection and task scheduling. They study the situation where a manager has many alternative projects to pursue such as developing new product platforms or technologies, incremental product upgrades, or continuing education of human resources. Project return is assumed to be a known function of project completion time. Resources are limited and renewable. Their objective is to maximize present worth of profit.

Janiak et.al [30] developed a scheduling problem with job values given as a power function of their completion times. They deals with a problem of scheduling jobs on the identical parallel machines, where job values are given as a power function of the job completion times. Minimization of the total loss of job values is the main objective of their work.

Grzegorz Waligo´ra [31] developed a model named Tabu search for discrete–continuous scheduling problems with heuristic continuous resource allocation. His objective is to minimize the make span .He considered problems of scheduling non-preempt able, independent jobs on parallel identical machines under an additional continuous renewable resource.

Valls et.al [32] developed skilled workforce scheduling in Service Centre's. Their main objective with SWPSP is to quickly obtain a feasible plan of action satisfying maximum established dates and timetable worker constraints. Secondary their objectives deal with the urgency levels imposed by the criticality task levels, to obtain well-balanced worker workloads and an efficient assignment of specialists to tasks.

Tiwari et.al [33] developed a model for scheduling projects with heterogeneous resources to meet time and quality objectives. Their approach guides decision-making concerning which workers to cross-train in order to extract the greatest benefits from worker-flexibility. They demonstrates how the output of the model can be used to identify bottlenecks (or critical resource skills), and also demonstrates how cross-training the appropriately skilled groups or individuals can increase throughput.

P.Y. Fung [34] developed a model for Lower bounds on online deadline scheduling with preemption penalties. He generalizes and improve results of online preemptive deadline scheduling with preemption penalties. He consider both the preemption-restart and the preemption resume models, and give new or improved lower bounds on the competitive ratio of deterministic online algorithms. In many cases his proposed bounds are optimal when the job deadlines are tight.

Tang and Zhao [35] developed a model for scheduling a single semi-continuous batching machine. They address a new problem, called semi-continuous batch scheduling, which arises in the heating-operation of tube billets in the steel industry. Each heating furnace can be regarded as a semi continuous batching machine, which can handle up to *C* jobs simultaneously. Their objectives are to schedule jobs on the machine so that the make span and the total completion time are minimized.

Eren and Guner [36] developed a bicriteria flow shop scheduling problem with a learning effect. They consider learning effect in a two-machine flow shop. Their objective is to find a sequence that minimizes a weighted sum of total completion time and make span.

Wu and Zhou [37] developed a model for Stochastic scheduling to minimize expected maximum lateness. They concerned with the problems in scheduling a set of jobs associated with random due dates on a single machine so as to minimize the expected maximum lateness in stochastic environment. This is a difficult problem and few efforts have been reported on their solution.

Yang and Geunes [38] developed a predictive–reactive scheduling model on a single resource with uncertain future jobs. Their objective is to minimize the sum of expected tardiness cost, schedule disruption cost, and wasted idle time cost.

Mosheiov and Sarig [39] developed a model for Due-date assignment on uniform machines. Their objective is to find the job schedule and the due-date that minimize a linear combination of all three (earliness, tardiness and due-date) cost factors.

Biskup and Herrmann [40] developed a model for Single-machine scheduling against due dates with past-sequence-dependent setup times. Their objective is to minimize the due date.

Chena et.al [41] developed a model for dense open-shop schedules with release times. They study open-shop scheduling problems with job release times. Their objective is to minimize the make span. Dense schedules, easy to construct, are often used as approximate solutions. Performance ratios of the make spans from dense schedules and that of the optimal schedule of the problem are used to evaluate the quality of approximate schedules.

2.3 Summary

Scheduling is an activity to select the right future operational program or diagram of an actual time plan for allocating competitive different demands of different products, delivery dates, by sequencing through different machines, operations, and routings for the combination of the high flexibility of job shop type with high productivity of flow-shop type and meeting delivery dates. The different types of approaches to the Manufacturing scheduling theory have been reported. Here these approaches show their various advantages and disadvantages to the development of new design problem. Taking the old approach in to consideration the development of new approaches conceptualized through these literatures.

Chapter-III

DESCRIPTION OF METHODS

CHAPTER-III

DESCRIPTION OF METHODS

3.1 Introduction of Sequencing Methods

1. Shortest processing time: select the job having the least processing time
2. Earliest due date: select the job that is due the soonest.
3. First-come, first served: select the job that has been waiting the longest for this workstation.
4. First-in-system, first-served: select the job that has been in the shop the longest.
5. Slack per remaining operations: select the job with the smallest ratio of slack to operations remaining to be performed.

3.2 Some Methods for Scheduling and Sequencing

3.2.1 Single machine scheduling methods:

1. Shortest processing time rule(SPT)
2. Earliest due date rule(EDD)
3. Weighted mean flow time method
4. Naughton's algorithm to minimize the number of tardy jobs
5. Hodgson's algorithm to minimize tardiness.

3.2.2 Flow shop scheduling methods:

1. Two-Machine Flow-shop Problem

a) Johnson's Rule

b) Kusiak's Rule

2. Heuristics for general *m*-Machine Problems

a) Palmer's Heuristic Algorithm

b) Gupta's Heuristic Algorithm

c) CDS Heuristic Algorithm

d) RA Heuristic Algorithm

3.2.3 Job shop scheduling methods:

1. JSP Mathematical / Graph Models

a) Integer Programming Model

b) Linear Programming Model

c) Disjunctive Graph Model

2. Conventional Heuristics for JSP

a) Priority Dispatching Heuristics

b) Shifting Bottleneck Heuristic

c) Randomized Dispatching Heuristics

3.3 Performance Measure Used to Decide the Best Optimal Solution

Average WIP inventory

1. Makespan (total time to finish processing)
2. Due date (lateness, earliness, and tardiness)
3. Machine Utilization
4. Labor Utilization

3.4 Assumptions for Solving Scheduling Problems

1. Set of Jobs to Schedule

a) Typically assume that our set of jobs is fixed

2. Time

a) Need to assume times are known,

b) Usually assume times are fixed and independent of processing order or activities that take place elsewhere in the factory

c) Quality

d) Assume we never produce a bad part

e) Machines
f) Assume we never have breakdowns.

"Optimal" scheduling methods' assumptions can be violated in many ways:

Variability in

a) Setup times
b) Processing times
c) Interruptions
d) Changes in the set of jobs.

3.5 Performance Evaluation Criteria for Scheduling Methods

Choice of sequencing methodology to choose is dependent on the performance evaluation criteria to be applied:

1. Total job completion time.
2. Avg job completion time.
3. Avg job waiting time.
4. Avg job lateness.
5. Avg number of jobs in the system.
6. Avg number of jobs waiting.
7. Set-up costs.
8. In-process inventory costs.

3.6 Goals of Scheduling Methods

Efficient utilization …

1. Staff.
2. Equipment.
3. Facilities.

Minimization of …

1. Customer waiting time.
2. Inventories.
3. Processing time.

3.7 Tools for Scheduling

Gantt Charts are used as a visual aid for loading and scheduling. The Gantt schedule can illustrate the relationship between work activities having duration, events without duration that indicate a significant completion, that represent major achievements or decision points.

3.8 Approaches to Scheduling

1. Forward scheduling
   a) scheduling ahead from a point in time (e.g., now)
   b) Useful to answer the question "How long will it take to complete this job?"
2. Backward scheduling
   a) scheduling backward from a future due date
   b) useful to answer the questions:
      - "Can we complete this job in time?"
      - "When is the latest we can start this job and still complete it by the due date?"

3.9 Summary

By Scheduling, we assign a particular time for completing a particular job. The main objective of scheduling is to arrive at a position where we will get minimum processing time. Scheduling is the process by which you look at the time available to you, and plan how you will use it to achieve the goals you have identified. By using a schedule properly, you can:

1. Understand what you can realistically achieve with your time;
2. Plan to make the best use of the time available;
3. Leave enough time for things you absolutely must do;
4. Preserve contingency time to handle 'the unexpected and
5. Minimize stress by avoiding over-commitment to others.

Chapter-IV

METHODS USED

CHAPTER-IV

METHODS USED

4.1 Introduction

Scheduling is the process by which you look at the time available to you, and plan how you will use it to achieve the goals you have identified. Some priority rules that are tested for scheduling are FCFS, SPT, LPT, and PR/TR (assign the highest priority to the part whose proportion of required output is lagging behind most). Machine utilization and production rate are used as the criteria for evaluating part input and scheduling procedures.

4.2 Methodology

Manufacturing scheduling theory is concerned with the right allocation of machines to operations over time. The basic work of scheduler is to design an optimal FMS schedule according to a certain measure of performance, or scheduling criterion. This work focuses on productivity oriented-make span criteria. Make span is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand.The approach used in this work were the comparison based on four heuristic algorithm namely Gupta’s algorithm , CDS algorithm , RA algorithm and Palmer’s algorithm were proposed. Here the main objective is to find the efficient heuristics algorithm for minimizing the make span. In this work hierarchical approach were used to determine the optimal make span criteria.

4.3 Problem Statement

There is a flow shop scheduling problem in which all the parameters like processing time, due date, refixturing time, setup time are given. The value of the make span of batch-processing machines in a flow shop based on comparison of Gupta's heuristics, RA heuristic's, Palmer's heuristics, CDS heuristics are proposed. Analytic solutions in all the heuristics are investigated. Gantt chart were generated to verify the effectiveness of the proposed approaches. Here the heuristics approach for planning problems are proposed which provides a way to optimize the make span which is our objective function.

4.4 Flow Shop Scheduling

It is a typical combinatorial optimization problem, where each job has to go through the processing in each and every machine on the shop floor. Each machine has same sequence of jobs. The jobs have different processing time for different machines. So in this case we arrange the jobs in a particular order and get many combinations and we choose that combination where we get the minimum make span.

In an m-machine flow shop, there are m stages in series, where there exist one or more machines at each stage. Each job has to be processed in each of the m stages in the same order. That is, each job has to be processed first in stage 1, then in stage 2, and so on. Operation times for each job in different stages may be different. We classify flow shop problems as:

1. Flow shop (there is one machine at each stage).
2. No-wait flow shop (a succeeding operation starts immediately after the preceding operation completes).
3. flexible (hybrid) flow shop (more than one machine exist in at least one stage)and
4. Assembly flow shop (each job consists of specific operations, each of which has to be performed on a pre-determined machine of the first stage, and an assembly operation to be performed on the second stage machine).

4.5 Flow Shop Scheduling Methods

Two-Machine Flow-shop Problem

1. Johnson's Rule.

2. Kusiak's Rule.

Heuristics for general *m*-Machine Problems

1. Palmer's Heuristic Algorithm.
2. Gupta's Heuristic Algorithm.
3. CDS Heuristic Algorithm.
4. RA Heuristic Algorithm.

4.6 General Description

1. There are *m* machines and *n* jobs.
2. Each job consists of *m* operations and
    a) each operation requires a different machine
3. *n* jobs have to be processed in the same sequence on *m* machines.
4. Processing time of job *i* on machine *j* is given by t_{ij}
    a) (i =1… n ; j =1,…,m)
5. Make span: find the sequence of jobs minimizing the maximum flow time.

