OPTIMIZATION OF TOTAL WAITING TIME OF JOBS IN TWO STAGE FLOW SHOP SCHEDULING

A MINOR PROJECT

(Under CPE Scheme for Enrichment of Research Capabilities)

Submitted To

General Shivdev Singh Diwan Gurbachan Singh

KHALSA COLLEGE, PATIALA

In

MATHEMATICS

By

BHARAT GOYAL
Assistant Professor

P.G. DEPARTMENT OF MATHEMATICS

G.S.S.D.G.S. Khalsa College, Patiala

CERTIFICATE

I declare that this report entitled "**OPTIMIZATION OF TOTAL WAITING TIME OF JOBS IN TWO STAGE FLOW SHOP SCHEDULING"** submitted to Khalsa College Patiala is a record of original work carried out by me and has not formed the basis for the award of any other degree or diploma. In keeping with ethical practice due acknowledgements have been made wherever the findings of other researchers have been cited.

Bharat Goyal

Assistant Professor

P.G. Department Of Mathematics

G.S.S.D.G.S. Khalsa College, Patiala

ACKNOWLEDGEMENTS

As of course first thanks are to the Almighty whose blessings are always shower upon.

I express my sincere thanks to **Dr. Dharminder Singh Ubha,** Principal, G.S.S.D.G.S., Khalsa College, Patiala for his kind motivation and providing all necessary help and cooperation in completion of this project.

My deep thanks to my intellectual supervisor **Dr. Deepak Gupta,** Professor and Head, Department of Mathematics and Humanities, Maharishi Markandeshwar (Deemed to be University), Mullana, Ambala, Haryana for his continuous guidance, and strong support during the course of study.

I extend my heartfelt thanks to **Dr. Jasleen Kaur,** Deputy Principal, Khalsa College, Patiala, for her valuable support and cooperation during the course of work. I am extremely grateful to her for always standing in favor of me.

I owe a debt of gratitude to **Dr. Ravinderjit Singh**, Dean Research and Controller Examination, Khalsa College, Patiala for his invaluable suggestions, generous help and important acumen during the project.

I acknowledge with thanks the encouragement, the motivation that I received from **Dr. Gurmeet Singh**, Khalsa College Patiala .

I convey my thanks to all other members of the Department of Mathematics, Khalsa College, Patiala for their inspiration and proficient support.

I would like to thank all the members of the administrative staff, Khalsa College Patiala, for their cooperation and help whenever needed.

The heavenly blessings of my father Late **Sh. Varinder Pal Goyal** and the earthly blessing of my mother **Smt. Saroj Goyal** gave me spiritual strength in completing my research work.

I owe my deepest gratitude towards my better half **Mrs. Silkina Goyal** and my son **Rachit Goyal** for their invaluable support. Their infallible love and support has always been my strength. I am always grateful to the patience and sacrifice they have made all the time.

As always it is impossible to mention everybody who had an impact to this work however there are those whose spiritual support is even more important. It is a great pleasure to thank everyone who supported me in any respect during the completion of course of my research work.

(Bharat Goyal)

TABLE OF CONTENTS

Description **Page Number**

Description	**Page Number**

LIST OF PUBLICATIONS

The entire report is based on the following research papers published in Journals of International repute.

1. **Two Stage Specially Structured Flow Shop Scheduling Including Transportation Time Of Jobs And Probabilities Associated With Processing Times To Minimize Total Waiting Time Of Jobs,** *Journal of Emerging Technologies and Innovative Research, Vol.5, No. 6, pp. 431-439* (2018).

2. **Specially Structured Flow Shop Scheduling in Two Stage with Concept of Job Block and Transportation Time to Optimize Total Waiting Time of Jobs,** *International Journal of Engineering and Technology (**Scopus Indexed)**, Vol.10, No.5, pp. 1273-1284* (2018).

LIST OF TABLES

Chapter - 1

INTRODUCTION AND LITERATURE SURVEY

1.1 OPERATIONS RESEARCH (OR)

OR is the examination of the operations to be carried out for the achievement of optimum of the objectives of an organization or for its enhancement. This discipline makes available and uses scientific methodology in the investigation for optimal solutions, such as supporting decision-making processes, as far as decision-making is concerned optimal and in systems that originate in actual life.

In OR, the following essential characteristics stand out:

- A well-built orientation to Systems Theory.
- The involvement of interdisciplinary teams.
- The relevance of the scientific method in support of decision making.

Based on these properties, a possible definition is

"Operations Research is the application of the scientific technique by interdisciplinary teams to problems that they incorporate the control and management of organized systems (man-machine); with the intention of finding solutions that better serve the objective of the system (organization) as a whole, enclosed in decision-making processes."

1.2 STUDY OF OPTIMIZATION TECHNIQUES

There is a large variety of activities in the day to day life that can be described as systems, from physical systems such as a industrial plant to conjectural entities such as economic models. The efficiency of operation of those systems usually necessitates an effort to optimize quite a few parameters that measure the presentation of the system. Sometimes, these factors are quantified and represented as algebraic variables. One must find values for those variables, which benefits the system such as maximize the profit or minimize the expenses or losses or production times etc. It is assumed that the variables depend on certain factors. Some of these factors are sometimes under the control (at least partially) of the analyst responsible for system performance.

The process of managing the scarce resources of a system is usually divided in six phases:

(i) **Mathematical Analysis of the System:** It involves the scope of the considered problem. The constraints involved, determination of the objective are well thought of.

(ii) **Construction of a Mathematical Model:** In this stage the problem is prepared as a mathematical model. The models possibly will be simulation, heuristic, mathematical which can wrap up the problem.

(iii) **Manipulation of the Model:** The solutions of the problem are carried out by employing the optimization algorithms.

(iv) **Verification of the Model:** In this step we validate the solution of the model. The solution is matched up with the accessible way outs already found for justification.

(v) **Implementation of the selected solution:** The verified results of the model are then altered into comprehensible operating directions.

(vi) **Working of the System:** The working of the system under this phase is checked. The alterations made in the performance are well thought of in this stage.

1.3 FLOW SHOP SCHEDULING (FSS)

Scheduling may be defined as the problem of deciding when to execute a given set of activities, subject to chronological constraints and resources capacities, in order to optimize some function. A Flow shop problem exists when all the jobs share the same processing order on all the machines. In Flow shop, technological constraints demand that the jobs pass between the machines in the same order. Hence there is natural sequence of the machines characterized by the technological constraints for each and every job in flow shop. The flow shop contains **n** different machines arranged in series on which a set of **m** jobs are to be processed. Each of the **m** jobs requires **n** operations and each operation is to be performed on a separate machine. The flow of the work is unidirectional; thus every job must be processed through each machine in a given prescribed order. The general **m** jobs, **n** machine flow shop scheduling is quite formidable. Consider an arbitrary sequence of jobs on each machine, there are $(n!)^m$ possible schedules which poses computational difficulties. With the aim to reduce the number of possible schedules it is reasonable to assume that all machines process jobs in the same order.

Today's large-scale markets and instantaneous communications mean that clients expect high-quality goods and services when they require them, where they require them. Organizations, whether public or private, need to provide these products and services as effectively and efficiently as possible. The basic purpose of the study is to assist the manager to find an optimal way out to a decision problem by systematically analyzing a variety of possible alternatives. The manager, usually, takes the help of the Operations Research (O.R.) analyst to analyze the problem through quantitative method and to provide appropriate information about the managerial problem. Thus, managerial effectiveness can be improved by using these techniques even though the techniques by themselves are not an answer to a given problem. The managers can mingle their past experience, inkling and quantitative analysis to make the best possible decision. In fact Operations Research is the quantitative decision making approach suggest an optimal solution to the problems which is in the best interest of organization. O.R. encompasses a wide range of problem-solving techniques and methods applied in the pursuit of improved decision-making and efficiency, such as simulation, mathematical optimization, queuing theory and other stochastic-process models, expert systems, decision analysis, and the analytic hierarchy process.

1.3.1 Assumptions in Flow Shop Scheduling

1. There being no more than one machine to perform the same action.

2. A machine can process only a job in a moment.

3. Technological constraints are well-known and unchanging.

4. Each job is an entity; hence it is not possible to process two operations in one job simultaneously.

5.There is no interruption, i.e. each process once started must be finished before another process can do so on the same machine.

6. Each job has one and only one operation on every machine, so all jobs consitute operations not greater than the counting of machines.

7.There is no recirculation.

8. Process times are independent of the sequence followed, which excludes setting times in the machines according to the sequence of work considered or transport times between machines.

1.3.2 Variables Used in FSS Problem

In the FSS Problem the jobs are merely recognized by the integers 1, 2, 3, …, n. The appropriate characteristics of j^{th} job that are specified as a component of the problem are symbolized by the variables defined as follows:

- **r_j:** It is the time at which job is ready to process or job arrives in the shop. At this time the job is sent for the processing. It is imperative as the earliest time that processing of the initial process of the job starts.

- **d_j:** It is the time at which the job leave the process. The very last operation should be finished by this time.

- **$m_{i,j}$:** It is the time on i^{th} machine that is required to process the j^{th} job. It is normally assumed that whatever the time taken by the machines to set-up or transportation time etc. all is incorporated with the time of processing the jobs.

This means that the time length required to prepare the machine for a particular operation does not depend on what the machine did last. However there are situations where the set-up time, transportation time etc. can be recognized separately.

- **$W_{i,j}$:** The time that was spent in waiting by the j^{th} job on the i^{th} machine after completion of $(j-1)^{th}$ job before beginning of j^{th} job.

- **C_j:** The time of completion of j^{th} job.

  $C_j = r_j + W_{1,j} + m_{1,j} + W_{2,j} + m_{2,j} + \ldots\ldots + W_{n,j} + m_{n,j}$

- **F_j:** It is the Flow time of j^{th} job. It is the duration of spending the time in the shop by the j^{th} job.

  $F_j = W_{1,j} + m_{1,j} + W_{2,j} + m_{2,j} + \ldots\ldots + W_{n,j} + m_{n,j}$

  $= C_j - r_j$

- **L_j:** Lateness of the j^{th} job.

