International Journal of Applied Operational Research
Vol. 3, No. 1, pp. 1-18, Winter 2013

Journal homepage: www.ijorlu.ir

A Mathematical Analysis on Linkage of a Network of Queues with Two Machines in a Flow Shop including Transportation Time

D. Gupta, S. Sharma*, S. Sharma

Received: 30 April 2012 ; **Accepted:** 22 August 2012

Abstract This paper represents linkage network of queues consisting of biserial and parallel servers linked to a common server in series with a flowshop scheduling system consisting of two machines. The significant transportation time of the jobs from one machine to another is also considered. Further, the completion time of jobs/customers (waiting time + service time) in the queue network is the setup time for the first machine. The objective of the paper is of two folds, on one hand it minimizes the total waiting time and service time of jobs/customers in the queue network, and on the other hand it minimizes the idle time of the machines for the optimal sequence of jobs/customers in a given queue flowshop linkage model. A computer programme followed by a numerical illustration is given to justify the proposed algorithm.

Keywords Flowshop, Biserial, Waiting Time, Service Time, Makespan, Idle Time, Transportation Time.

1 Introduction

Waiting lines or queues are a common occurrence both in everyday life and in a variety of business and industrial situations. Forming a queue being a social phenomenon is beneficial to the society if it can be managed so that both the units that wait and the one that serves get the most benefits. The unit providing service is known as the server. Some examples are: communication systems, voice or data traffic queuing up for the lines for transmission, manufacturing systems with several work stations, units completing work in one station waiting for access to the next, Vehicles requiring service waiting for their turn in a garage, Patients arriving at a doctor's clinic for treatment, etc. Scheduling models concerned with the determination of an optimal sequence in which the goal is to service customers, or to perform a set of jobs, in order to minimize total elapsed time or another suitable measure of performance.

One of the earliest results in flow shop scheduling theory is an algorithm by Johnson's [1] for scheduling jobs in a two or three machine flow shop to minimize the time at which all jobs are completed. Jacksons [2] studied queuing systems with phase type service. Little's [3] derived the formula for calculating the mean queue length. Maggu and Das [4] discussed the

* **Corresponding Author.** (✉)
E-mail: seemasharma7788@yahoo.com (S. Sharma)

D. Gupta
Department of Mathematics, M. M. University, Mullana, Ambala, Haryana, India.

S. Sharma, S. Sharma
Department of Mathematics, D. A. V. College, Jalandhar City, Punjab, India.

effect of independent transportation time on scheduling of jobs. Maggu [5] introduced the concept of bitendom in theory of queues. Singh, T.P. [6] discussed network of queuing and scheduling system. Singh *et al* [7] studied the transient behavior of a queuing network with parallel biserial queues. Gupta *et al* [8] studied network of queues model comprised of Biserial and parallel channels linked with a common server. Kumar *et al* [9] discussed the steady state behavior of a queue model comprised of two subsystems with biserial channels linked with a common channel. A glance into the literature reveals that only few efforts have been made to establish linkage between a network of queues and flowshop scheduling models. Maggu and kumar [10] introduced linkage between serial queuing and scheduling systems. Singh and Kumar [11] studied linkage of queues in semi-series to a flowshop scheduling system. Singh *et al* [12] discussed the linkage of scheduling system with a serial queue network. Singh and Kumar [13] established linkage of a scheduling system with a biserial queue network. This paper combines the study of network of queues for providing the phase service in series with the flowshop network of two machines in a given order for processing the jobs.

Recently Gupta *et al* [14] made an attempt to link a network of queues consisting of a system of parallel biserial servers and a system of two parallel servers linked with a common server to a flow shop scheduling model. This paper is an attempt to extend their work by introducing the concept of independent transportation time, .i.e. the moving time for a job from one machine to another machine in the processing of jobs. This situation can be visualized when the machines on which jobs are to be processed are planted at different places, and these jobs require additional times in their transplantation from one machine to another in the forms of loading time of jobs, moving time of jobs and then unloading time of jobs. The various queue characteristics have been obtained explicitly under steady state behavior of the system.

The paper is organized as follows: part two is devoted to the mathematical model in which the queuing scheduling linkage model is explained. Section three is explored to deal with mathematical analysis of the proposed linkage model. The various queue characteristics are also derived in this section. Section four deals with various assumptions made along with the theorem to find the optimal sequence of jobs processing with significant transportation time. section five is devoted to the algorithm proposed for optimizing the total flow time, average waiting time and mean service rate for the proposed linkage model.