4.7 Main Assumptions

1. Every job has to be processed on all machines in the order (j = 1,2,….,m).
2. Every machine processes only one job at a time.
3. Every job is processed on one machine at a time.
4. Operations are not preemptive.
5. Set-up times for the operations are sequence-independent and are included in the processing times.

Operating sequences of the jobs are the same on every machine, and the common sequence has to be determined.

4.8 Three Categories of FSP

1. Deterministic flow-shop scheduling problem:
    - Assume that fixed processing times of jobs are known.
2. Stochastic flow-shop scheduling problem:

- Assume that processing times vary according to chosen probability distribution.

3. Fuzzy flow-shop scheduling problem:
   - Assume that a fuzzy due date is assigned to each job to represent the grade of satisfaction of decision makers for the completion time of the job.

4.9 Two-Machine Flow Shop Problem:

4.9.1 Johnson's Rule:

Johnson's Algorithm:

- An optimal sequence is directly constructed with an adaptation of this result by a one-pass scanning procedure.
- Let *I* denote the job list and let *S* denote the schedule:

Step 1: Let $U = \{j|\ t_{i1} < t_{i2}\}$ and $V = \{j|t_{i1} \geq t_{i2}\}$

Step 2: Sort jobs in *U* with non-decreasing order of t_{i1}

Step 3: Sort jobs in *V* with non-increasing order of t_{i2}

Step 4: An optimal sequence is the ordered set *U* followed by the

Ordered set *V*.

Consider an 8-job problem :(8-job, 2-machine)

Table 4.1: Two-Machine Flow-shop Problem for Johnson's Algorithm:

Job *i*	1	2	3	4	5	6	7	8
$t_{i\,1}$	5	$\underline{2}$	$\underline{1}$	7	$\underline{6}$	$\underline{3}$	7	5
$t_{i\,2}$	$\underline{2}$	6	2	$\underline{5}$	6	7	$\underline{2}$	$\underline{1}$

The solution constructed as follows:

Step 1: Job sets U= {2, 3, 6} and V= {1, 4, 5, 7, 8}

Step 2: Sort jobs in U as follows:

Job i : 3 2 6

$t_{i\,1}$: 1 2 3

Step 3: Sort jobs in V as follows:

Job i : 5 4 7 1 8

$t_{i\,2}$: 6 5 2 2 1

Step 4: An optimal sequence is {3, 2, 6, 5, 4, 7, 1, 8}

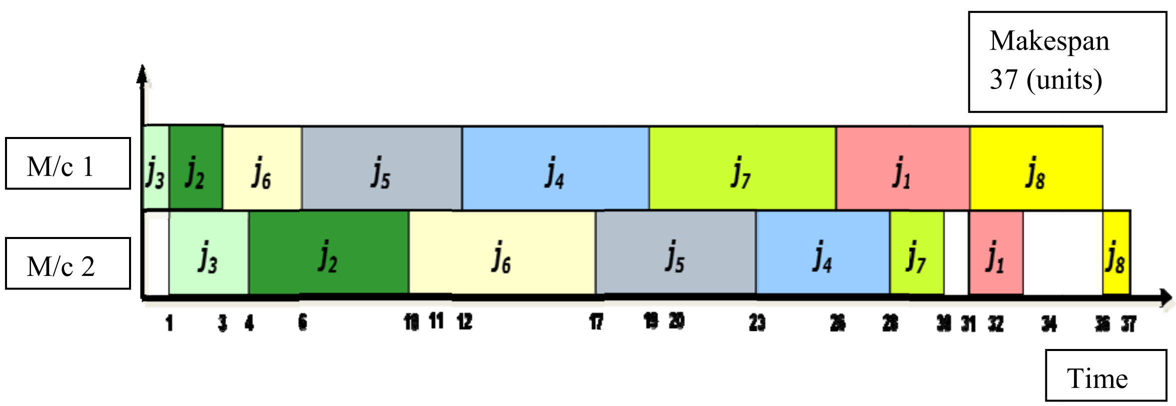

Figure 4.1 Gantt chart for 8-job, 2-machine problem by Johnson's Rule:

4.9.2 Kusiak's Rule:

Kusiak's Algorithm:

Step 1: Set $k = 1, l = n$

Step 2: For each operation, store the shortest processing time and corresponding machine number.

Step 3: Sort the resulting list, including the triplets “Operation number /processing time/machine number” in increasing value of processing time.

Step 4: For each entry in the sorted list:

If machine number is 1, then

(i) set the corresponding operation number in position *k*,

(ii) Set $k = k + 1$

else

(i) set the corresponding operation number in position *k*,

(ii) Set $l = l - 1$

end

Step 5: Stop if the entire list of operations has been exhausted.

Consider an 8-job problem :(8-job, 2-machine)

Table 4.2: Two-Machine Flow-shop Problem for Kusiak’s Algorithm:

Job i	1	2	3	4	5	6	7	8
t_{i1}	5	$\underline{2}$	$\underline{1}$	7	$\underline{6}$	$\underline{3}$	7	5
t_{i2}	$\underline{2}$	6	2	$\underline{5}$	6	7	$\underline{2}$	$\underline{1}$

The solution constructed as follows:

Table 4.3: Solution by Kusiak's Algorithm:

t_{i1}	t_{i2}
5	2
2	6
1	2
7	5
6	6
3	7
7	2
5	1

Job *i*	$t_{(i)}$	*m*
3	1	1
8	1	2
2	2	1
1	2	2
7	2	2
6	3	1
4	5	2
5	6	1

m/c 1 ↓

m/c 2 ↑

An optimal sequence is {3, 2, 6, 5, 4, 7, 1, 8}

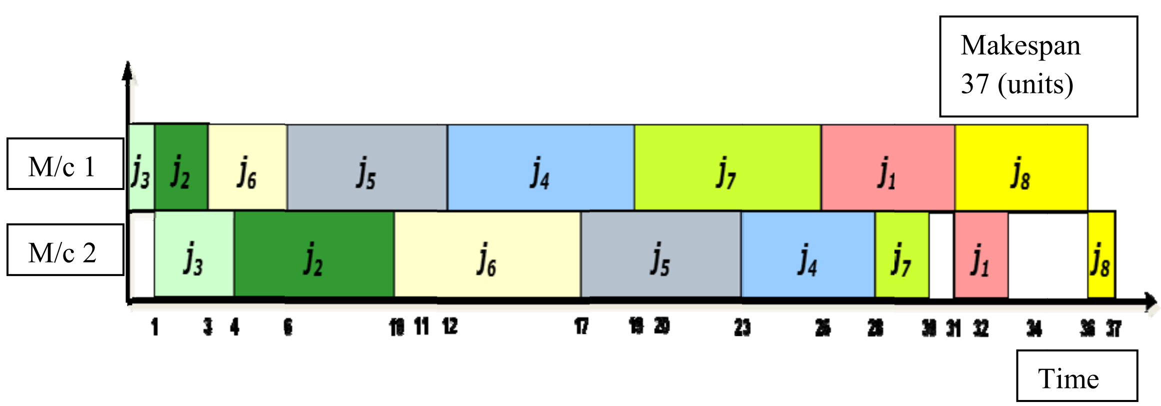

Figure 4.2 Gantt chart for 8-job, 2-machine problem by Kusiak's Rule

4.10 Heuristics for General 3-Machine and 8-Jobs Problems:

1. Palmer's Heuristic Algorithm.
2. Gupta's Heuristic Algorithm.
3. CDS Heuristic Algorithm.
4. RA Heuristic Algorithm.

4.10.1 Palmer's Heuristic Rule:

Algorithm: Palmer's Heuristic:

Procedure: Palmer's Heuristic

Input: job list *i, machine m*;

Output: schedule *S*;

begin

for $i = 1$ to n

for $j = 1$ to m

Calculates $s_i = s_i + (2j - m - 1)t_{ij}$; // step 1:

Permutation schedule is constructed by sequencing the jobs in

Non-increasing order of s_i such as: $s_{i_1} \geq s_{i_2} \geq \ldots \geq s_{i_n}$; // step 2:

end

Output optimal sequence is obtained as schedule S; // step 3:

end.

Consider an 8-job problem :(8-job, 3-machine):

Table 4.4: General 3-Machine and 8- jobs Problems:

Job i	1	2	3	4	5	6	7	8
$t_{i\,1}$	5	2	1	7	6	3	7	5
$t_{i\,2}$	2	6	2	5	6	7	2	1
$t_{i\,3}$	3	4	2	6	1	5	4	7

The solution constructed as follows:

Step 2: set the slope index s_i for job i as:

$$s_1 = -10 + 6 = -4$$
$$s_2 = -4 + 8 = 4$$
$$s_3 = -2 + 12 = 10$$
$$s_4 = -14 + 4 = -10$$
$$s_5 = -12 + 2 = -10$$
$$s_6 = -6 + 10 = 4$$
$$s_7 = -14 + 8 = -6$$
$$s_8 = -10 + 14 = 4$$

Step 3: jobs are sequenced according

$$s_3 \geq s_2 \geq s_6 \geq s_8 \geq s_1 \geq s_7 \geq s_4 \geq s_5$$
$$10 \geq 4 \geq 4 \geq 4 \geq -4 \geq -6 \geq -10 \geq -10$$

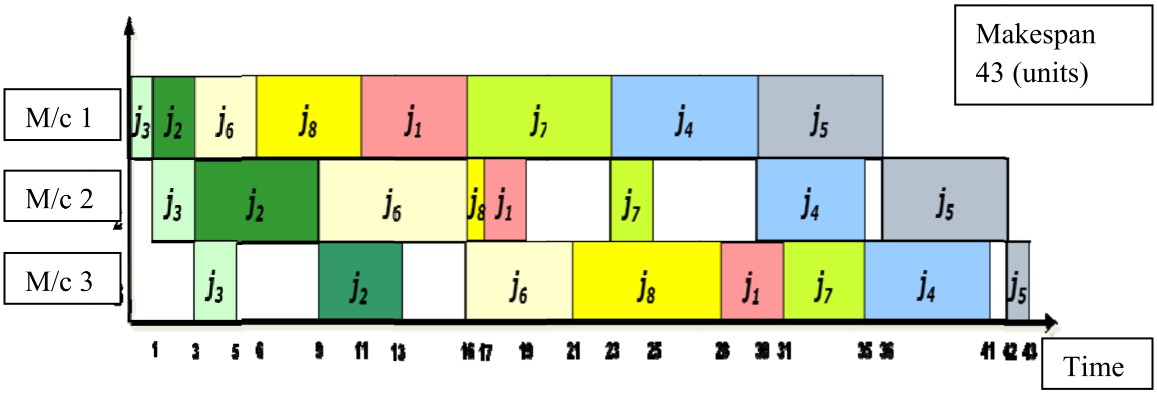

Figure 4.3 Gantt chart for 8-job, 3-machine problem by Palmer's Heuristic Rule:

4.10.2 Gupta's Heuristic Rule:

Algorithm: Gupta's Heuristic:

Procedure: Gupta's Heuristic

Input: job list i, machine m;

Output: schedule S;

begin

for $i = 1$ to n

for $k = 1$ to m-1

if $t_{i1} < t_{im}$ then

$e_i=1$;

else

$e_i = -1$;

calculate $s_i= e_i /\min\{t_{ik} + t_{i,k+1}\}$; // step 1:

end

permutation schedule is constructed by sequencing the jobs in

Non-increasing order of s_i such as: $s_{i_1} \geq s_{i_2} \geq \ldots \geq s_{i_n}$; // step 2:

end

Output optimal sequence is obtained as schedule S. // step 3:

end .