  $L_j = C_j - d_j$

- **T_j:** Tardiness of the j^{th} job.

  $T_j = \text{Max}(0, L_j)$

- **E_j:** Earliness of the j[th] job.

  T_j= Max (0, $-L_j$)

- **C_{max}:** It is the total elapsed time prior to the finishing of specified set of jobs on number of machines. Moreover it is known by the name makespan.

  $C_{max} = \max \{C_j\}; j = 1, 2, 3, \ldots, n.$

- **W_j:** It is the total of the times spent in waiting by j[th] job for processing.

  $W_j = \sum_i W_{i,j}$

- **W:** It is the sum of all the times spent in waiting by all the jobs.

1.4 SOLUTION METHODOLOGY

Following are the techniques developed in order to solve the different types of production scheduling problem. Mathematical techniques used in Flow- Shop Scheduling are described as follows:

1.4.1 Exact Methods

1.4.1.1 Branch and Bound Approach

Branch and bound technique is most widely used in scheduling. It is an enumeration technique and is applied to optimization. It is a useful method of solving many combinatorial problems and is a general purpose strategy for reduced enumeration. As its name applies the approach of two fundamental procedures.

- **Branching**

It is the process of partitioning a large number into two or more sub problems.

- **Bounding**

It is the process of calculating a lower bound on the optimal solution of given sub problems

The branching procedure replaces an original problem by asset of new problem that are; mutually exclusive and exhaustive subprograms of the original problem, Partially solved versions of the original problem are smaller problems than the original problem.

1.4.2 Approximate Methods

1.4.2.1 Heuristic approach

A heuristic technique often called simply a heuristic, is any approach to problem solving, learning, or discovery that employs a practical method not guaranted to be optimal or perfect, but sufficient for the immediate goals. Where finding an optimal solution is impossible or impractical, heuristic methods can be used to speed up the process of finding a satisfactory solution. Thus, heuristic methods tend to be ad hoc in nature. That is, each method is usually is designed to fit a specific problem type rather than a variety of applications.

1.4.2.2 Metaheuristic

In computer science and mathematical optimization, a metaheuristic is a higher-level procedure or heuristic designed to find, generate, or select a heuristic (partial search algorithm) that may provide a sufficiently good solution to an optimization problem, especially with incomplete or imperfect information or limited computation capacity. Metaheuristics sample a set of solutions which is too large to be completely sampled. Metaheuristics may make few assumptions about the optimization problem being solved, and so they may be usable for a variety of problems.

1.5 THE CLASSES P AND NP

Each and every problem for which the algorithms that have polynomial time performance are considered in class P. The set of problems for which algorithms that have exponential performance are considered in class NP. It is clear that class P is a subset of class NP. If one has a polynomial time performance of algorithm for a problem it can always be inflated inefficiently so that it takes exponential time. Also, occasionally a problem originally in NP, but not in P, is moved into P, as someone with flush of insight discovers a polynomial time algorithm.

The set of all judgment problems whose solutions be capable of confirmed in polynomial time are considered to be NP-complete (NPC). NP-complete class is contained in class NP. A problem *p* in NP also belongs to the class NPC if and only if each additional problem in NP in polynomial time can be altered into *p*.

It can be said that a problem is NP-hard when it is shown that any solution algorithm has a running time increases, in the worst cases, exponentially with the problem size. Which problem is listed in this category does not mean that cannot be solved, but should propose solution algorithms that exploit efficiently the same mathematical structure for which solutions are mostly the instances of the problem in time relatively small execution. For problems, that are NP-Hard, which are mostly arising in scheduling; at hand there are no easy solutions. FSS problems like many others in the field of sequencing tasks are difficult to solve and in the literature are classified as difficult NP.

The merely techniques available are those of implicit (or explicit) details. This may take an excessive amount of calculation. Certainly NP-Hard scheduling problems are for all realistic objectives. If a problem is large and NP-Hard, then one must consider using heuristic method. But it should be emphasized that NP-Hardness is not an enough cause to the choice of heuristic method. It must also be so large that enumerative method is untraceable.

1.6 LITERATURE SURVEY

In this subsection investigations are presented regarding the Flow Shop Scheduling.

The Johnson's [1] algorithm for FSS problem in two and three stage to lessen the total elapsed time is popular among the analytical approaches that are used for solving two and three stage scheduling problem. Jackson, J.R., [3] generalized Johnson's method [1] for capably solving certain two stage production scheduling problems together with cases in which a number of jobs need only one stage and also the jobs needed two stages and may possibly need the machines in both of the possible orders. Gupta, J.N.D., [10] with the intention of minimizing the total throughput time in which every jobs finish processing on each and every machine, has developed a number of straightforward algorithms however the processing times are not wholly arbitrary, but stands with a definite relationship to one another. Various Heuristics are presented by Campbell, H.A., et.al.[8], Rajendran, C., et.al.[23], [24], Lee, Y.H., et.al.[34], Wang, X., et.al.[39] to lessen the total elapsed time. Nawaz, M., et.al.[18] presented an efficient heuristic known as NEH algorithm for m-machine, n jobs FSS problem.

It has been observed by many researchers that scheduling models are also affected by various parameters. Maggu, P.L., et.al.[11] made an effort to widen the study by initiating the notion of equivalent job for job block. Mitten, L.G., [5] studied FSS in two stage with arbitrary time lags. Sometimes the machines are placed at far away distances so that the time consumed in transferring the jobs from one machine to other machine is a parameter that becomes also significant in scheduling theory. Various Researchers Maggu, P.L., et.al.[16],[19], Panwalker, S.S.,[22], Lee, C.Y., et.al.[30] has considered the transportation times as a significant parameter in scheduling theory. Maggu, P.L., et.al.[17] widened the study by considering the transportation times as well as the concept of job-block. Singh, T.P., [20], Belwal, O.K., et.al.[40] studied the two stage scheduling models with parameters job-block, transportation time of jobs, and interval of break down of machines. Scheduling jobs on machines was further investigated wisely by many researchers while considering the various objective functions.

Bhatnagar, V., et. al.[13] investigates the n x 2 FSS models with the intention to optimize the total of the waiting times of all the jobs. The heuristics to minimize the cost which is consumed while hiring the machines on rent are presented by Bagga, P.C., [6],[29], in two stage and three stage. Gupta, D., Singh, T.P., et.al.[32], studied scheduling models where probabilities are associated with processing times alongwith the job- block concept, in two stage with the intention to lessen the cost consumed when hiring the machines on rent. Singh, T.P., et.al.[37] further extended the study with the consideration of m machines. Gupta, D., et.al.[42], [43], [44], [45] with the objective of lessening the hiring cost of machines studied the specially structured scheduling models in two stage where the times of processing the jobs are not altogether arbitrary, but stands with a specific relationship with each other with the consideration of various parameters.

Oguz, C., et.al. [31], addresses the Flow-Shop Flexible problem with the main feature that each job in each stage can be processed by one or more machines simultaneously, but each processor is able to process merely one job at the same time, always considering the goal of lessening the makespan. This approach allows relax the classic problem of Flow-Shop Flexible to admit that a job can be dealt out by two or more machines at a time. It was demonstrated by computational evidence that the relaxed problem is now possible to be cracked in performance of polynomial time, but the introduction of precedence constraint makes it NP-hard.

Chapter - 2

SPECIALLY STRUCTURED 2-MACHINE, n-JOB FSS MODELS WITH TRANSPORTATION TIME TO MINIMIZE TOTAL WAITING TIME OF JOBS

The problem of scheduling in linear production systems, flow shop, has been a matter of great significance in the research of operations, which looks for to launch the optimal programming of jobs in machines inside a production process. Flow shop scheduling where the machines are prearranged in order that the flow of each and every of the products that are processed in them is unidirectional.

There are m machines and there may be jobs that have fewer operations than m. One of the farm duties is to determine the optimal sequencing of jobs, this task is complicated, as for the reason nature of the problem is combinatorial, only the minority can be solved exactly. In the production lines under FSS, where the sequence of all the products is the same and in one direction only, the problem will be to conclude the proper order in which different jobs have to be arranged without compromising on the optimal function.

Seldom are the machines placed at distances so in this kind of situation the time to transport the jobs is an important aspect that can't be ignored. Many Researchers in the past have observed the significance of transportation time of jobs. Chikhi, N.[41], developed his theory in two stage FSS with the consideration of times of transportation of jobs. Further the studies are developed by considering the transportation times with various objectives such as lessening the total elapsed time, optimization of the cost of hiring of the machines on rent, minimizing total tardiness. Gupta, D., et.al.[43] with the intention of lessening the cost of hiring the machines on the rent observed the specially structured scheduling models in two stage with the time of transportation.

In Section 2.1 we shall develop an algorithm to lessen the total of the times spent in waiting by the jobs in which the time to transport jobs from first machine to second machine has been well thought of.

In section 2.2 the times of processing the jobs on machines are associated with probabilities and the time to transport the jobs from 1st machine to 2nd machine is considered separately with the same objective of section 2.1.

SECTION 2.1

MINIMIZATION OF TOTAL WAITING TIME OF JOBS IN TWO STAGE SPECIALLY STRUCTURED FSS MODEL WITH TRANSPORTATION TIME OF JOBS

2.1.1 Problem Formulation

Assuming the two machines P and Q are processing n jobs in the sort PQ. P_i and Q_i are the times to process i^{th} job, $1 \leq i \leq n$, on machine P and Q respectively. The times of transporting i^{th} job from machine P to machine Q are denoted by $t_{i,1\rightarrow 2}$. The fictitious machines X, Y with equivalent times X'_i, Y'_i correspondingly to process the jobs are defined by Maggu, P.L., et.al.[16], [19], Gupta,D., et.al.[43] are given as

$X'_i = P_i + t_{i,1\rightarrow 2}$ (2.1.1)

$Y'_i = Q_i + t_{i,1\rightarrow 2}$

Satisfying the processing time structural relationship

Max $X'_i \leq$ Min Y'_i (2.1.2)

Job	Machine P	Transportation Time	Machine Q
i.	P_i	$t_{i,1\rightarrow 2}$	Q_i
1.	P_1	$t_{1,1\rightarrow 2}$	Q_1
2.	P_2	$t_{2,1\rightarrow 2}$	Q_2
3.	P_3	$t_{3,1\rightarrow 2}$	Q_3
….	…..	…..	…..
n.	P_n	$t_{n,1\rightarrow 2}$	Q_n

TABLE 2.1.1 Processing Time Model Formulation

Our intention is to find sequence of jobs lessening the total of the times exhausted in waiting by all the jobs.

