OBTAIN MAKESPAN OF JOB SHOP SCHEDULING OPERATION

A Project Report Submitted in Partial Fulfilment of the Requirements for the Degree of
B. Tech (Mechanical Engineering**)**

By

BINAY KUMAR SAHOO

ROLL NO 109ME0412

Department of Mechanical Engineering

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

JULY, 2013

OBTAIN MAKESPAN OF JOB SHOP SCHEDULING OPERATION

A Project Report Submitted in Partial Fulfilment of the Requirements for the Degree of
B. Tech (Mechanical Engineering**)**

By

BINAY KUMAR SAHOO

ROLL NO 109ME0412

Under the Supervision of:

Dr. SIBA SANKAR MOHAPATRA

Professor

(Department of Mechanical Engineering, NIT Rourkela)

Department of Mechanical Engineering

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

JULY, 2013

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

CERTIFICATE

This is to certify that the work in this thesis entitled ***Jobshop Schedule*** by **Binay Kumar Sahoo,** has been carried out under my supervision in partial Fulfilment of the requirements for the degree of **Bachelor of Technology** in *Mechanical Engineering* during session 2012 - 2013 in the Department of Mechanical Engineering, National Institute of Technology, Rourkela.

To the best of my knowledge, this work has not been submitted to any other University/Institute for the award of any degree or diploma.

Dr. Siba Sankar Mohapatra
(Superviser**)**
Professor
Dept. Of Mechanical Engg .
National Institute of Technology
Rourkela , 769008 .

ACKNOWLEDGEMENT

It's a great pleasure to express my deep sense of gratitude and respect to my supervisor Prof. Siba Sankar Mohapatra for his excellent guidance, suggestions and support throughout the project. I feel extremely lucky to be able to work under the guidance of such a dynamic personality. I am also thankful to Prof. K.P. Maity, H.O.D, Department of Mechanical Engineering, N.I.T. Rourkela for his constant support and encouragement.

Last but not the least, I extend my sincere thanks to all other faculty members, my senior research fellow and my friends at the Department of Mechanical Engineering, NIT, Rourkela, for their help and valuable advice in every stage for successful completion of this project report.

Date: Binay Kumar Sahoo

N.I.T. Rourkela 109me0412

Dept. Of Mechanical Engg.

CONTENTS

ABSTRACT

When a large number of jobs and machines are taken into consideration, efficiency in job shop scheduling plays a vital role. As the number on jobs & machines increase, the job shop scheduling problems approaches NP hard difficulty. Many heuristic methods are devised that produce solutions with near optimal solution. This work deals with the job shop scheduling using an algorithm aimed at creating a mathematical model without machine availability constraint. Operation based representation is used to decode the schedule in the algorithm. A C++ code was used to generate an algorithm for finding the optimal solution. The input parameters are operation time and operation sequence for each job in the machines provided. This work used the makespan values of the schedules to compare the results.

LIST OF FIGURES AND TABLE

Chapter 1

The Introduction

Chapter 1

The Introduction

Flexible Manufacturing System (FMS):

A flexible manufacturing system (FMS) is a manufacturing system in which there is some amount of flexibility that allows the system to react in the case of changes, whether predicted or unpredicted. This flexibility is generally considered to fall into two categories, which both contain numerous subcategories. In this type of production where small batches of a variety of custom products are made. Most of the products produced require a unique setup and sequencing of the processing steps.

Scheduling:

A job is characterized by its route, its processing requirement and priority. In a shop scheduling the key issue is to decide how and when to schedule. Job may not be scheduled based on the shortest processing time. Scheduling is categorized into 1. Open Shop Scheduling, 2. Flow Shop Scheduling, 3 Job Shop Scheduling

Open Shop Scheduling:

It is a scheduling problem where, given *n* jobs and *m* workstations, each job has to be processed on a workstation at least once. However, some of these processing times may be zero. The order in which this happens is not relevant.