Consider the above 8-job and 3-machine problem:

The solution constructed as follows:

step 1: set the slope index s_i for job i as:

$$s_1 = \frac{-1}{\min\{7,5\}} = -0.2 \qquad s_5 = \frac{-1}{\min\{12,7\}} = -0.143$$

$$s_2 = \frac{1}{\min\{8,10\}} = 0.125 \qquad s_6 = \frac{1}{\min\{10,12\}} = 0.1$$

$$s_3 = \frac{1}{\min\{3,8\}} = 0.333 \qquad s_7 = \frac{-1}{\min\{9,6\}} = -0.166$$

$$s_4 = \frac{-1}{\min\{12,7\}} = -0.143 \qquad s_8 = \frac{1}{\min\{6,8\}} = 0.166$$

Step 2: jobs are sequenced according:

$$s_3 \geq s_8 \geq s_2 \geq s_6 \geq s_4 \geq s_5 \geq s_7 \geq s_1$$

$$0.333 \geq 0.166 \geq 0.125 \geq 0.1 \geq -0.143 \geq -0.143 \geq -0.166 \geq -0.2$$

Step 3 output optimal sequence is {3, 8, 2, 6, 4, 5, 7, 1}

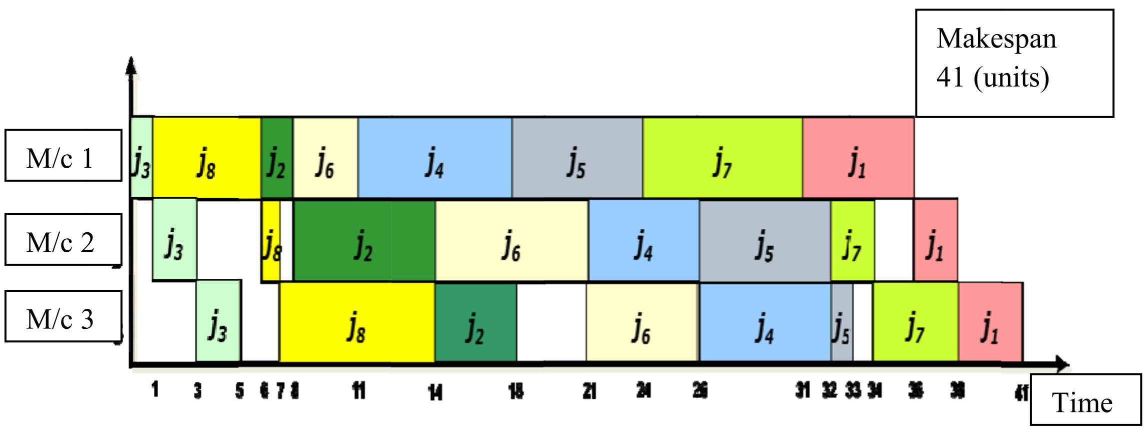

Figure 4.4 Gantt chart for 8-job, 3-machine problem by Gupta's Heuristic Rule:

4.10.3 CDS Heuristic Rule:

Algorithm: CDS Heuristic:

Procedure: CDS Heuristic

Input: job list *I*, machine *m;*

Output: schedule *S*;

begin

for $i = 1$ to n

for $j = 1$ to $m-1$

$t_{i1}' = t_{i1}' + t_{ij}'$;

for j=m to 2;

$t_{i2}' = t_{i2}' + t_{ij}'$;

end

calculate $U = \{i|\ t_{i1}' < t_{i2}'\}$ and $V = \{i|t_{i1}' \geq t_{i2}'\}$; // step 1

sort jobs in U with non-decreasing order of t_{i1}'; // step 2

sort jobs in V with non-increasing order of t_{i2}'; // step 3

Output optimal sequence is obtained as schedule S by U and V // step 4

end

Consider the above 8-job and 3-machine problem:

The solution constructed as follows:

Table 4.5: Solution by CDS Algorithm:

job i	$t_{i\,1}$’	$t_{i\,2}$’
1	7	5
2	8	10
3	3	4
4	12	11
5	12	7
6	10	12
7	9	6
8	6	8

Step 2:

Job sets U={2, 3, 6, 8} and V={1, 4, 5, 7}

Step 3: Sort jobs in U as follows:

Job i : 3 8 2 6

$t_{i\,1}$’ : 3 6 8 10

Step 4: Sort jobs in V as follows:

Job i : 4 5 7 1

$t_{i\,2}$’ : 11 7 6 5

Step 5 output optimal sequence is {3, 8, 2, 6, 4, 5, 7, 1}

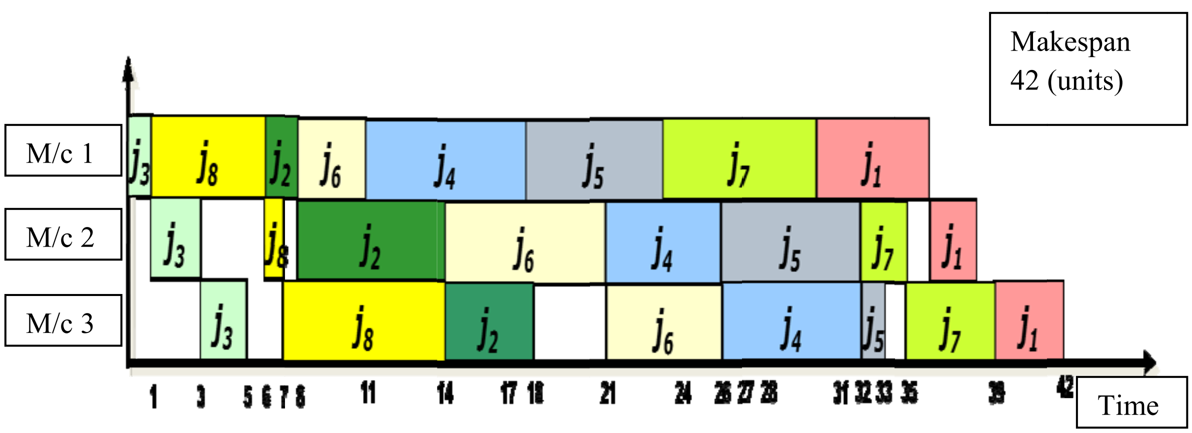

Figure 4.5 Gantt chart for 8-job, 3-machine problem by CDS Heuristic Rule:

4.10.4 RA Heuristic Rule:

Algorithm: RA Heuristic:

Procedure: RA Heuristic

Input: job list I, machine m;

Output: schedule S;

begin

for $i = 1$ to n

for $j = 1$ to m-1

$w_{j1} = m-(j-1), \quad w_{j2} = j$;

$$t_{i1}' = \sum_{j=1}^{m} w_{j1} t_{ij} \quad \textbf{and} \quad t_{i2}' = \sum_{j=1}^{m} w_{j2} t_{ij}$$

Where weights are defined as follows:

$$W_1 = \{ w_{j1} \mid j = 1, 2, ..., m \} = \{ m, m-1, ... 2, 1 \}$$
$$W_2 = \{ w_{j2} \mid j = 1, 2, ..., m \} = \{ 1, 2, ..., m-1, m \}$$

Calculate U = $\{i \mid t_{i1}' < t_{i2}'\}$ and V = $\{i \mid t_{i1}' \geq t_{i2}'\}$; // step 1

sort jobs in U with non-decreasing order of t_{i1}'; // step 2

sort jobs in V with non-increasing order of t_{i2}'; // step 3

output : optimal sequence is obtained as schedule S by U and V // step 4

end

Consider the above 8-job and 3-machine problem:

The solution constructed as follows:

Table 4.6 : Solution by RA Algorithm:

job i	$t_{i\,1}$’	$t_{i\,2}$’
1	22	18
2	22	26
3	9	11
4	37	35
5	31	21
6	28	32
7	29	23
8	24	28

Step 2: Job sets U={2, 3, 6, 8} and V={1, 4, 5, 7}

Step 3: Sort jobs in U as follows:

Job i : 3 2 8 6

$t_{i\,1}$’ : 9 22 24 28

Step 4: Sort jobs in V as follows:

Job i : 4 7 5 1

$t_{i\,2}$’ : 35 23 21 18

Step 5 :output optimal sequence is {3, 2, 8, 6, 4, 7, 5, 1}

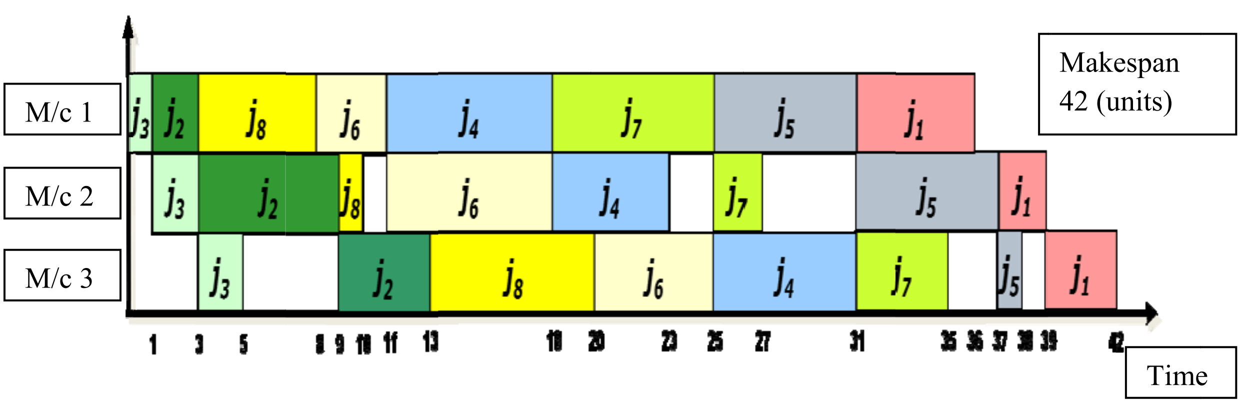

Figure 4.6 Gantt chart for 8-job, 3-machine problem by RA Heuristic Rule:

4.11 Heuristics for General 10-Machine and 10-Jobs Problems

Let us consider the case of 10-jobs and 10-machines to compare the various heuristics procedure: The problem in terms of processing times is as follows:

Table shows the processing time of 10 -jobs on 10 –machines where processing time =operation time (TO) and refixturing time (RT)

Table 4.7: (10-Machine and 10- jobs Flow-shop Problem) :

Jobs ⇨ M/C ⇩	1	2	3	4	5	6	7	8	9	10
1	5	2	1	7	6	3	7	5	7	4
2	2	6	2	5	6	7	2	1	8	3
3	3	4	2	6	1	5	4	7	6	5
4	5	2	1	3	8	2	6	1	9	8
5	7	6	3	2	6	2	5	7	1	3
6	9	2	7	3	4	1	5	3	8	1
7	7	5	2	2	3	5	1	6	2	3
8	8	2	5	4	9	3	2	6	1	8
9	2	6	4	2	6	2	5	2	6	3
10	7	1	4	2	4	6	2	2	6	7

4.11.1 For Guptas Heuristics:

The solution is constructed as follows:

Step 1: set the slope index s_i for job i as:

Calculate $s_i = e_i / \min\{t_{ik} + t_{i,k+1}\}$;

if $t_{i1} < t_{im}$ then

$e_i=1$;

else

$e_i = -1$;

$S_1=1/\min(48,50)$	$=0.0208$
$S_2=-1/\min(35,34)$	$=-0.0294$
$S_3=1/\min(27,30)$	$=0.0370$
$S_4=-1/\min(34,29)$	$=-0.0345$
$S_5=-1/\min(49,47)$	$=-0.0213$

$S_6=1/\min(30,33)$	$=0.0333$
$S_7=-1/\min(37,32)$	$=-0.0313$
$S_8=-1/\min(38,35)$	$=-0.0286$
$S_9=-1/\min(48,47)$	$=-0.0213$
$S_{10}=1/\min(38,41)$	$=0.0263$

Optimal sequence is constructed on the basic of decreasing order of slope values:

Optimal sequence:

$0.0370 \geq 0.0333 \geq 0.0263 \geq 0.0208 \geq -0.0213 \geq -0.0213 \geq -0.0286 \geq -0.0294 \geq -0.0313 \geq -0.0345$

$S_3 \geq S_6 \geq S_{10} \geq S_1 \geq S_5 \geq S_9 \geq S_8 \geq S_2 \geq S_7 \geq S_4$

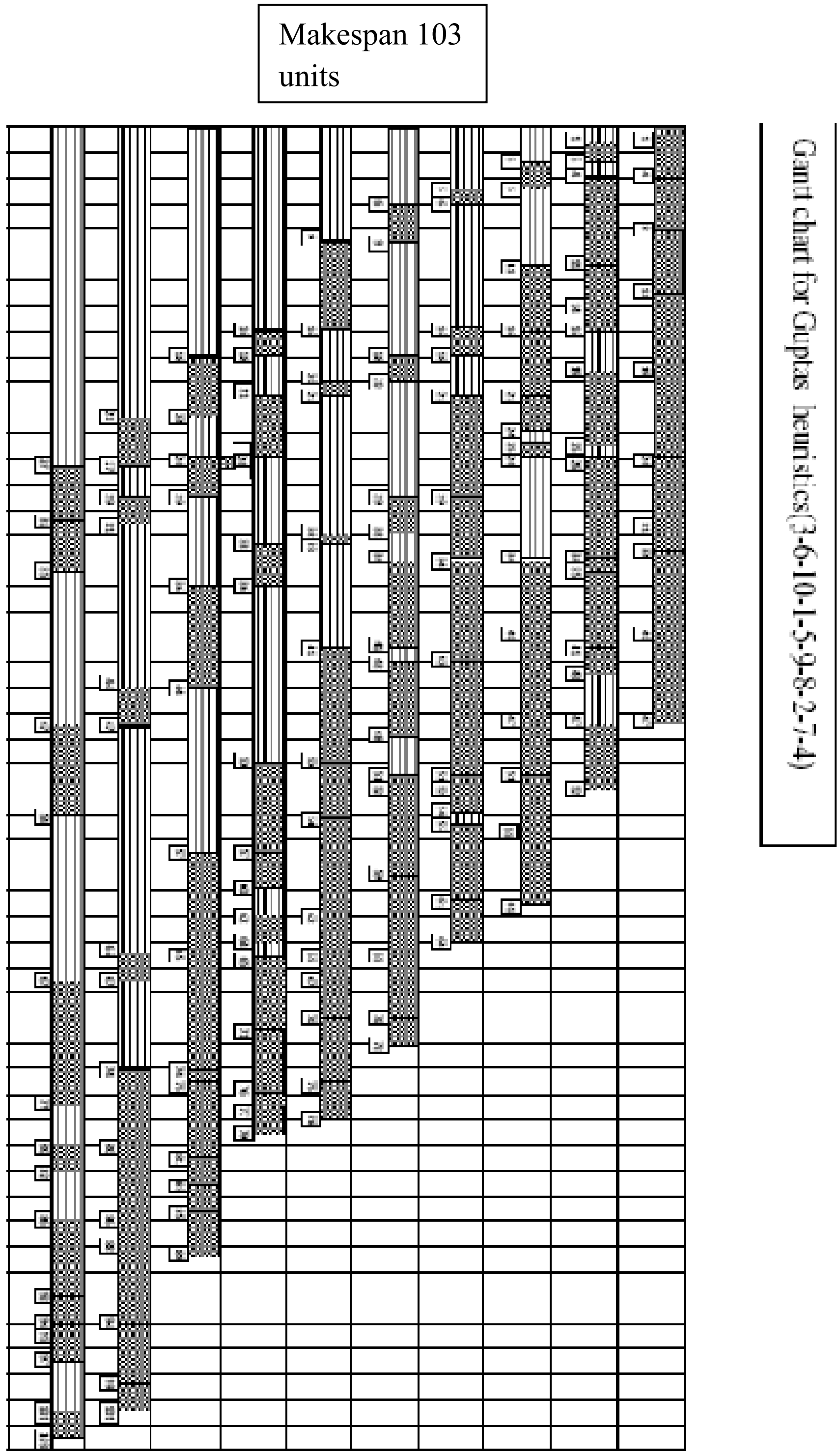

Figure 4.7 Gantt chart for 10-jobs and 10-machines problem by Gupta's Heuristic Rule:

4.11.2 For CDS Heuristics:

The solution for the above problem is constructed as follows:

Table 4.8: Solution by CDS Algorithm for 10-Machine and 10- jobs Problems:

job	t_{i1}’	t_{i2}’
1	<u>48</u>	50
2	35	<u>34</u>
3	<u>27</u>	30
4	34	<u>29</u>
5	49	<u>47</u>
6	<u>30</u>	33
7	37	<u>32</u>
8	38	<u>35</u>
9	48	<u>47</u>
10	<u>38</u>	41

Optimal sequence is constructed on the basic of increasing order of t_{i1}’ and t_{i2}’

Optimal sequence: 3-6-10-1-9-5-8-2-7-4

Makespan 102 units

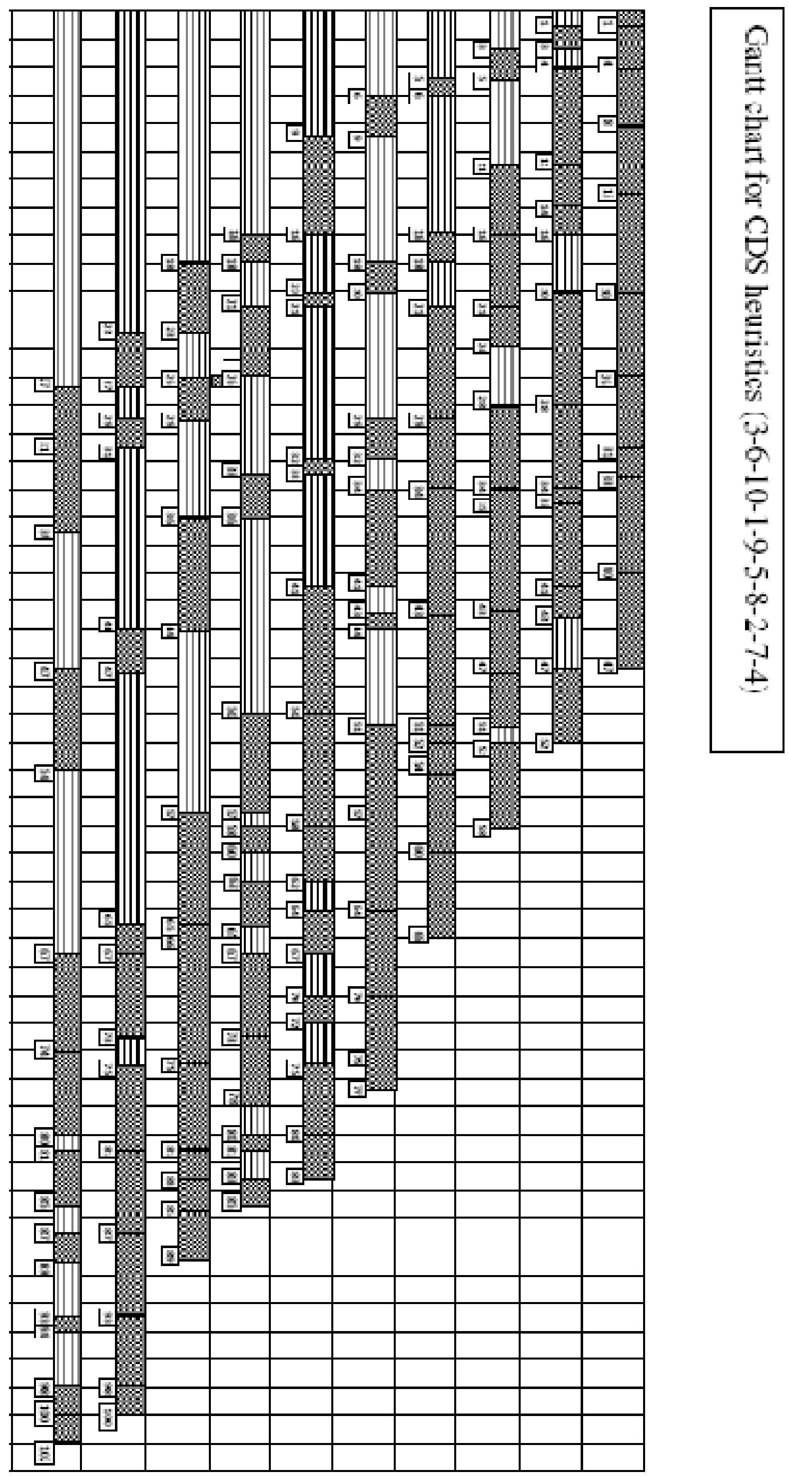

Figure 4.8 Gantt chart for 10-jobs and 10-machines problem by CDS Heuristic Rule:

4.11.3 For RA Heuristics:

The solution for the above problem is constructed as follows:

Table 4.9: Solution by RA Algorithm for 10-Machine and 10- jobs Problems

job	$t_{i\,1}$’	$t_{i\,2}$’
1	<u>277</u>	328
2	205	<u>191</u>
3	<u>139</u>	202
4	237	<u>159</u>
5	<u>289</u>	294
6	203	<u>193</u>
7	239	<u>190</u>
8	227	<u>213</u>
9	328	<u>266</u>
10	<u>235</u>	260

Optimal sequence is constructed on the basic of increasing order of $t_{i\,1}$’ and $t_{i\,2}$’

Optimal sequence: 3-10-1-5-9-8-6-2-7-4

Makespan 97 units

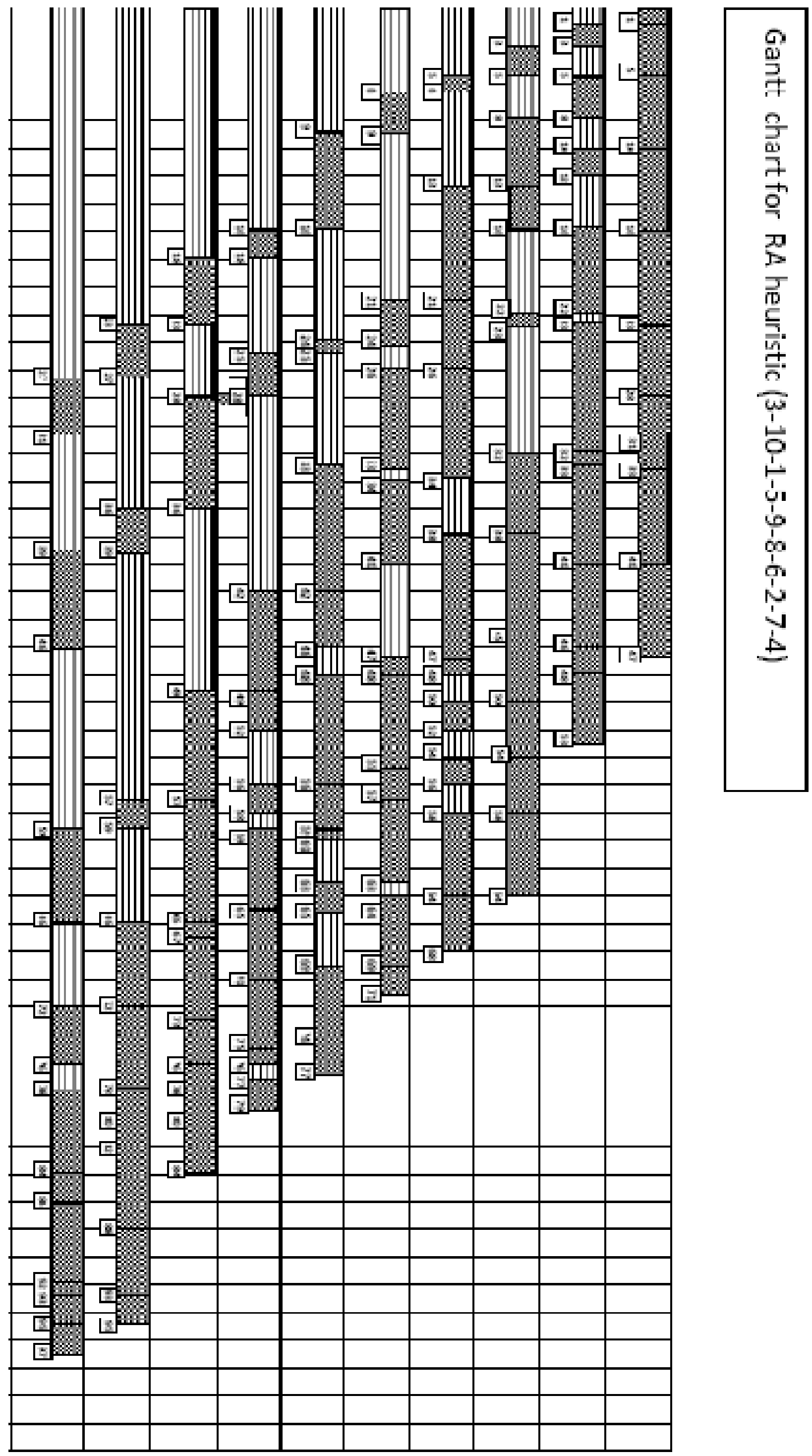

Figure 4.9 Gantt chart for 10-jobs and 10-machines problem by RA Heuristic Rule:

4.11.4 For Palmers Heuristics:

The solution for the above problem is constructed as follows:

Slope=(m-1) tj,m +(m-3)tj,(m-1)+(m-5)tj,(m-2)……………………………….

For 10-jobs and 10- machines:

S1=(m-1)t1,10+(m-3)t1,9…………………………………………+(m-19)t1,1

For 10 machines (m=10)

=(10-1)*7+(10-3)*2+(10-5)*8+(10-7)*7+(10-9)*9+(10-11)*7+(10-13)*5+

(10-15)*3+(10-17)*2+(10-19)*5

=51

Similarly

S2=-14

S3=63

S4=-78

S5=5

S6=-10

S7=-49

S8=-14

S9=-62 Optimal sequence is constructed on the basic of decreasing order

S10=25 of slope values

Optimal sequence:

$S_3 \geq S_1 \geq S_{10} \geq S_5 \geq S_6 \geq S_2 \geq S_8 \geq S_7 \geq S_9 \geq S_4$

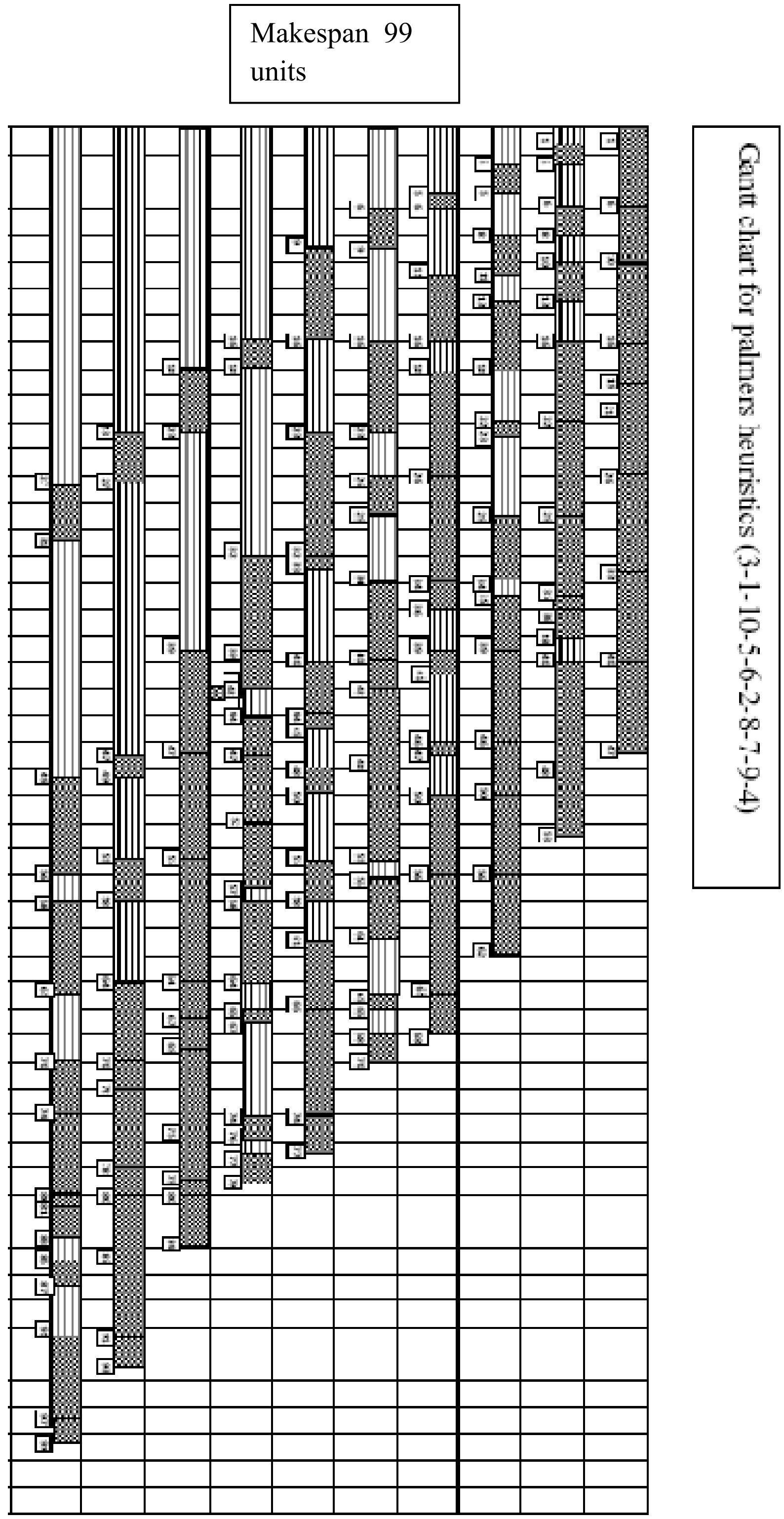

Figure 4.10 Gantt chart for 10-jobs and 10-machines problem by Palmers Heuristic Rule:

4.12 Heuristics for General 8-Machine and 10-Jobs Problems

Again let us consider the case of 10-jobs and 8-machines to compare the various heuristics procedure: The problem in terms of processing times is as

Table shows the processing time of 10 -jobs on 8 –machines where processing time =operation time (TO) and refixturing time (RT)

Table 4.10: (8-Machine and 10- jobs Flow-shop Problem) :

Jobs ⇨ M/C ⇩	1	2	3	4	5	6	7	8	9	10
1	6	3	8	4	9	3	5	2	1	6
2	5	9	1	6	8	2	9	8	4	3
3	1	5	6	3	2	4	4	9	6	5
4	7	7	4	1	9	3	2	1	2	5
5	9	2	3	5	2	7	4	6	5	2
6	3	5	9	6	5	2	8	3	4	7
7	4	6	5	7	9	3	6	4	3	1
8	2	1	9	7	6	5	6	8	9	9

4.12.1 For Guptas Heuristics:

The solution is constructed as follows:

Step 1: set the slope index s_i for job i as:

Calculate $s_i = e_i / \min\{t_{ik} + t_{i,k+1}\}$;

if $t_{i1} < t_{im}$ then

e_i=1;

else

e_i = -1;

S_1=-1/min(35,31) =-0.0323	S_6=1/min(24,26) =0.0417
S_2=-1/min(37,35) =-0.0286	S_7=1/min(38,39) =0.0263
S_3=1/min(36,37) =0.0278	S_8=1/min(33,39) =0.0303
S_4=1/min(32,35) =0.0313	S_9=1/min(25,33) =0.0400
S_5=-1/min(44,41) =-0.0244	S_{10}=1/min(29,32) =0.0345