2.1.2 Assumptions

1) P and Q machines are processing n jobs, firstly on machine P then on machine Q and no passing is permissible.

2) The machines cannot process the same job tagether in the same span of time.

3) The processing of the machine cannot be interrupted until a job which is in execution can't be completed.

4) Break down interval and set-up time of machines is negligible.

2.1.3 Lemma. Assuming the two machines P and Q are processing n jobs in the sort PQ. P_i and Q_i are the times to process i^{th} job, $1 \leq i \leq n$, on machine P and Q respectively. The times of transporting i^{th} job from machine P to machine Q are denoted by $t_{i,1\to 2}$. Equivalent times of processing of i^{th} job on fictitious machines X, Y are defined as in equations (2.1.1) satisfying structural relationship defined in equations (2.1.2) then for the n job sequence

S: $\sigma_1, \sigma_2, \sigma_3, \dots \dots \dots \sigma_n$

$T_{\sigma_n Y}$, the time of completion of n^{th} job on machine Y is given by

$$T_{\sigma_n Y} = X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_n} \qquad (2.1.3)$$

Proof Applying the induction principle on 'n'.

Consider the statement $S(n): T_{\sigma_n Y} = X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_n}$

$T_{\sigma_1 X} = X'_{\sigma_1}$

$T_{\sigma_1 Y} = X'_{\sigma_1} + Y'_{\sigma_1}$

For n=1, S (n) holds.

Suppose S (n) holds for k jobs, i.e. for n=k

$T_{\sigma_k Y} = X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_k}$

Now,

$$T_{\sigma_{k+1} Y} = \mathrm{Max}\left(T_{\sigma_{k+1} X}, T_{\sigma_k Y}\right) + Y'_{\sigma_{k+1}}$$

$$= \mathrm{Max}\left(X'_{\sigma_1} + X'_{\sigma_2} \dots + X'_{\sigma_{k+1}}, X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_k}\right) + Y'_{\sigma_{k+1}}$$

As Max $X'_i \leq$ Min Y'_i

Hence

$T_{\sigma_{k+1} Y} = X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_k} + Y'_{\sigma_{k+1}}$

S(n) holds true for n=k+1.

Hence S(n): $T_{\sigma_n Y} = X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_n}$ stands true for every n.

2.1.4 Lemma For n- job sequence S: $\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n$, W_{σ_i}(say) is the time spent in waiting by i[th] job on machine Y

$$\left.\begin{aligned} & W_{\sigma_1} = 0 \\ & W_{\sigma_k} = X'_{\sigma_1} + \textstyle\sum_{r=1}^{k-1} x_{\sigma_r} - X'_{\sigma_k} \\ & \text{Where } x_{\sigma_r} = Y'_{\sigma_r} - X'_{\sigma_r}, \quad \sigma_r \in (1,2,3,\dots,n) \end{aligned}\right\} \qquad (2.1.4)$$

And the notations have usual meaning as defined in 2.1.3 Lemma

Proof. $W_{\sigma_1} = 0$

$$W_{\sigma_k} = \text{Max}\left(T_{\sigma_k X}, T_{\sigma_{k-1} Y}\right) - T_{\sigma_k X}$$

$$= \text{Max}\left(X'_{\sigma_1} + X'_{\sigma_2} \dots + X'_{\sigma_k}, X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_{k-1}}\right) - (X'_{\sigma_1} + X'_{\sigma_2} \dots + X'_{\sigma_k})$$

$$= X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_{k-1}} - X'_{\sigma_1} - X'_{\sigma_2} \dots - X'_{\sigma_k}$$

$$= X'_{\sigma_1} + \sum_{r=1}^{k-1} (x_{\sigma_r}) - X'_{\sigma_k}$$

2.1.5 Theorem For n- job sequence S: $\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n$.The total of the times spent by jobs in waiting W (say) is given by

$$W = nX'_{\sigma_1} + \textstyle\sum_{r=1}^{n-1} (n-r) x_{\sigma_r} - \sum_{i=1}^{n} X'_i \qquad (2.1.5)$$

Where $x_{\sigma_r} = Y'_{\sigma_r} - X'_{\sigma_r}$, $\sigma_r \in (1,2,3,\dots,n)$ and the notations have usual meaning as defined in 2.1.3 Lemma

Proof. Using 2.1.4 Lemma one can get

$W_{\sigma_1} = 0$

When k = 2

$$W_{\sigma_2} = X'_{\sigma_1} + \sum_{r=1}^{1} x_{\sigma_r} - X'_{\sigma_2}$$

$$= X'_{\sigma_1} + x_{\sigma_1} - X'_{\sigma_2}$$

When k = 3

$$W_{\sigma_3} = X'_{\sigma_1} + \sum_{r=1}^{2} x_{\sigma_r} - X'_{\sigma_3}$$

$$= X'_{\sigma_1} + x_{\sigma_1} + x_{\sigma_2} - X'_{\sigma_3}$$

Carrying on the process for k = n,

$$W_{\sigma_n} = X'_{\sigma_1} + \sum_{r=1}^{n-1} x_{\sigma_r} - X'_{\sigma_n}$$

$= X'_{\sigma_1} + x_{\sigma_1} + x_{\sigma_2} + \cdots + x_{\sigma_{n-1}} - X'_{\sigma_n}$

Hence total waiting time

$W = W_{\sigma_1} + W_{\sigma_2} + W_{\sigma_3} + \cdots + W_{\sigma_n}$

$$W = 0 + \left(X'_{\sigma_1} + x_{\sigma_1} - X'_{\sigma_2}\right) + \left(X'_{\sigma_1} + x_{\sigma_1} + x_{\sigma_2} - X'_{\sigma_3}\right) + \cdots + (X'_{\sigma_1} + x_{\sigma_1} + x_{\sigma_2} + \cdots + x_{\sigma_{n-1}} - X'_{\sigma_n})$$

$$W = nX'_{\sigma_1} + \sum_{r=1}^{n-1} (n-r) x_{\sigma_r} - \sum_{i=1}^{n} X'_{\sigma_i}$$

2.1.6 Theorem For a natural number 'm' and real numbers $x_1, x_2, x_3, \ldots \ldots x_m$ if $x_{\sigma_1} \leq x_{\sigma_2} \leq x_{\sigma_3 \leq} \ldots \ldots \leq x_{\sigma_m} \Rightarrow m x_{\sigma_1} + (m-1)x_{\sigma_2} + (m-2)x_{\sigma_3} + \cdots + x_{\sigma_m}$ is minimum for $\sigma \in S_m$, group of permutation on m-symbols.

Proof: Applying Induction hypothesis on m:

The result holds trivially in the case m=1

Suppose the result stands true for less than m real numbers

For $x_{\sigma_1} \leq x_{\sigma_2} \leq x_{\sigma_3 \leq} \ldots \ldots \leq x_{\sigma_m}$

$m x_{\sigma_1} + (m-1)x_{\sigma_2} + (m-2)x_{\sigma_3} + \cdots \ldots + x_{\sigma_m}$

$$= (m-1)x_{\sigma_1} + (m-2)x_{\sigma_2} + (m-3)x_{\sigma_3} + \cdots \ldots + x_{\sigma_{m-1}} + \sum_{k=1}^{m} x_{\sigma_k}$$

As $\sum_{k=1}^{m} x_{\sigma_k}$ is constant, by induction hypothesis $m x_{\sigma_1} + (m-1)x_{\sigma_2} + (m-2)x_{\sigma_3} + \cdots \ldots + x_{\sigma_m}$ is minimum for $\sigma \in S_m$.

2.1.7 Algorithm for the Formulated Problem

Step 1: Find the equivalent times X'_i and Y'_i of processing the jobs on fictitious machines X and Y correspondingly as defined in equations (2.1.1)

Step2: Validate the processing time structural relationship $\operatorname{Max} X'_i \leq \operatorname{Min} Y'_i$ as described in equation (2.1.2).

Step3: Evaluate the values of $x_i = Y'_i - X'_i$ in the table 2.1.2.

Job	Machine X	Machine Y	$Y'_i - X'_i$
i	X'_i	Y'_i	x_i
1.	X'_1	Y'_1	x_1
2.	X'_2	Y'_2	x_2
3.	X'_3	Y'_3	x_3
….	….	….	…..
n.	X'_n	Y'_n	x_n

TABLE 2.1.2

Step 4: Assemble the jobs in ascending order of x_i. Let the sequence thus found be $(\sigma_1, \sigma_2, \sigma_3, \dots\dots\dots \sigma_n)$

Step 5: Check for $\min\{X'_i\}$

If $X'_{\sigma_1} = \min\{X'_i\}$ in that case sequence obtained in 4th step is the required sequence for scheduling.

If $X'_{\sigma_1} \neq \min\{X'_i\}$ then go to 6th step.

Step 6: Find the sequences $S_1, S_2, S_3, \dots\dots, S_n$. S_1 is the sequence attained in 4th step, sequences $S_i, 2 \leq i \leq n$ can be obtained by taking ith job in the sequence S_1 to the 1st position and considering respite of the sequence same.

Step 7: Calculate the total of the times spent in waiting W by the jobs for the sequences $S_1, S_2, S_3, \dots\dots, S_n$ using the formula defined in equation (2.1.5).

The sequence with minimum W is the requisite sequence for scheduling.

2.1.8 Numerical Illustration

Consider the processing of 5 jobs in flow shop on two machines P and Q. P_i and Q_i are the times to process ith job, $1 \leq i \leq 5$, on machine P and Q respectively. The time to transport ith job from machine P to machine Q is denoted by $t_{i,1\to2}$. Assume the values of the processing time matrix in table 2.1.3.