Flow Shop Scheduling:

In flow shop scheduling there are a set of m number of jobs and n number of machines, where a strict sequence of operations for each job is followed. A minimal downtime and minimal waiting time are the constraints in the continuous flow of processes. Production facilities are generally found to be using flow shop scheduling problems. A scheduling problem for flow shop is a generalized version of the problem for job shop scheduling of flexible manufacturing systems. Here each machine has the ability to perform more than one operation for a particular job.

Job Shop Production:

In the classical job shop problem, a finite number of jobs are to be processed by a finite number of machines. Each job consists of a predetermined sequence of operations, which will be processed without interruption for a period of time on each machine. The operations corresponding to the same job are processed according to their technological sequence and none of them will be able to start processing before the preceding operation is over. There is no initial machine that performs only the first operation of a job, nor is there a terminal machine that performs only the last operation of a job. A viable program is an assignment of operations in time on a machine without violation of restrictions workshops. A makespan is defined as the maximum time for completion of all jobs.

Our goal is to generate such a schedule in the process of job shop scheduling to minimize the makespan i.e. the time length of the schedule, in which all the operations of each job is completed.

The job shop schedule provides a set of resources to tasks over time. In recent years, lot of research has been done in this field of operational research. It mainly focuses on finding ways to give jobs to the machines that meet certain criteria and an objective function is optimized. So far a 100 percent perfect method to get the optimal solution in any kind of workshop programming has not been obtained.

Constraints involved in a job shop scheduling problems:

- A job should not visit the same team more than once.
- Presence of no precedence constraints on operations of different jobs.
- Operations ones started can’t be interrupted.
- Each machine can process only one job at a time.
- Each job must go through a particular predefined sequence of operations

Each optimization problem should have an objective function that has to be either minimized or maximized in order to obtain a certain solution. In this case, the objective function is the value of makespan or the length interval of programming. The span value to be defined as follows: Here, each job has number of operations and each operation has a particular make value. When these operations get arranged in the sequence of the program and then each machine gets a particular make value. Maximum of the make values of all the machines is the value of Makespan (longest path).

Chapter 2

Literature Review

Chapter 2

Literature Review

Job shop scheduling problems are N-P hard. Brucker [1] and Garey [2] stated this and thus getting solution of these types of problems are very difficult. Many heuristic approaches have been developed in the last decades by researchers to optimize programming problems of job shop schedules. Some of them are: despatch rules, genetic algorithm, artificial immune system, fuzzy logic, simulated annealing, particle swarm optimization, ant colony optimization etc.

Brandimarte (1993) was the first to apply this heuristic method to solve job shop scheduling [3]. Brucker [5] Carlier and Pison [4] and devised branch and bound methods for the solution to small problems. To solve larger size problems Blazewicz [6] developed effective local search method. The results in his method were found for at least one preferred program and which reduced the searching efforts.

In-Chan Choi [7] aimed to develop local search algorithm to solve job shop scheduling problem. The objective function is to minimize makespan. Sequence dependent setup condition is added to the problem. The setup time of each job depends on sequence of jobs in each machine. This paper solves the problem by local search algorithm. Local search algorithm helps to reduce computation time.

Erscher et al. a branch and bound method with three parts. Step 1 is to calculate the lower limit, the step 2 is branching and step 3 is the removal of the node [8]. Hurik, Jarisch and Thole (1999) and Dazere-Peres use different methods of tabu search for scheduling jobshop problems. [9]

Mastrolilli and Gambardella (2000) worked on FJSS's neighbourhood functions that can be used in Meta heuristic optimization techniques. This method performed better than any other method in terms of computational results and quality of the solution. [10]

D. A. Koonce [11] used data mining to find the programming model for the job shop scheduling problems. Propose of this work is to apply the methodology of data mining to explore the pattern. The problem objective is to minimize makespan. Genetic algorithm is used to generate a good solution. Data mining is used to find the relationship between sequences and predict the next job in sequence. The result of data mining can be used to summarize new rule which gives the result like result of the genetic algorithm.