Optimal sequence:

$0.0417 \geq 0.0400 \geq 0.0345 \geq 0.0313 \geq 0.0303 \geq 0.0278 \geq 0.0263 \geq -0.0244 \geq -0.0286 \geq -0.0323$

$S_6 \geq S_9 \geq S_{10} \geq S_4 \geq S_8 \geq S_3 \geq S_7 \geq S_5 \geq S_2 \geq S_1$

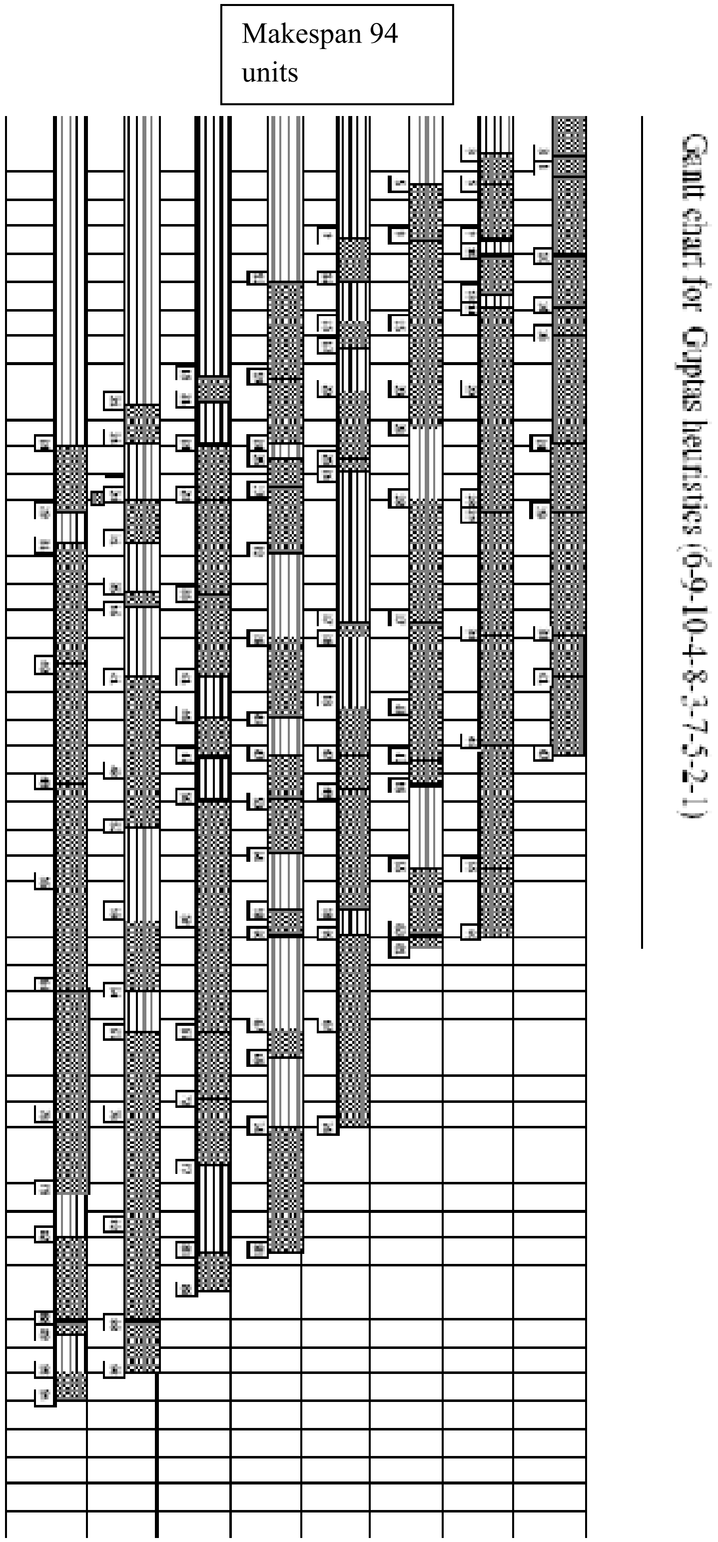

Figure 4.11 Gantt chart for 10-jobs and 8-machines problem by Gupta's Heuristic Rule:

4.12.2 For CDS Heuristics:

The solution for the above problem is constructed as follows:

Table 4.11: Solution by CDS Algorithm for 8-Machine and 10- jobs Problems

job	$t_{i\,1}'$	$t_{i\,2}'$
1	35	31
2	37	35
3	36	37
4	32	35
5	44	41
6	24	26
7	38	39
8	33	39
9	25	33
10	29	32

Optimal sequence is constructed on the basic of increasing order of $t_{i\,1}'$ and $t_{i\,2}'$

Optimal sequence: 6-9-10-4-8-3-7-5-2-1

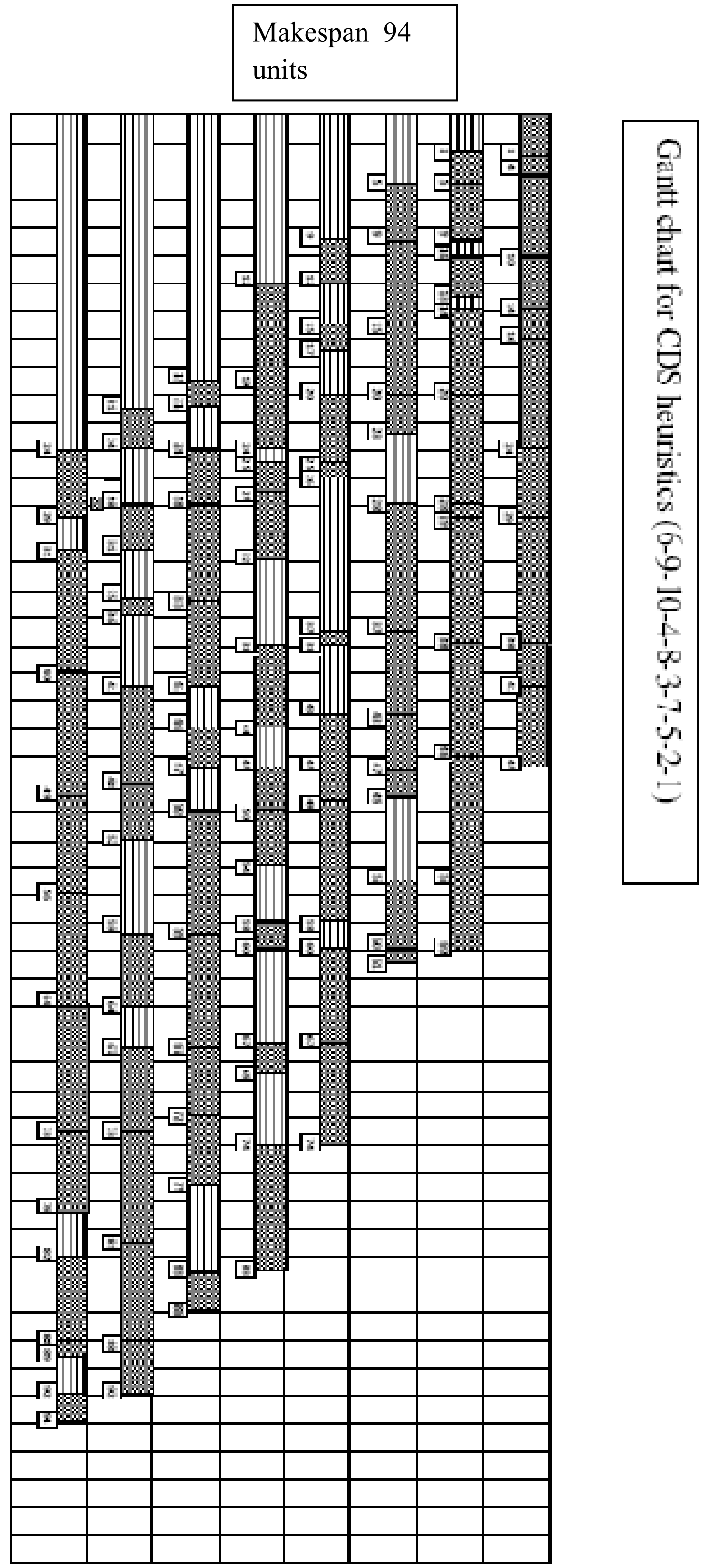

Figure 4.12 Gantt chart for 10-jobs and 8-machines problem by CDS Heuristic Rule:

4.12.3 For RA Heuristics:

The solution for the above problem is constructed as follows:

Table 4.12: Solution by RA Algorithm for 8-Machine and 10- jobs Problems

job	$t_{i\,1}$’	$t_{i\,2}$’
1	179	154
2	188	154
3	185	220
4	156	195
5	232	218
6	122	139
7	195	201
8	180	189
9	129	177
10	164	178

Optimal sequence is constructed on the basic of increasing order of $t_{i\,1}$’ and $t_{i\,2}$’