Job	Machine P	Transportation time	Machine Y
i	P_i	$t_{i,1\to 2}$	Q_i
1.	5	4	15
2.	8	3	21
3.	12	2	24
4.	11	4	19
5.	14	2	23

TABLE 2.1.3

Our aspiration is to obtain sequence of jobs, lessening the total of the times spent in waiting by all the jobs.

Solution

As per step 1- Finding the equivalent times X_i' and Y_i' of processing the jobs on fictitious machines X and Y respectively using the equations (2.1.1).

Job	Machine X	Machine Y
i.	X_i'	Y_i'
1.	9	19
2.	11	24
3.	14	26
4.	15	23
5.	16	25

TABLE 2.1.4

As per step 2- Processing time structural relationship Max X_i'=9≤ Min Y_i'=19 as defined in equation (2.1.2) is satisfied.

As per step 3- The values $x_i = Y_i' - X_i'$ are evaluated in the table 2.1.5.

Job	Machine X	Machine Y	x_i
i	X_i'	Y_i'	$Y_i' - X_i'$
1.	9	19	10
2.	11	24	13
3.	14	26	12
4.	15	23	8
5.	16	25	9

TABLE 2.1.5

As per step 4- Assembling the jobs in ascending order of x_i.

The sequence thus found is 4, 5, 1, 3, 2.

As per step 5- $\text{Min}\{X'_i\}$ =9$\neq X'_4$. So moving on to the 6th step.

As per step 6- Considering the possible sequences:

S_1:4, 5, 1, 3, 2

S_2:5, 4, 1, 3, 2

S_3:1, 4, 5, 3, 2

S_4:3, 4, 5, 1, 2

S_5:2, 4, 5, 1, 3

As per step 7- Calculate the total of the times spent in waiting by all the jobs for the sequences S_1, S_2, S_3, S_4, S_5 using the equations (2.1.5)

Here, $\sum_{i=1}^{5} X'_i = 65$

For the sequence S_1:4, 5, 1, 3, 2

W =101

For the sequence S_2:5, 4, 1, 3, 2

W =107

For the sequence S_3:1, 4, 5, 3, 2

W =74

For the sequence S_4:3, 4, 5, 1, 2

W =105

For the sequence S_5:2, 4, 5, 1, 3

W =94

Hence sequence S_3: 1, 4, 5, 3, 2 is the required sequence for scheduling with waiting time 74.

SECTION 2.2

SPECIALLY STRUCTURED FSS MODEL IN TWO STAGE INCLUDING TRANSPORTATION TIME OF JOBS AND PROBABILITIES ASSOCIATED WITH PROCESSING TIMES TO OPTIMIZE TOTAL WAITING TIME OF JOBS

2.2.1 Problem Formulation

Assuming the two machines P and Q are processing n jobs in the sort PQ. P_i and Q_i are the times to process i^{th} job, $1 \leq i \leq n$, on machine P and machine Q correspondingly. The probabilities allied with the times of processing are denoted by p_i and q_i such that $\sum p_i = \sum q_i = 1$ on machine P and Q correspondingly. Expected times to process the jobs on machine P and Q as defined by Gupta, D., et.al. [44], Singh, T.P., et.al.[32], are given as

$$P_i' = P_i \times p_i \quad (2.2.1)$$

$$Q_i' = Q_i \times q_i$$

The time to transport i^{th} job from machine P to machine Q is denoted by $t_{i,1\to2}$. The fictitious machines X, Y with equivalent times X'_i and Y'_i to process the jobs are defined by Maggu, P.L., et.al.[16], [19], Gupta, D. et.al.[43] are given as

$$X_i' = P'_i + t_{i,1\to2} \quad (2.2.2)$$

$$Y_i' = Q'_i + t_{i,1\to2}$$

Satisfying the processing time structural relationship

$$\text{Max } X_i' \leq \text{Min } Y_i' \quad (2.2.3)$$

Job	Machine P		Transportation Time	Machine Q	
i.	P_i	p_i	$t_{i,1\to2}$	Q_i	q_i
1.	P_1	p_1	$t_{1,1\to2}$	Q_1	q_1
2.	P_2	p_2	$t_{2,1\to2}$	Q_2	q_2
3.	P_3	p_3	$t_{3,1\to2}$	Q_3	q_3
….	…..	…..	…..	…..	…..
n.	P_n	p_n	$t_{n,1\to2}$	Q_n	q_n

TABLE 2.2.1: Processing Time Model Formulation

Then our intention is to attain a sequence in which the total of the times exhausted in waiting by all the jobs is minimum.

2.2.2 Assumptions

1) P and Q machines are processing n jobs, firstly on machine P then on machine Q and no passing is permissible.

2) The machines cannot process the same job together in the same span of time.

3) The processing of the machine cannot be interrupted until a job which is in implementation cannot be completed.

4) Break down interval and set up times of machines are negligible.

5) $\sum p_i = \sum q_i = 1$

2.2.3 Algorithm for the Formulated Problem

Step 1: Find the equivalent times X_i' and Y_i' of processing the jobs on fictitious machines X, Y respectively using the equations (2.2.1), (2.2.2).

Step2: Validate the processing time structural relationship as defined in equation (2.2.3).

Step 3: Find $x_i = Y_i' - X_i'$ and write the values in the table 2.2.2

Job	**Machine X**	**Machine Y**	$\mathbf{x_i}$
i.	$\mathbf{X_i'}$	$\mathbf{Y_i'}$	$\mathbf{Y_i' - X_i'}$
1.	X_1'	Y_1'	x_1
2.	X_2'	Y_2'	x_2
3.	X_3'	Y_3'	x_3
….	…..	…..	…..
n.	X_n'	Y_n'	x_n

TABLE 2.2.2

Step 4: Assemble the jobs in ascending order of x_i. Let the sequence of jobs thus found be $(\sigma_1, \sigma_2, \sigma_3, \dots \dots \dots \sigma_n)$.

Step 5: Check for min{ X_i' }

If $X_{\sigma_1}' = \min\{ X_i' \}$ in that case sequence obtained in 4th step is the required sequence for scheduling.

If $X_{\sigma_1}' \neq \min\{ X_i' \}$ then go to 6th step.

Step 6: Consider sequences $S_1, S_{2,} S_3, \ldots \ldots, S_n$. S_1 is the sequence attained in 4th step, Sequences $S_i, 2 \le i \le n$ can be attained by taking ith job in the sequence S_1 to the 1st position and taking same respite of the sequence.

Step 7: Evaluate the total waiting time W for sequences $S_1, S_{2,} S_3, \ldots \ldots, S_n$ using the equations (2.1.5)

The sequence with minimum W is the required optimal sequence for scheduling.

2.2.4 Numerical Illustration

Consider the processing of 5 jobs in Flow Shop on two machines P and Q. P_i and Q_i are the times to process ith job, $1 \le i \le 5$, on machine P and Q correspondingly. The probabilities allied with the times of processing are denoted by p_i and q_i on machine P and Q correspondingly. The time to transport ith job from machine P to machine Q is denoted by $t_{i,1\to 2}$. Assume the values of the processing time matrix in table 2.2.3

Job	**Machine P**		**Transportation time**	**Machine Q**	
i	$\mathbf{P_i}$	$\mathbf{p_i}$	$\mathbf{t_{i,,1\to 2}}$	$\mathbf{Q_i}$	$\mathbf{q_i}$
1.	6	0.2	4	12	0.2
2.	7	0.2	3	21	0.2
3.	12	0.2	2	34	0.2
4.	11	0.3	3	22	0.2
5.	13	0.1	2	24	0.2

TABLE 2.2.3

Our intention is to attain a sequence of jobs lessening the total of times spent in waiting by all the jobs.

As per step 1- Finding the equivalent times X_i' and Y_i' of processing the jobs on fictitious machines X & Y respectively as defined in equations (2.2.1), (2.2.2).

Job	Machine X	Machine Y
i.	$\mathbf{X'_i}$	$\mathbf{Y'_i}$
1.	5.2	6.4
2.	4.4	7.2
3.	4.4	8.8
4.	6.3	7.4
5.	3.3	6.8

TABLE 2.2.4

As per step 2: The processing time structural relationship Max $X'_i = 6.3 \leq$ Min $Y'_i = 6.4$ as defined in equation (2.2.3) is satisfied.

As per step 3: Finding the values of $x_i = Y'_i - X'_i$ in the table 2.2.5

Job	Machine X	Machine Y	$\mathbf{x_i}$
i.	$\mathbf{X'_i}$	$\mathbf{Y'_i}$	$\mathbf{Y'_i - X'_i}$
1.	5.2	6.4	1.2
2.	4.4	7.2	2.8
3.	4.4	8.8	4.4
4.	6.3	7.4	1.1
5.	3.3	6.8	3.5

TABLE 2.2.5

As per step 4- Assemble the jobs in ascending order of x_i.The sequence thus found is 4, 1, 2, 5, 3.

As per step 5- Min$\{X'_i\}$ =3.3$\neq X'_4$. So moving on to the 6th step.

As per step 6- Considering the possible sequences:

S_1: 4, 1, 2, 5, 3

S_2: 1, 4, 2, 5, 3

S_3: 2, 4, 1, 5, 3

S_4: 5, 4, 1, 2, 3

S_5: 3, 4, 1, 2, 5

As per step 7- Calculate the total of waiting time of all the jobs for the sequences S_1, S_2, S_3, S_4, S_5 as given by equations (2.1.5)

For this problem $\sum_{i=1}^{5} X_i' = 23.6$

For the sequence S_1: 4, 1, 2, 5, 3

W =25

For the sequence S_2: 1, 4, 2, 5, 3

W = 19.6

For the sequence S_3: 2, 4, 1, 5, 3

W = 18.8

For the sequence S_4: 5, 4, 1, 2, 3

W= 15.4

For the sequence S_5: 3, 4, 1, 2, 5

W=24.5

Hence sequence S_4: 5, 4, 1, 2, 3 is the requisite sequence with waiting time 15.4.