Chandrasekharan [12] introduced three new dispatching rules for dynamic flow problem shop and job shop problem. The performance of these rules, compared with 13 sequencing rules. The case study is the simulation study for the problem of flow shop scheduling. The problems are modified again by random route jobs. The problems are changed shop scheduling problem programming problem flow workshop. The study can conclude that the performance of dispatching rules is being influenced by the routing of job and shop floor settings.

Hiroshi [13] used shift bottleneck procedure to solve the job shop scheduling problem. The problem objective is to minimize the total cost of holding. The specific restriction adds to the problem. The additional constraint is any limitation to the later work. The experiment shows that the bottleneck procedure can reduce the time change calculation.

Anthony [14] presented Memmetic algorithm for job shop for delay. The mean minimum and maximum delay between the start of operations. In this paper we present a framework for solving job shop scheduling problem based on a disjunctive graph to modify the problem and solve it by the Memmetic algorithm.

Jansen [15] scheduling problem solved job shop under the assumption that the jobs have controllable processing time. That means you can reduce the processing time of work by paying some cost. Jansen presented two models. The first is the continuous model and the other is model reduction. The test could prove manifest that both of them can solve the approximation scheme is fixed polynomial time when the number of machines and the number of operations. Job shop scheduling problem with minimizing makespan is investigated.

Guinet [16] reduced the problem of job shop problems to flow precedence-constrained jobs. After that he extended the Johnson's rule solve this problem. He noted that the optimization of extended Johnson's rule is demonstrated by two state machines and efficient for about three or four machines job shop problems.

Drobouchevitch [17] presented two heuristics to solve a special case of the job shop scheduling. The case based on the assumption that each job consists of a maximum of two operations. One of which is to be processed in one machine m. While other operations should be performed in a single bottleneck machine. One of two heuristic ensures worst case performance ratio 3/2. It was also observed that these techniques can be applied to related problem, such as flow shop scheduling problem with parallel machines.

Ganesen [18] solved the special case of the problem of job shop scheduling. Minimum variance time competition restriction (CTV) adds to the problem. The lower limit of the CTV is developed for the problem. To solve this problem using programming approach backwards. To show performance programming approach backwards, the result is compared with programming approach forward. The study showed setback programming approach so performance for this special case of the problem of job shop scheduling. And two layers technique is a technique to solve the job shop scheduling problem

Pan [19] described mixed binary integer programming for reentrant job shop scheduling problem and solves the problem by using two layers technique. Ganesen [20] studied the problem of job shop scheduling with two goals. The first is to minimize the total absolute difference in completion time and the other is mean flow time. Programming backward scheduling technique was studied again. Moreover, static optimal technique used. In this study, we took 82 issues to consider. The result is a new benchmark for the problem.

Pham [21] solved special case of the problem of job shop scheduling also called the multi-mode block job shop scheduling problem. The problem is that from the hospital in order to allocate resources for hospital surgical cases. CPLEX was used to solve the problem. Because computation time limit, the model is capable of small and medium size of the problem. That suggests the study. Other research investigated meta-heuristics.

Watanabe [22] and Koonce [23] genetic algorithm used to solve the problem of programming workshop. Ganesen uses simulated annealing to solve the job shop scheduling [24], [25]. Some research uses neural network to select dispatching rules.

Research deals with job shop scheduling being solved by the hybrid algorithm between two Meta heuristics [26], [27], [28]. This paper focuses on the development of algorithms for solving job shop scheduling. The algorithm is designed taking into account machine availability constraint. The following section describes the details of a problem and the mathematical modeling of the problem. The following describes the machine availability constraint

Chapter 3

Methodology Adopted

Chapter 3

Methodology Adopted

Here we review the algorithms used for solving the job shop scheduling problems, the general job shop scheduling mathematical model without the machine availability constraint. In general, variable are as follows:

Let $t_{i,j}$ be start time of job j that is perform on machine i,

Let $f_{i,j}$ be finish time of job j that is performed on machine i,

Let $p_{i,j}$ be processing time of job j that is performed on machine i,

Let C_{max} be Makespan (finish time of latest job).