Optimal sequence: 6-9-4-10-8-3-7-5-2-1

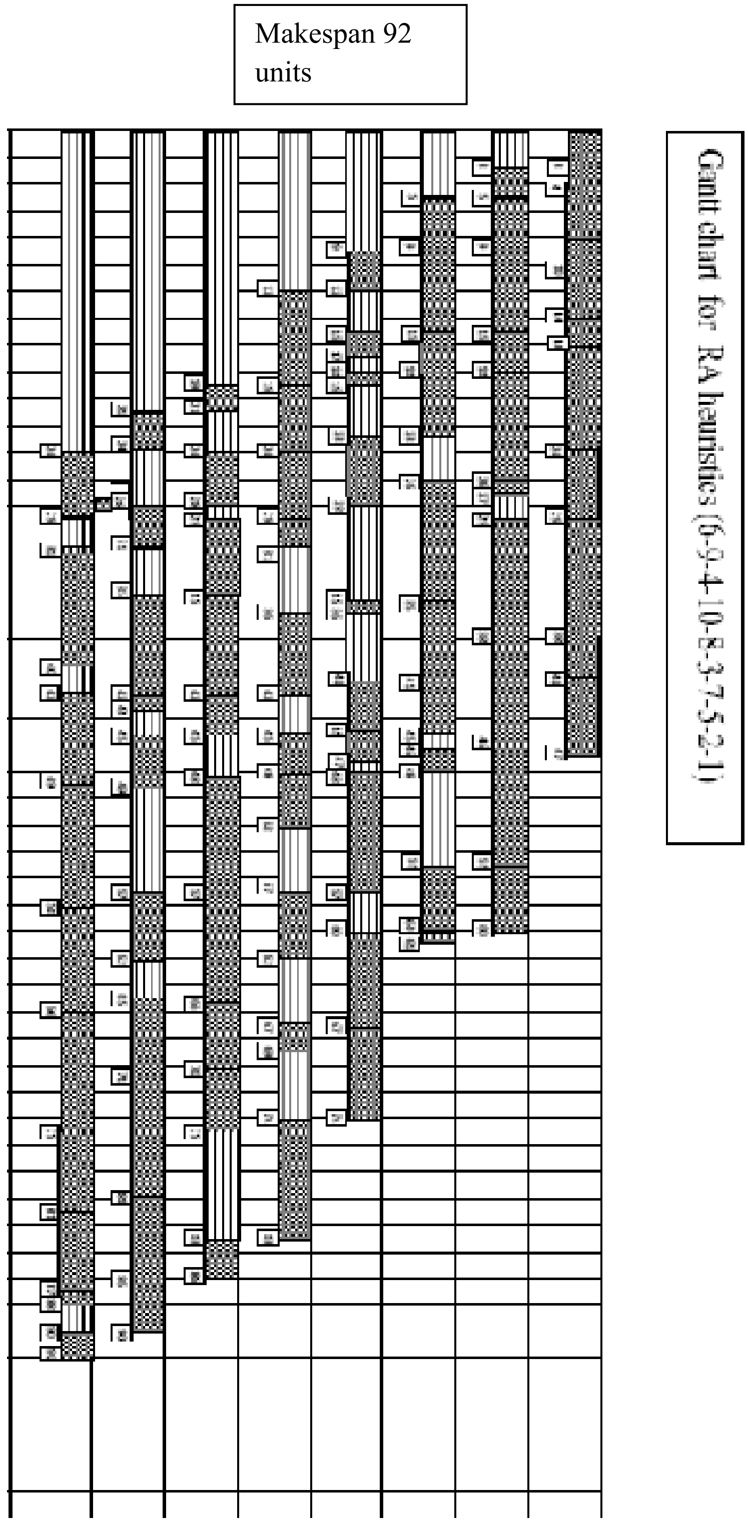

Figure 4.13 Gantt chart for 10-jobs and 8-machines problem by RA Heuristic Rule:

4.12.4 For Palmers Heuristics:

The solution for the above problem is constructed as follows:

Slope=(m-1) tj,m +(m-3)tj,(m-1)+(m-5)tj,(m-2)……………………………….

For 10-jobs and 8- machines:

S1=(m-1)t1,8+(m-3)t1,7…………………………………………+(m-15)t1,1

For 8 machines (m=8)

=(8-1)*2+(8-3)*4+(8-5)*3+(8-7)*9+(8-9)*7+(8-11)*1+(8-13)*5+

(8-15)*6

=-25

Similarly

S2=-34

S3=35

S4=39

S5=-14

S6=17

S7=6

S8=9 Optimal sequence is constructed on the basic of decreasing

S9=48 order of slope values

S10=14

Optimal sequence:

$S_9 \geq S_4 \geq S_3 \geq S_6 \geq S_{10} \geq S_8 \geq S_7 \geq S_5 \geq S_1 \geq S_2$

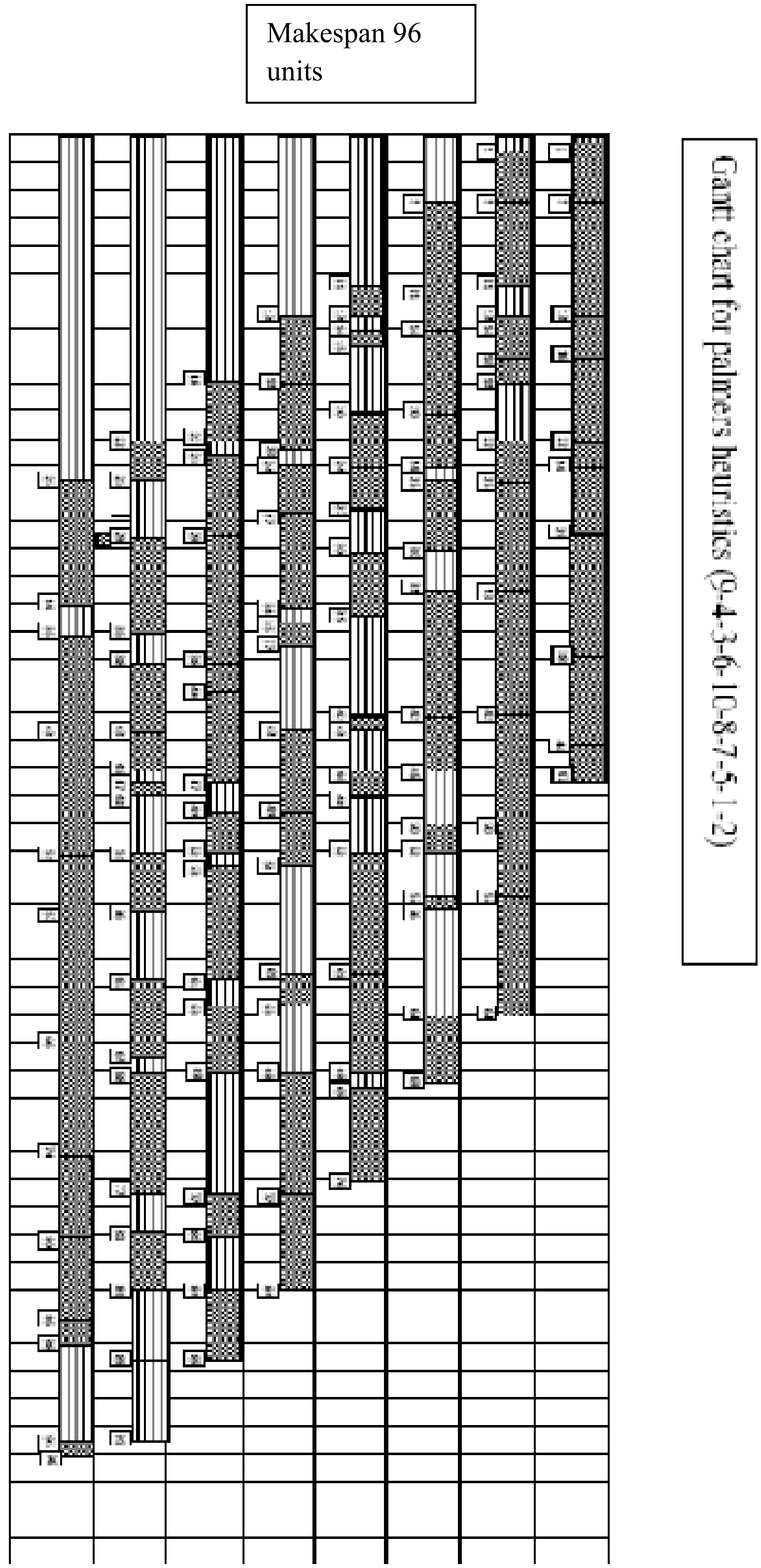

Figure 4.14 Gantt chart for 10-jobs and 8-machines problem by Palmers Heuristic Rule:

4.13 Scope of the Present Work

Another important consideration is the choice of appropriate criteria for scheduling. Although the ultimate objective of any enterprise is to maximize the net present value of the shareholder wealth, this criterion does not easily lend itself to operational decision-making in scheduling. Some researchers are developing methodologies which take revenue and cost effects of schedules into consideration. Researchers and practitioners have so far used operational surrogates that influence costs and revenues. These include: number of parts tardy, average tardiness, weighted tardiness, throughput (this is a revenue-influencing surrogate), as well as average number of parts in the system, machine utilization, and work-in- process inventory, for example. Analyses of these surrogates indicate that a scheduling procedure which does well for one criterion is not necessarily the best for some other. For example, attempts to reduce mean tardiness can lead to an increase in mean flow time. Minimizing make span can result in higher mean flow time.

Further, a criterion which is appropriate at one level of decision-making may be unsuitable at another level. These raise further complications in the context of FMSs because of the additional decision variables involved in including, for example, routing and sequencing alternatives. Scheduling may likely be a more complicated function when each part needs to visit several machines and when several operations have a choice of machines. The availability of alternative routing can improve system performance as well as increase scheduling complexities. In general, queuing network models have been used to address FMS design problems quantitatively and FMS planning problems qualitatively.

Queuing network models generally do not have sufficient modeling capability to address detailed scheduling problems. Further research is required into using queuing models to address some scheduling problems.

4.14 Summary:

Conventional methods of solving scheduling problems based on priority rules (FIFO, SPT, EDD …) determined the corresponding schedule but usually, still having idle times. To reduce these and improving CIM productivity optimization is necessary. Single factory production in traditional manufacturing has been gradually replaced by multi-factory production due to the trend of globalization. These factories may be geographically distributed in different locations, which allow them to be closer to their customers, to comply with the local laws, to focus on a few product types, to produce and market their products more effectively, and to be responsive to market changes more quickly.

Each factory is usually capable of manufacturing a variety of product types. Some may be unique in a particular factory, while some may not. In addition, they may have different production efficiency and various constraints depending on the machines, labor skills and education levels, labor cost, government policy, tax, nearby suppliers, transportation facilities, etc. This induces different operating costs, production lead time, customer service levels, etc. in different factories. The objective of this approach is to maximize the system efficiency by finding an optimal planning for a better collaboration among various processes

Chapter-V

RESULTS AND DISCUSSION

CHAPTER-V

RESULTS AND DISCUSSION

5.1 Results of the Above Applied Heuristics Rules to Flow Shop Scheduling

5.1.1 For 3x8 problems (3-machines and8-jobs problem):

Table 5.1: 3x8 Flow-shop Problem :

Job i	1	2	3	4	5	6	7	8
$t_{i\,1}$	5	2	1	7	6	3	7	5
$t_{i\,2}$	2	6	2	5	6	7	2	1
$t_{i\,3}$	3	4	2	6	1	5	4	7

Makespan for the applied heuristics rules are:

Rule	Guptas	CDS	RA	Palmers
Makespan	41 units	42 units	42 units	43 units

Make span is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand

5.1.2 For 10x10 problems (10-machines and10-jobs problem):

Table 5.2: 10x10 Flow-shop Problem :

Jobs ⇒ M/C ⇓	1	2	3	4	5	6	7	8	9	10
1	5	2	1	7	6	3	7	5	7	4
2	2	6	2	5	6	7	2	1	8	3
3	3	4	2	6	1	5	4	7	6	5
4	5	2	1	3	8	2	6	1	9	8
5	7	6	3	2	6	2	5	7	1	3
6	9	2	7	3	4	1	5	3	8	1
7	7	5	2	2	3	5	1	6	2	3
8	8	2	5	4	9	3	2	6	1	8
9	2	6	4	2	6	2	5	2	6	3
10	7	1	4	2	4	6	2	2	6	7

Rule	Guptas	CDS	RA	Palmers
Makespan	103 units	102 units	97 units	99 units

5.1.3 For 8x10 problem(8-machines and10-jobs problem):

Table 5.3 : 8x10 flow shop problem:

Jobs ⇨ M/C ⇩	1	2	3	4	5	6	7	8	9	10
1	6	3	8	4	9	3	5	2	1	6
2	5	9	1	6	8	2	9	8	4	3
3	1	5	6	3	2	4	4	9	6	5
4	7	7	4	1	9	3	2	1	2	5
5	9	2	3	5	2	7	4	6	5	2
6	3	5	9	6	5	2	8	3	4	7
7	4	6	5	7	9	3	6	4	3	1
8	2	1	9	7	6	5	6	8	9	9

Rule	Guptas	CDS	RA	Palmers
Makespan	94 units	94 units	92 units	96 units

Make span is the time length from the starting of the first operation of the first demand to the finishing of the last operation of the last demand.