Chapter-3

J0B BLOCK CONCEPT IN TWO STAGE SPECIALLY STRUCTURED FSS MODEL WITH TRANSPORTATION TIME TO OPTIMIZE TOTAL WAITING TIME OF JOBS

In the present chapter we study the flow shop scheduling models in two stage in which the times of processing the jobs are not chosen arbitrarily but they satisfy a definite structural relationship. The time of waiting by the jobs for turn on first machine is considered to be zero. The jobs have to wait for operation on the 2nd machine as the previous job which is in operation can take some time. The concept of performing two jobs as a group job(Job- Block) has been observed wisely.

Chikhi, N.[41], developed his theory in two stage FSS with the consideration of times of transportation of jobs. Maggu, P.L., et.al.[16] has also given the significance to transportation time of jobs. Maggu, P.L., et. al. [17] also initiated the notion of group job or job block in two stage FSS model while considering the time to transport the jobs from 1st machine to 2nd machine to minimize the total elapsed time.

SECTION 3.1
OPTIMIZATION OF TOTAL WAITING TIME OF JOBS WITH CONCEPT OF JOB BLOCK AND TRANSPORTATION TIME IN 2-MACHINE n- JOB FSS PROBLEM

3.1.1 Problem Formulation

Assuming the two machines P and Q are processing n jobs in the sort PQ. P_i and Q_i are the times to process i^{th} job, $1 \leq i \leq n$, on machine P and Q respectively. The times of transporting i^{th} job from machine P to machine Q are denoted by $t_{i,1\to 2}$. The fictitious machines X, Y with equivalent times X'_i and Y'_i to process the jobs as defined by Maggu, P.L., et.al.[16], [19], Gupta,D. et.al.[43] are given as:

$$X'_i = P_i + t_{i,1\to 2} \quad (3.1.1)$$

$$Y'_i = Q_i + t_{i,1\to 2}$$

Satisfying the processing time structural relationship

Max $X_i' \leq$ Min Y_i' (3.1.2)

(l, m) is the group job or job block which is equivalent to the single job β.

Job	Machine P	Transportation Time	Machine Q
i.	$\mathbf{P_i}$	$t_{i,1\to2}$	$\mathbf{Q_i}$
1.	P_1	$t_{1,1\to2}$	Q_1
2.	P_2	$t_{2,1\to2}$	Q_2
3.	P_3	$t_{3,1\to2}$	Q_3
….	…..	…..	…..
n.	P_n	$t_{n,1\to2}$	Q_n

TABLE 3.1.1 Processing Time Model Formulation

Then our intention is to attain sequence of jobs including job block concept in which the total of the times spent in waiting by all the jobs is minimum.

3.1.2 Assumptions

1) P and Q machines are processing n jobs, firstly on machine P then on machine Q and no passing is permissible.
2) The machines cannot process the same job together in the same span of time.
3) The processing of the machine can't be interrupted until a job which is in implementation can't be completed.
4) Break down interval and set up times of machines are negligible.
5) It is given two jobs l, m as a block or group job in the order (l, m) showing priority of job l over m.

3.1.3 Lemma. Assuming the two machines P and Q are processing n jobs in the sort PQ. P_i and Q_i are the times to process i^{th} job, $1 \leq i \leq n$, on machine P and Q correspondingly with their respective set up times s_i and t_i. Equivalent times to process i^{th} job on fictitious machines X, Y are defined as in equations (3.1.1) and satisfying structural relationship defined in equations (3.1.2) then for the n job sequence S: $\sigma_1, \sigma_2, \sigma_3, \dots \dots \dots \sigma_n$

$T_{\sigma_n Y}$, the time of completion of n^{th} job on machine Y is given by

$T_{\sigma_n Y} = X'_{\sigma_1} + Y'_{\sigma_1} + Y'_{\sigma_2} \dots + Y'_{\sigma_n}$ (3.1.3)

The **proof** of the Lemma is presented in Chapter 2, Section 2.1, Lemma 2.1.3

3.1.4 Lemma For n- job sequence S: $\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n$, W_{σ_i}(say) is the time spent in waiting by i[th] job on machine Y

$$\left.\begin{aligned} & W_{\sigma_1} = 0 \\ & W_{\sigma_k} = X'_{\sigma_1} + \textstyle\sum_{r=1}^{k-1} x_{\sigma_r} - X'_{\sigma_k} \\ & \text{Where } x_{\sigma_r} = Y'_{\sigma_r} - X'_{\sigma_r}, \quad \sigma_r \in (1, 2, 3, \dots, n) \end{aligned}\right\} \quad (3.1.4)$$

And the notations have usual meaning as defined in Lemma 3.1.3

The **proof** of the Lemma is presented in Chapter 2, Section 2.1, Lemma 2.1.4

3.1.5 Theorem For n- job sequence S: $\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n$.The total of the times spent by jobs in waiting W (say) is given by

$$W = nX'_{\sigma_1} + \sum_{r=1}^{n-1}(n-r)x_{\sigma_r} - \sum_{i=1}^{n} X'_i \quad (3.1.5)$$

Where $x_{\sigma_r} = Y'_{\sigma_r} - X'_{\sigma_r}$, $\sigma_r \in (1, 2, 3, \dots, n)$ and the notations have usual meaning as defined in Lemma 3.1.3

The **proof** of the theorem is presented in Chapter 2, Section 2.1, Theorem 2.1.5

3.1.6 Theorem: Equivalent Job Block Theorem

Assuming the two machines X, Y are processing n jobs in the sort XY. X'_i and Y'_i are the times to process i[th] job, $1 \le i \le n$, on machine X and Y respectively. (l, m) is the group job or job block which is the same as the single job β (called equivalent job β). Job β has processing times X'_β and Y'_β on the machines X and Y and are given by

$$\left.\begin{aligned} & X'_\beta = X'_l + X'_m - \min(X'_m, Y'_l) \\ & Y'_\beta = Y'_l + Y'_m - \min(X'_m, Y'_l) \end{aligned}\right\} \quad (3.1.6)$$

The theorem is presented with proof by Maggu, P.L., et.al.[20].

3.1.7 Algorithm For The Formulated Problem

Step 1: Find the equivalent times X'_i and Y'_i of processing the jobs on fictitious machines X and Y respectively using the equations (3.1.1).

Step2: Validate the processing time structural relationship as defined in equation (3.1.2).

Step 3: Consider equivalent job β= (l, m) and calculate the times of processing using equations (3.1.6) and write the single job β in spite of the pair of jobs (l, m).

Step 4: Find $x_i = Y'_i - X'_i$ and write the values in the table 3.1.2.

Job	Machine X	Machine Y	x_i
i.	X'_i	Y'_i	$Y'_i - X'_i$
1.	X'_1	Y'_1	x_1
2.	X'_2	Y'_2	x_2
3.	X'_3	Y'_3	x_3
….	…..	…..	…..
β .	X'_β	Y'_β	x_β
…..	…..	…..	…..
n-1.	X'_{n-1}	Y'_{n-1}	x_{n-1}

TABLE 3.1.2

Step 5: Assemble the jobs in ascending order of x_i. Let the sequence thus found be $(\sigma_1, \sigma_2, \sigma_3, \dots \dots \dots \sigma_{n-1})$

Step 6: Consider the sequences $S_1, S_{2,} S_3, \dots \dots, S_{n-1}$. S_1 is the sequence obtained in 5th step, Sequences $S_i, 2 \le i \le n-1$ can be attained by taking ith job in the sequence S_1 to the 1st position and taking same respite of the sequence.

Step 7: Calculate the total waiting time W for all the sequences $S_1, S_{2,} S_3, \dots \dots, S_{n-1}$ using the equations (3.1.5).

The sequence with minimum W is the required sequence for scheduling.

3.1.8 Numerical Illustration

Consider the processing of 5 jobs in Flow Shop on two machines P and Q. P_i and Q_i are the times to process ith job, $1 \le i \le 5$, on machine P and Q respectively.

Time to transport i^{th} job from machine P to machine Q is denoted by $t_{i,1\rightarrow 2}$. Assume the values of the processing time matrix in table 3.1.3.

Job	Machine P	Transportation time	Machine Y
i.	P_i	$t_{i,1\rightarrow 2}$	Q_i
1.	3	2	6
2.	1	3	8
3.	6	2	7
4.	2	1	11
5.	2	2	12

TABLE 3.1.3

Our intention is to attain sequence of jobs lessening the total of the times exhausted in waiting by all the jobs by taking jobs 3, 5 as a job block or group job (3, 5).

As per step 1- Finding the equivalent imes X_i' and Y_i' of processing the jobs on fictitious machines X, Y respectively as defined in equations (3.1.1) in table 3.1.4

Job	Machine X	Machine Y
i.	X'_i	Y'_i
1.	5	8
2.	4	11
3.	8	9
4.	3	12
5.	4	14

TABLE 3.1.4

As per step 2: The processing time structural relationship Max X_i'=8≤ Min $Y_i' = 8$ as defined in equation (3.1.2) is satisfied.

As per step 3-Taking (3, 5) as a job block and`denoting this job block by β. The times of processing of single job β on machines X and Y as defined in 3.1.6 equation are evaluated as follows:

$X'_\beta = X'_3 + X'_5 - \min\ (X'_5, Y'_3)$ =8; $Y'_\beta = Y'_3 + Y'_5 - \min\ (X'_5, Y'_3)$ =19

As per step 4: Finding the values of $x_i = Y_i' - X_i'$ in the table 3.1.5.

Job	**Machine X**	**Machine Y**	$\mathbf{x_i}$
i.	$\mathbf{X_i'}$	$\mathbf{Y_i'}$	$\mathbf{Y_i' - X_i'}$
1.	5	8	3
2.	4	11	7
β.	8	19	11
4.	3	12	9

TABLE 3.1.5

As per step 5- Assemble the jobs in ascending order of x_i.The sequence thus found is 1, 2, 4, β.

As per step 6- Considering the possible sequences:

S_1: 1, 2, 4, β.