The objective of the problem is to minimize makespan. The mathematical model of job shop scheduling problem without machine availability constraint is shown below.

1. Min C_{max}
2. $t_{h,j} - t_{i,j} \geq p_{i,j}$
3. $C_{max} - t_{i,j} \geq p_{i,j}$
4. $t_{i,j} - t_{i,k} \geq p_{i,k}$ or $t_{i,k} - t_{i,j} \geq p_{i,j}$
5. $t_{i,j} \geq 0$
6. $t_{i,j} \geq r_i$
7. $t_{i,j} + p_{i,j} \leq d_j$

To ensure that the next step on machine *h* of job *j* begins after the finish time of the stage on machine *i* of job *j*, Equation 2 is used. Then equation 3 ensures that C_{Max} must be something more than the finish time of the last job. Equation 4 is used for sequencing jobs on machines. This equation means that only one job can be processed on a single machine once. Using Equation 5, the start time of the processes is not negative. At some point, a problem requires conditions of job released time. Equation 6 ensures that job should begin after the release time. The last constraint is used to control the work should be completed before the due date.

Gantt chart:

Devised by Henry Gantt in 1910s, Gantt chart is the representation type of bar chart used to represent a feasible schedule of a scheduling problem. Gantt chart also provides the details about the precedence of operations under taken by the Jobs in various machines.

Gantt chart is an apt medium for portraying a resultant schedule in a small problem, but a problem with large number of activities that is very difficult to represent the schedule. Gantt does not represent the relative size of work items or the total size of the project. Therefore, it becomes too tough in some cases to compare two projects with the same number of time of completion.

Chapter 4

Results

And

Discussions

Chapter 4

Results And Discussion

A C++ code was generated in Dev C++ using the above algorithm. The program was tested on a 4 job 3 machine problem.

	Job 1		Job 2		Job 3		Job 4	
	Operation	P_{ijk}	Operation	P_{ijk}	Operation	P_{ijk}	Operation	P_{ijk}
Machine 1	(1,1,1)	4	(2,2,1)	4	(3,3,1)	3	(4,3,1)	1
Machine 2	(1,2,3)	3	(2,1,2)	1	(3,2,2)	2	(4,1,2)	3
Machine 3	(1,3,3)	2	(2,3,3)	3	(3,1,3)	3	(4,2,3)	3

Table: 1 Problem involving four jobs and three machines

Is the problem, an operation with a triplet (i, j, k) denotes that operation j of job i requires machine k. P_{ijk} is the processing time of the j^{th} operation of i^{th} job on machine k.

The Program generated on C++ on Dev was executed and input values were provided to it

```
 JOB 1 DETAILS
Processing time for operation number 1 and Machine number 1 is : 4

Processing time for operation number 2 and Machine number 2 is : 3

Processing time for operation number 3 and Machine number 3 is : 2

 JOB 2 DETAILS
Processing time for operation number 1 and Machine number 2 is : 1

Processing time for operation number 2 and Machine number 1 is : 4

Processing time for operation number 3 and Machine number 3 is : 4

 JOB 3 DETAILS
Processing time for operation number 1 and Machine number 3 is : 3

Processing time for operation number 2 and Machine number 2 is : 2

Processing time for operation number 3 and Machine number 1 is : 3

 JOB 4 DETAILS
Processing time for operation number 1 and Machine number 2 is : 3

Processing time for operation number 2 and Machine number 3 is : 3

Processing time for operation number 3 and Machine number 1 is : 1
```

Figure 1, Input of Data

Result:

```
The scheduling is as follows

Schedule for machine 1
111   111   111   111   221   221   221   221   431   331   331   331   idle

Schedule for machine 2
212   412   412   412   122   122   122   322   322   idle  idle  idle  idle

Schedule for machine 3
313   313   313   idle  423   423   423   133   133   233   233   233   233
```