Chapter-VI

CONCLUSION AND FUTURE SCOPE

CHAPTER-VI

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

By Scheduling, we assign a particular time for completing a particular job. The main objective of scheduling is to arrive at a position where we will get minimum processing time. The problem examined here is the n-job, m-machine problem in a flow shop .This work arrange the jobs in a particular order and get many combinations and choose that combination where we get the minimum make span. This study try to solve the problem of a flow shop scheduling with the objective of minimizing the makes pan. Here the objective is to minimize the make span of batch-processing machines in a flow shop. Comparison based on Gupta's heuristics, RA heuristic's, Palmer's heuristics, CDS heuristics are proposed here. Analytic solutions in all the heuristics are investigated. Gantt chart were generated to verify the effectiveness of the proposed approaches. As a result of the work proposed here the researcher found that out of the four proposed algorithm RA heuristics yields efficient result because here the processing times are determined from a weighting scheme. The main advantage of RA heuristics for yielding the best result as compared to others is that, it is the combination of two heuristics approaches (Palmers slope index + CDS method) .Here in this work RA heuristics yields efficient result, This were explained with the help of numerical examples and their performances are examined with the help of Gantt charts. The algorithm is written in a very few lines of code, and requires only specification of the problem and a few parameters in order to solve it.

The Gantt schedule can illustrate the relationship between work activities having duration, events without duration that indicate a significant completion, that represent major achievements or decision points. The researchers proposed the comparison between four heuristics approaches and tell which is more efficient. For a given task, the order in which tasks are considered and the criteria by which machine centers are selected play major roles for optimizing the scheduling policies. The model introduced here proposed a new approach for planning scheduling problems – providing a way to optimize the make span. Scheduling is an activity to select the right future operational program or diagram of an actual time plan for allocating competitive different demands of different products, delivery dates, by sequencing through different machines, operations, and routings for the combination of the high flexibility of job shop type with high productivity of flow-shop type and meeting delivery dates. The researcher has provided a technique for processing information that is at once elegant and versatile.

6.2 Scope for Future Work

Further research may be conducted to investigate the applications of other metaheuristics to the lot-streaming flow shop problem. It is also worthwhile to design other versions of RA heuristics to continue pursuing the best performance of RA heuristics. Future research should address problems with different shop environments, including parallel machines flow shop, job shop, and open shop. Problems with other performance measures, such as minimum due dates, maximum lateness, and multi-criteria measures should also be studied. Future research should be directed to generalize the method to multipart, multi machine group cases.

REFERENCES

REFERENCES

1. Felix T.S. Chan, S.H. Chung, L.Y. Chan, G. Finke, M.K. Tiwari , "Solving distributed FMS scheduling problems subject to maintenance: Genetic algorithms approach", Robotics and Computer-Integrated Manufacturing 22 (2006) 493–504

2. Safia Kedad-Sidhoum, Yasmin Rios Solis, Francis Sourd, "Lower bounds for the earliness–tardiness scheduling problem on parallel machines with distinct due dates" , European Journal of Operational Research 189 (2008) 1305–1316

3. Moosa Sharafali , Henry C. Co , Mark Goh , "Production scheduling in a flexible manufacturing system under random demand" , European Journal of Operational Research 158 (2004) 89–102

4. K.H. Ecker , J.N.D. Gupta, "Scheduling tasks on a flexible manufacturing machine to minimize tool change delays" , European Journal of Operational Research 164 (2005) 627–638

5. Xiaoping Li, QianWang, ChengWu , "Efficient composite heuristics for total flow time minimization in permutation flow shops" , Omega 37 (2009) 155 – 164

6. I.Drstvensek, I. Pahole, J. Balic, "A model of data flow in lower CIM levels" , Journal of Materials Processing Technology 157–158 (2004) 123–130

7. Ezedeen Kodeekha," a new method of FMS scheduling using optimization and simulation**" ,** Department of Production, Informatics, Management and Control Faculty of Mechanical Engineering Budapest University of Technology and Economics

8. Clarence H Martin, "A hybrid genetic algorithm/mathematical programming approach to the multi-family flow shop scheduling problem with lot streaming" , Omega 37 (2009) 126 – 137

9. Chin-Chia Wu, Wen-Chiung Lee, "A note on the total completion time problem in a permutation flow shop with a learning effect" , European Journal of Operational Research 192 (2009) 343–347

10. Christos Koulamas , George J. Kyparisis, “Single-machine scheduling with waiting-time-dependent due dates” , European Journal of Operational Research 191 (2008) 577–581

11. Kanchan Das , M.F. Baki , Xiangyong Li , “Optimization of operation and changeover time for production planning and scheduling in a flexible manufacturing system” , Computers & Industrial Engineering xxx (2008) xxx–xxx

12. Bo Chen , Chung-Yee Lee , “Logistics scheduling with batching and transportation” , European Journal of Operational Research 189 (2008) 871–876

13. Konstantinos N. Androutsopoulos, Konstantinos G. Zografos, “Solving the multi-criteria time-dependent routing and scheduling problem in a multimodal fixed scheduled network” , European Journal of Operational Research 192 (2009) 18–28

14. Nadia Hamani**,** Nathalie Dangoumau, Etienne Craye , “Reactive mode handling of flexible manufacturing systems” , Mathematics and Computers in Simulation xxx (2008) xxx–xxx

15. Tiente Hsu , Ouajdi Korbaa , Re´my Dupas , Gilles Goncalves , “Cyclic scheduling for F.M.S.: Modelling and evolutionary solving approach” , European Journal of Operational Research 191 (2008) 464–484

16. Ruslan Sadykov, “A branch-and-check algorithm for minimizing the weighted number of late jobs on a single machine with release dates”, European Journal of Operational Research 189 (2008) 1284–1304

17. Xianyi Wu , Xian Zhou, “Stochastic scheduling to minimize expected maximum lateness” , European Journal of Operational Research 190 (2008) 103–115

18. Andrea Rossi, Gino Dini, “Flexible job-shop scheduling with routing flexibility and separable setup times using ant colony optimisation method” Robotics and Computer-Integrated Manufacturing 23 (2007) 503–516

19. Shi-jin Wang, Li-feng Xi, Bing-hai Zhou , “FBS-enhanced agent-based dynamic scheduling in FMS” , Engineering Applications of Artificial Intelligence 21 (2008) 644–657

20. J.F. Gonc¸alves , J.J.M. Mendes , M.G.C. Resende , “A genetic algorithm for the resource constrained multi-project scheduling problem” , European Journal of Operational Research 189 (2008) 1171–1190

21. T.C.E. Cheng , C.T. Ng , J.J. Yuan, “Single-machine scheduling of multi-operation jobs without missing operations to minimize the total completion time” , European Journal of Operational Research 191 (2008) 320–331

22. Ruud Teunter , Konstantinos Kaparis , Ou Tang, “Multi-product economic lot scheduling problem with separate production lines for manufacturing and remanufacturing” , European Journal of Operational Research 191 (2008) 1241–1253

23. Lixin Tang , Hua Gong, “A hybrid two-stage transportation and batch scheduling problem” , Applied Mathematical Modelling 32 (2008) 2467–2479

24. Chao-Tang Tseng , Ching-Jong Liao, “A discrete particle swarm optimization for lot-streaming flow shop scheduling problem” , European Journal of Operational Research 191 (2008) 360–373

25. Pei Chann Chang, Shih Hsin Chen , V. Mani , “A hybrid genetic algorithm with dominance properties for single machine scheduling with dependent penalties” , Applied Mathematical Modelling 33 (2009) 579–596

26. T.C.E. Cheng , B.M.T. Lin , “Johnson’s rule, composite jobs and the relocation problem” , European Journal of Operational Research 192 (2009) 1008–1013

27. Seong-Jong Joo, “Scheduling preventive maintenance for modular designed components: A dynamic approach” , European Journal of Operational Research 192 (2009) 512–520

28. Yaohua He, Chi-Wai Hui, “A rule-based genetic algorithm for the scheduling of single-stage multi-product batch plants with parallel units” , Computers and Chemical Engineering 32 (2008) 3067–3083

29. Jiaqiong Chen , Ronald G. Askin, “Project selection, scheduling and resource allocation with time dependent returns” , European Journal of Operational Research 193 (2009) 23–34

30. Adam Janiak , Tomasz Krysiak , Costas P. Pappis , Theodore G. Voutsinas , “A scheduling problem with job values given as a power function of their completion times” , European Journal of Operational Research 193 (2009) 836–848

31. Grzegorz Waligo´ra, “Tabu search for discrete–continuous scheduling problems with heuristic continuous resource allocation” , European Journal of Operational Research 193 (2009) 849–856

32. Vicente Valls , A´ ngeles Pe´rez , Sacramento Quintanilla, “Skilled workforce scheduling in Service Centres” , European Journal of Operational Research 193 (2009) 791–804

33. Vikram Tiwari , James H. Patterson , Vincent A. Mabert, “Scheduling projects with heterogeneous resources to meet time and quality objectives” , European Journal of Operational Research 193 (2009) 780–790

34. Stanley P.Y. Fung, “Lower bounds on online deadline scheduling with preemption penalties” , Information Processing Letters 108 (2008) 214–218

35. Lixin Tang, Yufang Zhao, “Scheduling a single semi-continuous batching machine” , Omega 36 (2008) 992 – 1004

36. Tamer Eren , Ertan Gu¨ner, “A bicriteria flowshop scheduling with a learning effect” , Applied Mathematical Modelling 32 (2008) 1719–1733

37. Xianyi Wu , Xian Zhou, “Stochastic scheduling to minimize expected maximum lateness” , European Journal of Operational Research 190 (2008) 103–115

38. Bibo Yang , Joseph Geunes, “Predictive–reactive scheduling on a single resource with uncertain future jobs” , European Journal of Operational Research 189 (2008) 1267–1283

39. Gur Mosheiov , Assaf Sarig, “Due-date assignment on uniform machines” , European Journal of Operational Research 193 (2009) 49–58

40. Dirk Biskup , Jan Herrmann , “Single-machine scheduling against due dates with past-sequence-dependent setup times” , European Journal of Operational Research 191 (2008) 587–592

41. Rongjun Chen, Wanzhen Huang, Guochun Tang, “Dense open-shop schedules with release times” , Theoretical Computer Science 407 (2008) 389_399