S_2: 2, 1, 4, β

S_3: 4, 1, 2, β

S_4: β, 1, 2, 4

As per step 7- Calculate the total of waiting time of all the jobs for the sequences $S_1, S_2, S_3, S_4,$ as derived in 3.1.5 Theorem and is given by equations (3.1.5)

For this problem $\sum_{i=1}^{5} X_i' = 24$

For the sequence S_1:1, 2, 4, β or S_1:1, 2, 4, 3, 5

W=53

For the sequence S_2: 2, 1, 4, β or S_2:2, 1, 4, 3, 5

W=52

For the sequence S_3: 4, 1, 2, β or S_3: 4, 1, 2, 3, 5

W=51

For the sequence S_4: β, 1, 2, 4 or S_4: 3, 5, 1, 2, 4

W=63

Hence sequence S_3: 4, 1, 2, 3, 5 is the requisite sequence for scheduling with waiting time 51 alongwith consideration of (3, 5) as a group job.

Chapter - 4

RESULTS AND DISCUSSIONS

The report covers up the basic concepts of OR and scheduling theory. The report focuses on the FSS Models in two stage in which the times to process the jobs are not at the whole arbitrary but have a definite relation with one another. It is assumed the minimum of the times to process the jobs on second machine can never be less than the maximum of the times to process the jobs on the first machine. The main intention of the study is to lessen the total of the times spent in waiting by all the jobs.

Chapter 1 is introductory chapter highlighting the definition of Operations Research, and Scheduling theory. It highlights the various concepts used in scheduling theory, its solution methodology.Literature review of Scheduling theory is also presented in the chapter.

Chapter 2, divided into two sections, signifies the time of transporting jobs from first machine to second machine in FSS models in two stage. Section 2.1 deals in two stage specially structured flow shop for finding the sequence lessening the total of the waiting time of all the jobs. In section 2.2 the time to transport the job from 1^{st} machine to 2^{nd} machine and probabilities with the processing times are taken into account. The heuristic algorithms and the solved numerical example in both of the sections are presented.

Chapter 3, highlights concept of job block in 2 machine n job FSS specially structured model along with the consideration of transportation time of job from first machine to second machine. The heuristic algorithm with example is presented minimizing the total waiting time of jobs.

Chapter - 5

CONCLUSIONS AND FURTHER SCOPE

The proposed study deals with the two stage flow shop scheduling with the key thought to lessen the total of the waiting time of all the jobs. The parameters like transportation time of jobs and concept of job block have been applied and their heuristic algorithms are presented. Chapter wise report throws light on the detail examination of the factor waiting time of jobs by proposing the heuristic algorithms.

The parameter transportation time of jobs becomes significant in the cases when the machines placed at far places from one another. Therefore it's the factor which cannot be under estimated when there is intention to optimize total waiting time of jobs.

Though the intention of the study may add to the additional costs like penalty cost of the jobs or machine idle cost, so far the proposal of minimizing the times spent in waiting by all the jobs is a matter that cannot be avoided in the cases when there is a minimum time contract with the customers.

The study can be widened by making an allowance for a choice of parameters like weightage of jobs, break down interval of machines etc.

The study can be further extensive to three or more machines. The various parameters in the scheduling theory can also be applied to optimize the total of the times spent in waiting by all the jobs.

The study can also be extended by considering the models in dynamic environment and also applying the parallel servers in one or both of the machines to reduce more waiting time.

BIBLIOGRAPHY

[1]**Johnson**, Optimal two and three stage production schedule with set up times included, *Naval Res Log Quart* Vol. 1, pp. 61-68, (1954).

[2] **Bellman, R.,** Mathematical Aspects of Scheduling Theory, *RAND Report P-651, Rand Corporation, Santa Monica*, April 11, 1955.

[3] **Jackson, J.R.,** An extension of Johnson's results on job scheduling, *Naval Res Log. Quart.,* Vol. 3, pp. 201-203, (1956).

[4] **Smith, W.E**., Various Optimizers for single stage production, *Naval Res. Log.*, Vol. 3, No.1-2, pp.59-66, (1956).

[5] **Mitten L.G.**, Sequencing n jobs on two machines with arbitrary time lags. *Management Science,* Vol. 5, No.3, pp. 293-298(1959).

[6] **Bagga, P.C.**, Sequencing in a rental situations, *Journal of Canadian Operation Research Society*, Vol. 7, pp.152-153, (1969).

[7] **Bagga, P.C.**, n-jobs, 2-machine sequencing problem with stochastic service times, *Operations Research*, Vol. 7, pp.184-199, (1970).

[8] **Campell, H.A., Duder, R.A. and Smith, M.L.,** A heuristic algorithm for n-jobs, m-machines sequencing problem, *Management Science*, Vol. 16, pp.B630-B637, (1970).

[9] **Baker, K.R.**, Introduction to Sequencing and Scheduling, *Wiley, New York*, (1974).

[10] **Gupta J.N.D.**, Optimal Schedules for Special Structure Flow- Shops, *Naval Research Logistics Quarterly*, Vol.2, No.2, pp. 255-269, (1975).

[11] **Maggu, P.L., Das G.**, Equivalent jobs for job-blocks in job sequencing, *Operations Research*, Vol.14, No. 4, pp. 271-281, (1977).

[12] **Panwalker S.S., Iskander W.**, A survey of scheduling rules, *Operation Research*, Vol. 25, pp. 45-61, (1977).

[13] **Bhatnagar V., Das, G., and Mehta, O.P.**, n-job two machine flow-job shop scheduling problem having minimum total waiting time for all jobs., *PAMS*, Vol. X, No.1-2, (1979).

[14] **Garey, M.R., Johnson, D.S.**, Computers and Intractability: A Guide to the theory of NP-Completeness. *W. H. Freeman and Company: New York*, (1979).

[15] **Gilett, B.E.**, Introduction to Operations Research, *Tata McGraw Hill: New York*, (1979).

[16] **Maggu P.L., Das G.**, On 2 x n sequencing problem with transportation times of jobs, *Pure and Applied Mathematika Sciences,* Vol. 12, pp.1-6, (1980).

[17] **Maggu P.L., Das G., and Kumar R.**, On equivalent-job for job-block in 2xn sequencing problem with transportation times, *Journal of the OR Society of Japan*, Vol. 24, No.2, pp. 136-146, (1981).

[18] **Nawaz, M., Enscore Jr, E. E., and Ham, L.**, A heuristic algorithm for the m-machine, n-job flow shop sequencing problem, *OMEGA, Elsevier,* Vol. 11, No.1, pp. 91-95, (1983).

[19] **Maggu P.L., Yadav S.K., Singh T.P. and Dev, Arjun**, Flow shop scheduling problems involving job-weight and transportation time, *PAMS*, Vol. 20, No.1-2, pp.153-158, (1984).

[20] **Singh, T.P.**, On n x 2 flow shop problem involving job block, transportation times, and break-down machine time, *PAMS*, Vol. XXI, No. 1-2, (1985).

[21] **Gupta, J.N.D. and Darrow, W.P.**, The two machine sequence dependent flow shop scheduling problem, *European Journal of Operational Research*, Vol. 24, No.3, pp. 439-446, (1986).

[22] **Panwalkar S.S.**, Scheduling of a two machine flow-shop with travel time between machines, *Journal of the Operational Research Society*, Vol.42, No.7, pp. 609-613, (1991).

[23] **Rajendran, C., Chaudhuri, D.**, An efficient heuristic approach to the scheduling of jobs in a flow-shop, *European Journal of Operational Research*, Vol.61, No. 3, pp.318- 325, (1992).

[24] **Rajendran, C.,** Heuristic algorithm for scheduling in flow shop to minimize total flow time, *International Journal of Production Economics*, Vol.29, pp.65–73, (1993).

[25] **Blazewicz, J., Ecker, K.H., Schmidt, G., and Weglarz J.**, Scheduling in Computer and Manufacturing System, *Springer*, (1994).

[26] **Woeginger G.J.**, Heuristics for parallel machine scheduling with delivery times, *Acta Informatica,* Vol. 31, pp. 503-512, (1994).

[27] **Parker, R.G.**, Deterministic Scheduling Theory, *Chapman and Hall: New York,* (1995).

[28] **Stevens J.W., Gemmill D.D.**, Scheduling a two-machine flow-shop with travel times to minimize maximum lateness, *International Journal of Production Research,* Vol. 35, pp.1-15, (1997).

[29] **Narain, L. and Bagga, P.C.**, Minimizing hiring cost of machines in n x 3 flow shop problems, *XXXI annual ORSI Convention and International Conference on Operation Research and Industry, Agra (India),* (1998).

[30] **Lee, C.Y. and Chen, Z.L.**, Machine scheduling with transportation considerations, *Journal of scheduling*, Vol. 4, pp. 3-24, (2001).

[31] **Oguz C., Zinder, Y., Do, V., Janiak, A., Lichtenstein, M.,** Hybrid flow-shop scheduling Problems with multiprocessor task systems, *European Journal of Operational Research*, Vol. 152, No.1, pp. 115- 131,(2004).

[32] **Singh, T. P., Gupta, D.**, Minimizing Rental Cost in Two Stage Flow Shop, the Processing Time Associated with Probabilities Including Job Block, *Reflections de ERA*, Vol.1, No.2, pp.107-120(2005).

[33] **Hillier, F.S., and Liberman, G.J.,** Introduction to Operations Research, *Tata McGraw Hill, Eighth Edition*, (2005).

[34] **Lee, Y.H. and Jung, J.W.**, New heuristics for no-wait flow shop scheduling with precedence constraints and sequence dependent setup time, *Proceedings Computer Science and its Applications, ICCSA 2005, Springer,* Vol. 3483, pp. 467-476, (2005).

[36] **Gupta, D., Singh, T.P. and Kumar, R.**, Minimization of rental cost in nx2 production scheduling, Set up time separated including job block criteria, *Journal of Mathematical Science, Reflection des ERA*, Vol.1, pp. 245-256, (2006).