Figure 2, Resultant Schedule

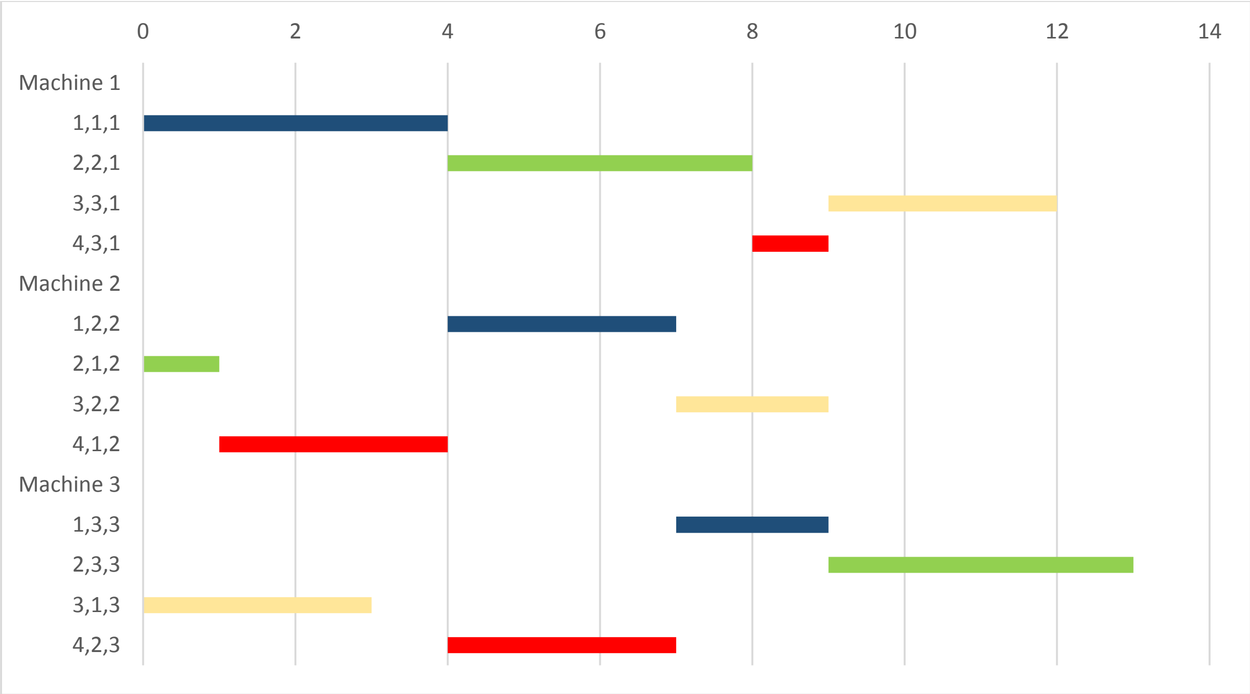

Figure 3, Gantt Chart of Given Schedule

Makespan of the given Schedule was found to be 13 units.

Chapter 5

Conclusion

Chapter 5

Conclusion

This current work focuses on the scheduling of jobs in the job shops. Jobs hob scheduling aims to minimize makespan time. The algorithm used was aimed at creating a mathematical model without machine availability constraint. The algorithm has been coded in DevC + +. The algorithm was effective in many problems. The schedules obtained have makespan value near to optimal.

Chapter 6

REFERENCES

Chapter 6

References

[1] Brucker P., 1995. Scheduling algorithms 2nd edn. Springer. Berlin Heidelberg New York.

[2] Garey M., et al, 1976. The complexity of flow shop and Job shop scheduling. Mathematical Operation Research 1, pp. 117-119.