[37] **Singh T.P. and Indira V.**, Minimize Rental Cost of machines in m stage scheduling with job-block concept, *Reflections Des Era, J. of Mathematical Sciences,* Vol. 2, No. 4, pp 359-370(2007).

[38] **Brucker, P.**, Scheduling Algorithm, *Springer, Berlin, Heidelberg*, (2007).

[39] **Wang X. and Cheng T.C.E.,** Heuristic for two machine flow shop scheduling with setup times and an availability constraint, *Computer and Operation Research*, Vol. 34, No.1, pp. 152-162, (2007).

[40] **Belwal, O.K. and Mittal, A.,** n job 2 machine flow shop scheduling problem with break-down of machines, transportation time and equivalent job block, *Bulletin of Pure & Applied Sciences-Mathematics*, Vol. 27, No. 1, (2008).

[41] **Chikhi, N.**, Two machine flow-shop with transportation time, *Thesis, Faculty of Mathematics*, USTHB University, Algiers, (2008).

[42] **Gupta D., Sharma S., Bala S.**, Optimal two stage specially structured flow shop scheduling: Mminimize the rental cost with independent setup time,

International Journal of Applied Operational Research, Vol.2, No.3, pp.31-43, (2012).

[43] **Gupta, D., Sharma, S., Bala, S.** , n x 2 specially structured flow shop scheduling with transportation time to minimize the rental cost of machines*, International Journal of Mathematical Archive,* Vol.3 No.2, (2012).

[44] **Gupta, D., Bala S., Singla, P.,** Specially structured two stage flow shop scheduling to minimize the rental cost, processing time each associated with probabilities including weightage of jobs*, International Journal of Research in Management*, Vol.2, No.2, (2012).

[45] **Gupta, D., Bala S., Singla, P.,** Minimizing rental cost for specially structured two stage flow shop scheduling, processing time, setup time associated with probabilities including weightage of jobs, *Advances in Applied Science Research,* Vol. 3, No.5, pp. 2906-2911, (2012).

[46] **Pinedo, M.L.,** Scheduling: Theory, Algorithms and Systems, *Springer, New York*.

APPENDICES

C++ Program for 2.1.7 Algorithm

```
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
void main( )
{
clrscr( );
float t[50], P[50], Q[50], x_k[50], y_k[50], xk[50], tempx, tempy, tempsort, kcopy[50], temp,
job1[50], job2[50], job3[50], job4[50], minx, tem, temx;
int i, j, nbr, job[50], jobx, minjobx;
cout<<" Input the number of jobs\n";
cin>>nbr;
cout<<" Input the processing times of Machine P\n";
for(i=0;i<nbr;i++)
{
cin>>P[i];
}
cout<<" Input the processing times of Machine Q\n";
for(i=0;i<nbr;i++)
{
cin>>Q[i];
}
cout<<"\n Input the transportation times from Machine P to Machine Q\n";
for(i=0;i<nbr;i++)
{
cin>>t[i];
}
cout<<"Fictitious Machine X\n";
cout<<"Job\tX'i\t\t\n";
float sumx_k=0;
```

```
for(i=0;i<nbr;i++)
{
jobx=i+1;
x_k[i]=P[i]+t[i];
sumx_k=sumx_k+x_k[i];
cout<<jobx<<"\t"<<x_k[i]<<"\n";
}
getch();
cout<<"Fictitious Machine Y\n";
cout<<"Job\tY'i\t\t\n";
for(i=0;i<nbr;i++)
{
y_k[i]=Q[i]+t[i];
cout<<i+1<<"\t"<<y_k[i]<<"\n";
}
getch();
tempx=x_k[0];
for(i = 0;i < nbr; ++i)
{
if(tempx<x_k[i])
tempx=x_k[i];
}
cout <<"Maximum X'i = "<<tempx<<"\n";
tempy=y_k[0];
for(i = 0;i < nbr; ++i)
{
if(tempy> y_k[i])
tempy=y_k[i];
}
cout << "Minimum Y'i= " <<tempy<<"\n";
if(tempx<=tempy)
```

```
{
cout<<"Processing time structural relationship is satisfied\n";
cout<<"Job\txi=Y'i-X'i\n";
for(i=0;i<nbr;i++)
{
xk[i]=y_k[i]-x_k[i];
cout<<i+1<<"\t"<<xk[i]<<"\n";
kcopy[i]=xk[i];
job1[i]=i+1;
}
getch();
for(i=0;i<nbr;i++)
{
for(j=i+1;j<nbr;j++)
{
if(kcopy[i]>kcopy[j] || job1[i]>job1[j] )
{
tempsort=kcopy[i];
kcopy[i]=kcopy[j];
kcopy[j]=tempsort;
temp=job1[i];
job1[i]=job1[j];
job1[j]=temp;
}
}
}
float sum=0.0;
int index;
int j1=0;
int j=nbr-1;
index=job1[0];
```

```
sum=sum+(x_k[index-1]*nbr);
cout<<"Sequence S1 is:";
for(i=0;i<nbr;i++,j--)
{
cout<<job1[i]<<"\t";
j1=job1[i];
sum=sum +(j*xk[j1-1]);
job2[i]=job1[i];
}
float sumfinal[30];
sumfinal[0]=sum-sumx_k;
cout<<"\nWaiting Time W="<<sumfinal[0];
minx=x_k[0];
minjobx=1;
for(i=0;i<nbr;++i)
{
if(minx>x_k[i])
{
minx=x_k[i];
minjobx=i+1;
}
}
cout <<"\n Minimum X'i="<<minx;
if(minjobx==job1[0])
{
cout<<"\n Sequence S1 is the required Sequence";
}
else
{
cout<<"\n Other Possible sequences are:\n";
int aa=nbr,pp=1,m=0,j;
```

```
while(aa>1)
{
sum=0;
int templl=job2[0];
job2[0] = job2[pp];
job2[pp]=templl;
index=job2[0];
cout<<"S"<<pp+1;
sum=sum+(x_k[index-1]*nbr);
j=nbr-1;
for(i=0;i<nbr;i++,j--)
{
cout<<"\t"<<job2[i];
j1=job2[i];
sum=sum +(j*xk[j1-1]);
}
sumfinal[m+1]=sum-sumx_k;
cout<<"\n Waiting Time W = "<<setprecision(2)<<sumfinal[m+1]<<"\n";
pp++;
aa--;
m++;
getch();
}
float minsum;
minsum=sumfinal[0];
for(i=1;i<nbr;++i)
{
if(minsum>sumfinal[i])
{
minsum=sumfinal[i];
}
```

```
}
cout <<"\n Minimum W = "<<minsum;
cout<<"\n  Sequence With Minimum Waiting Time W is the Required Sequence";
}
}
else
{
cout<<"Processing time structural relationship is not satisfied";
}
getch( );
}
```

C++ Program for 2.2.3 Algorithm

```
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
void main()
{
clrscr();
float  p[50], q[50], P[50], Q[50], mul_P[50], mul_Q[50], tot_p, tot_q, t[50], x_k[50], y_k[50],
xk[50], tempx, tempy, tempsort,  kcopy[50], temp, job1[50], job2[50], job3[50],  job4[50], minx,
tem, temx;
int i, j, nbr, job[30], jobx, minjobx;
cout<<"Input the number of jobs\n";
cin>>nbr;
cout<<" Input the processing times of Machine P\n";
for(i=0;i<nbr;i++)
{
cin>>P[i];
}
abc:
tot_p=0.0;
cout<<"\n Input the probabilities for processing times of machine P\n";
for(i=0;i<nbr;i++)
{
cin>>p[i];
tot_p=tot_p+p[i];
}
if(tot_p>1.0001)
{
cout<<"sum of probabilities is greater than 1\n enter the values again";
goto abc;
```

```
}
if(tot_p<.9999)
{
cout<<"sum of probabilities is less than 1\n enter the values again";
goto abc;
}
cout<<"The total sum of probabilities\t"<<tot_p<<"\n";
cout<<" Input the processing times of Machine Q\n";
for(i=0;i<nbr;i++)
{
cin>>Q[i];
}
abcd:
tot_q=0.0;
cout<<"\n Input the probabilities for processing times of Machine Q\n";
for(i=0;i<nbr;i++)
{
cin>>q[i];
tot_q=tot_q+q[i];
}
if(tot_q>1.0001)
{
cout<<"sum of probabilities is greater than 1\n enter the values again";
goto abcd;
}
if(tot_q<0.9999)
{
cout<<"sum of probabilties is less than 1\n enter the values again";
goto abcd;
}
cout<<"The total sum of probabilities\t"<<tot_q<<"\n";
```

```
cout<<"\n Input the transportation times from machine P to machine Q\n";
for(i=0;i<nbr;i++)
{
cin>>t[i];
}
cout<<" Fictitious Machine X\n";
cout<<"Job\t X'i\n";
float sumx_k=0;
for(i=0;i<nbr;i++)
{
jobx=i+1;
x_k[i]=P[i]*p[i]+t[i];
sumx_k=sumx_k+x_k[i];
cout<<jobx<<"\t"<<x_k[i]<<"\n";
}
cout<<"Fictitious Machine Y\n";
cout<<"Job\tY'i\n";
for(i=0;i<nbr;i++)
{
y_k[i]=Q[i]*q[i]+t[i];
cout<<i+1<<" \t"<<y_k[i]<<"\n";
}
tempx=x_k[0];
for(i = 0;i < nbr; ++i)
{
if(tempx<x_k[i])
tempx=x_k[i];
}
cout <<"Maximum X'i = "<<tempx<<"\n";
tempy=y_k[0];
for(i = 0;i < nbr; ++i)
```

```
{
if(tempy> y_k[i])
tempy=y_k[i];
}
cout << "Minimum Y'i= " <<tempy<<"\n";
if(tempx<=tempy)
{
cout<<"Processing time structural relationship is satisfied\n";
cout<<"Job\txi=Y'i-X'i\n";
for(i=0;i<nbr;i++)
{
xk[i]=y_k[i]-x_k[i];
cout<<i+1<<"\t"<<xk[i]<<"\n";
kcopy[i]=xk[i];
job1[i]=i+1;
}
for(i=0;i<nbr;i++)
{
for(j=i+1;j<nbr;j++)
{
if(kcopy[i]>kcopy[j] || job1[i]>job1[j] )
{
tempsort=kcopy[i];
kcopy[i]=kcopy[j];
kcopy[j]=tempsort;
temp=job1[i];
job1[i]=job1[j];
job1[j]=temp;
}
}
}
```

```
float sum=0.0;
int index;
int j1=0;
int j=nbr-1;
index=job1[0];
sum=sum+(x_k[index-1]*nbr);
cout<<"Sequence S1 is:";
for(i=0;i<nbr;i++,j--)
{
cout<<job1[i]<<"\t";
j1=job1[i];
sum=sum +(j*xk[j1-1]);
job2[i]=job1[i];
}
float sumfinal[30];
sumfinal[0]=sum-sumx_k;
cout<<"\nWaiting Time W="<<sumfinal[0];
minx=x_k[0];
minjobx=1;
for(i=0;i<nbr;++i)
{
if(minx>x_k[i])
{
minx=x_k[i];
minjobx=i+1;
}
}
cout <<"\nMinimum X'i="<<minx;
if(minjobx==job1[0])
{
cout<<"\nSequence S1 is the required Sequence";
```

```
}
else
{
cout<<"\nOther Possible sequences are:\n";
int aa=nbr,pp=1,m=0,j;
while(aa>1)
{
sum=0;
int templl=job2[0];
job2[0] = job2[pp];
job2[pp]=templl;
index=job2[0];
cout<<"S"<<pp+1;
sum=sum+(x_k[index-1]*nbr);
j=nbr-1;
for(i=0;i<nbr;i++,j--)
{
cout<<"\t"<<job2[i];
j1=job2[i];
sum=sum +(j*xk[j1-1]);
}
sumfinal[m+1]=sum-sumx_k;
cout<<"\n Waiting Time W = "<<setprecision(2)<<sumfinal[m+1]<<"\n";
pp++;
aa--;
m++;
}
float minsum;
minsum=sumfinal[0];
for(i=1;i<nbr;++i)
{
```

```
if(minsum>sumfinal[i])
{
minsum=sumfinal[i];
}
}
cout <<"\n Minimum W = "<<minsum;
cout<<"\n Sequence With Minimum Waiting Time W is the Required Sequence";
}
}
else
{
cout<<"Processing time structural relationship is not satisfied";
}
getch();
}
```

C++ Program for 3.1.7 Algorithm

```
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
#include<stdlib.h>
void main()
{
clrscr();
float P[50], Q[50],  t[50], x_k[50], y_k[50], xk[50], tempx, tempy, tempsort, kcopy[50], temp;
int  job1[50],job3[50],job4[50];
int i, j, nbr, job[50], jobx, minjobx;
cout<<" Input the number of jobs\n";
cin>>nbr;
cout<<" Input the processing times of jobs on Machine P\n";
for(i=0;i<nbr;i++)
{
cin>>P[i];
}
cout<<" Input the processing times of jobs on Machine Q \n";
for(i=0;i<nbr;i++)
{
cin>>Q[i];
}
cout<<"\n Input the transportation times of jobs\n";
for(i=0;i<nbr;i++)
{
cin>>t[i];
}
cout<<"Fictitious Machine X\n";
cout<<"Job\t Pi \t\t ti \t \t X'i \n";
```

```
float sumx_k=0.0;
for(i=0;i<nbr;i++)
{
jobx=i+1;
x_k[i]=P[i]+t[i];
sumx_k=sumx_k+x_k[i];
cout<<jobx<<"\t"<<P[i]<<"\t+\t"<<t[i]<<"\t=\t"<<x_k[i]<<"\n";
}
getch();
cout<<"Fictitious Machine Y\n";
cout<<"Job\t  Qi \t\t ti \t \t Y'i \n";
for(i=0;i<nbr;i++)
{
y_k[i]=Q[i]+t[i];
cout<<i+1<<"\t"<<Q[i]<<"\t+\t"<<t[i]<<"\t=\t"<<y_k[i]<<"\n";
}
getch();
tempx=x_k[0];
for(i = 0;i < nbr; i++)
{
if(tempx<x_k[i])
tempx=x_k[i];
}
cout <<"Maximum X'i = "<<tempx<<"\n";
tempy=y_k[0];
for(i = 0;i < nbr; i++)
{
if(tempy > y_k[i])
tempy=y_k[i];
}
cout << "Minimum Y'i= " <<tempy<<"\n";
```

```
if(tempx<=tempy)
{
cout<<"Processing time structural relationship is satisfied\n";
int jb1,jb2;
cout<<" Input the two jobs for job block\n";
cin>>jb1>>jb2;
int jbcon;
unsigned int jb1cpy,jb2cpy;
jb1cpy=jb1;
jb2cpy=jb2;
do
{
 jb2cpy /=10;
jb1cpy *=10;
} while(jb2cpy>0);
jbcon=jb1cpy+jb2;
jb1--;
jb2--;
float xp;
xp=x_k[jb1];
float yp;
yp = y_k[jb1]+y_k[jb2]-x_k[jb2];
cout<<"Equivalent processing time of job block on machine X and Y\n";
cout<<"Xα = "<<xp<<"\n"<<"Yα = "<<yp<<"\n";
cout<<"Fictitious Machine X\n";
cout<<"Job\t\tX'i\n";
for(i=0;i<nbr;i++)
{
jobx=i+1;
if (i==jb1 || i==jb2)
continue;
```

```
cout<<jobx<<"\t=\t"<<x_k[i]<<"\n";
}
x_k[i]=xp;
cout<<"Xα"<<"\t=\t"<<x_k[i]<<"\n";
cout<<"Fictitious Machine Y\n";
cout<<"Job\t\tY'i\n";
for(i=0;i<nbr;i++)
{
if(i==jb1 || i==jb2)
continue;
cout<<i+1<<"\t=\t"<<y_k[i]<<"\n";
}
y_k[i]=yp;
cout<<"Y@"<<"\t=\t"<<y_k[i]<<"\n";
float zk[50],zcopy[50];
int jobz[50];
cout<<"Job\t\txj=Y'j-X'j\n";
for(i=0;i<nbr;i++)
{
zk[i]=y_k[i]-x_k[i];
jobz[i]=i+1;
zcopy[i]=zk[i];
if(i==jb1 || i==jb2)
continue;
cout<<jobz[i]<<"\t=\t"<<zk[i]<<"\n";
}
zk[i]=yp-xp;
jobz[i]=jbcon;
cout<<jbcon<<"\t=\t"<<zk[i]<<"\n";
zcopy[i]=zk[i];
for(i=0;i<nbr+1;i++)
```

```
{
if(i==jb1 || i==jb2)
continue;
for(j=i+1;j<nbr+1;j++)
{
if(j==jb1 || j==jb2)
continue;
if(zcopy[i]>zcopy[j])
{
tempsort=zcopy[i];
zcopy[i]=zcopy[j];
zcopy[j]=tempsort;
temp=jobz[i];
jobz[i]=jobz[j];
jobz[j]=temp;
}
}
}
cout<<"Sequence S1 ";
int jobfinal[50];
int p=0;
for(i=0;i<nbr+1;i++)
{
if(i==jb1 || i==jb2)
continue;
cout<<"\t"<<jobz[i];
jobfinal[p]=jobz[i];
p++;
}
float sum=0.0;
int index,counter=0;
```

```
if(jobfinal[0]==jbcon)
{
sum=sum+(x_k[jb1]*nbr);
counter++;
}
else
{
index=jobfinal[0];
sum=sum+(x_k[index-1]*nbr);
counter++;
}
j=nbr-1;
int j1=0;
int cc=0;
p=0;
while(counter<nbr+1)
{
 if(jobfinal[p]==jbcon)
 {
if(cc==0)
{
sum=sum+(j*zk[jb1]);
j--;
cc++;
counter++;
}
else
{
sum=sum+(j*zk[jb2]);
j--;
p++;
```

```
counter++;
}
continue;
 }
 else
 {
j1=jobfinal[p];
sum=sum+(j*zk[j1-1]);
j--;
p++;
counter++;
}
}
float sumfinal[50];
sumfinal[0]=sum-sumx_k;
cout<<"\nWaiting Time W = "<<sumfinal[0];
getch();
cout<<"\nOther Possible sequences are:\n";
int pp=1,m=0;
while(pp<nbr-1)
{
int templl=jobfinal[0];
jobfinal[0] = jobfinal[pp];
jobfinal[pp]=templl;
cout<<"\nSequence S"<<++pp;
for(i=0;i<nbr-1;i++)
{
cout<<"\t"<<jobfinal[i];
}
getch();
sum=0.0;
```

```
index,counter=0;
if(jobfinal[0]==jbcon)
{
sum=sum+(x_k[jb1]*nbr);
counter++;
}
else
{
index=jobfinal[0];
sum=sum+(x_k[index-1]*nbr);
counter++;
}
j=nbr-1;
int j1=0;
int cc=0;
p=0;
while(counter<nbr+1)
{
if(jobfinal[p]==jbcon)
{
if(cc==0)
{
sum=sum+(j*zk[jb1]);
j--;
cc++;
counter++;
}
else
{
sum=sum+(j*zk[jb2]);
j--;
```

```
p++;
counter++;
}
continue;
}
else
{
j1=jobfinal[p];
sum=sum+(j*zk[j1-1]);
j--;
p++;
counter++;
}
}
sumfinal[m+1]=sum-sumx_k;
cout<<"\nWaiting Time W = "<<sumfinal[m+1]<<"\n";
m++;
}
float minsum=0.0;
minsum=sumfinal[0];
for(i=1;i<nbr-1;i++)
{
if(minsum>sumfinal[i])
{
minsum=sumfinal[i];
}
}
cout <<"\n Minimum W = "<<minsum;
cout<<"\n Sequence With Minimum Waiting Time W is the Required Sequence for scheduling";
}
else
```

```
{
cout<<"Processing time structural relationship is not satisfied";
}
getch();
}
```