[3] Brandimarte P., 1993. Routing and scheduling in a flexible job shop by taboo search. Annals of operation research, 41, 157-183

[4] Brucker P., 1994 A branch and bound algorithm for Jobshop problem. Discrete Applied Mathematics, 49:107-127

[5] Carlier J., Pison E., 1989.An algorithm for solving Job shop problem. Management Science, 35: 164-176

[6] Blazewickz J., 1996. The Job shop scheduling Problem: Conventional and new solutions techniques. European Journal of Operational Researchpp, 931-30

[7] Choi I. and Choi, D., 2002. "A local search algorithm for job shop scheduling problems with alternative operations and sequence dependent setups". Computer and industrial engineering, 2, 43-58

[8] Erscher J.F., Roubellat J.P., Verntes , 1976. Finding some essential characteristics of the feasible solutions for a scheduling problem. Operational research, 24, pp. 774-783

[9] Heurik E., Jarisch B. and Thole M., 1994. Tabu search for the job shop scheduling problem with multipurpose machines. Operation research Spectrum, 15, pp. 205-215

[10] Mastrolli M. and Gambardella C. M., 2000. Effective neighborhood functions for the flexible jobshop problem. Journal of scheduling, 3(1), pp. 3-20.

[11] Koonce D. A. and Tsai S. C., 2000. "Using data mining to find patterns in genetic algorithm to a job shop schedule", Computer and industrial engineering, 38, pp. 361-374.

[12] Holthaus O. and Rajendran C., 1990. “A comparative of study of dispatching rules in dynamic flow shops and job shops”. European journal of operational research , 116, pp. 156-170.

[13] Ohta H. and Nakatani T., 2006. “A heuristic job shop scheduling algorithm to minimize total holding cost of completed and in progress products subject to no tardy job”. International journal production economics, 11, pp. 19-29.

[14] Caumond A., Lacomme P., and Tchernev N., “A memmetic algorithm for job shop with time lag”. Computer and operational research.

[15] Jansen K., Mastrolilli M. and Oba R.S., 2005. “Approximation scheme for job shop scheduling problem with controllable processing time”. European journal of operational research, 1672, pp. 97-319.

[16] Guinet A., 2000. “Efficiency of reduction of job shop to flow shop problems”. European journal of operational research, 125, pp. 469-485.

[17] Drobouchevitch I.G. and Strusevich V.A., 2000. “Heuristic for the two stage job shop scheduling problem with a bottleneck machine”. European journal of operational research, 123, pp. 229-240.

[18] Ganesen V.K., Sivakumar A. I., and Srinivasan G., 2006. “Hierarchical minimization of completion time variance and makespan in job shops”. Computer and operations research, 33, pp. 1345-1367.

[19] Jasan C. H. and Chen J. S., 2005. “Mixed binary integer programming formulations for the reentrant job shop scheduling problem”. Computer and operational research, 32, pp. 1197-1212.

[20] Ganesen V. K. and Sivakumar A. I., 2006. “Scheduling in static job shops for minimizing mean flow time subject to minimum total deviation of job completion times”. International journal production economics, 103, pp. 633-647.

[21] Pham D.N. and Klingert A., 2008.” Surgical case scheduling as a generalized job shop scheduling problem”. Europian journal of operational research, 185, pp. 1011-1025.

[22] Watanabe M., Ida K., and Gen M., 2005. “A genetic algorithm with modified crossover operator and search area adaptation for the job-shop scheduling problem”. Computer and industrial engineering, 48, pp.743-752.

[23] Koonce D.A. and Tsai S.C., 2000. “Using data mining to find patterns in genetic algorithm solutions to a job shop schedule”. Computer and industrial engineering, 38, pp. 361-374.

[24] Ganesen V. K., Sivakumar A. I., 2006. “Hierarchical minimization of completion time variance and makespan in job shops. Computer and operational research, 33, pp.1345-1367.

[25] Ganesen V. K. and Sivakumar A. I., 2006. “Scheduling in static job shops for minimizing mean flow time subject to minimum total deviation of job completion times”. International journal production economics, 103, pp. 633-647.

[26] Bouri A. El and Shah P., 2006. “A neural network for dispatching rule selection in a job shop”. International journal advance manufacturing technology, 31, pp. 342-349.

[27] Dominic P. D. D., Kaliyamoorthy S., and Murugan R., 2004. “A conflict-based priority dispatching rule and operation-based approaches to job shops”. International journal advance manufacturing technology, 24, pp. 76-80.