2 The Mathematical Model

The entire model is comprised of three servers S_1, S_2, S_3 which is further linked with two machines M_1 and M_2 in series. The server S_1 consists of two biserial service servers S_{11} and S_{12}. The server S_2 contains two parallel servers S_{21} and S_{22}. Server S_3 is commonly linked in series with each of two servers S_1 and S_2 for completion of first phase service demanded either at a subsystem S_1 or S_2. The service time at S_{ij} ($i, j=1, 2$) are exponentially distributed. Let mean service rate at S_{ij} ($i, j=1, 2$) be $\mu_1, \mu_2, \mu_1', \mu_2'$ and μ_3 at S_3 respectively. Queues Q_1, Q_2, Q_3, Q_4 and Q_5 are said to be formed in front of the servers if they are busy. Customers coming at rate λ_1 after completion of service at S_{11} will go to the network of the servers $S_{11} \to S_{12}$ or $S_{11} \to S_3$ with probabilities p_{12} or p_{13} such that $p_{12} + p_{13} = 1$. Further Customers coming at rate λ_2 after completion of service at S_{12} will go to the network of the servers $S_{12} \to S_{11}$ or $S_{12} \to S_3$ with probabilities p_{21} or p_{23} such that $p_{21} + p_{23} = 1$. The completion time (waiting time + service time) of customers/jobs through queues Q_1, Q_2, Q_3, Q_4 and Q_5 form the setup time for the machine M_1. Let t_i be the transportation time of the i^{th} job from machine M_1 to M_2. After coming out from the phase I, customers/jobs proceed to machines M_1 and M_2 for processing in phase II with processing times A_{i1} and A_{i2}. The objective is to develop a heuristic algorithm to find an optimal sequence of jobs/customers with minimum make span in this queuing – scheduling linkage model.

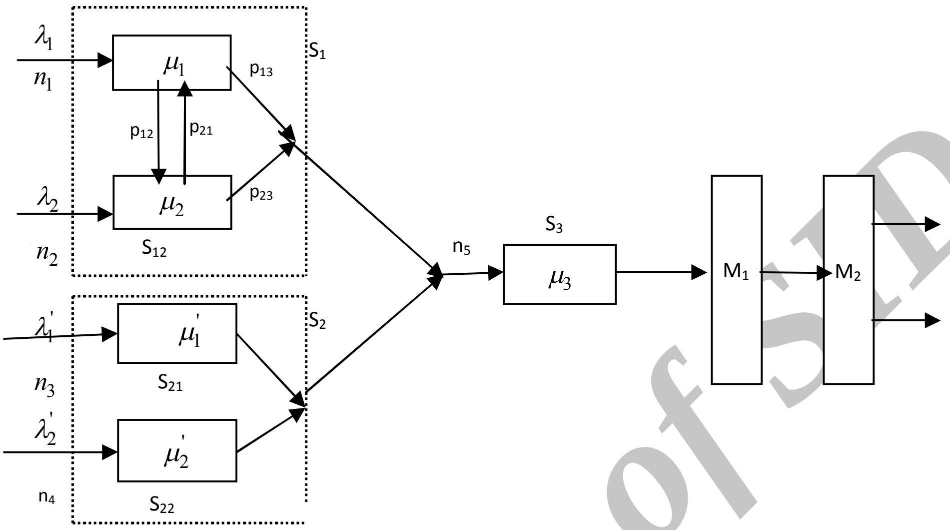

Fig. 1 Linkage Model

3 Mathematical Analysis

Let P_{n_1,n_2,n_3,n_4,n_5} be the joint probability that there are n_1 units waiting in queue Q_1 in front of S_{11}, n_2 units waiting in queue Q_2 in front of S_{12}, n_3 units waiting in queue Q_3 in front of S_{21}, n_4 units waiting in queue Q_4 in front of S_{22}, and n_5 units waiting in queue Q_5 in front of S_3 as shown in figure 1. In each case the waiting includes a unit in service, if any. Also, n_1, n_2, n_3, n_4, $n_5 > 0$.

The standard arguments lead to the following differential difference equations in transient form as

$$P_{n_1,n_2,n_3,n_4,n_5}(t) = -\left(\lambda_1+\lambda_2+\lambda_1^{'}+\lambda_2^{'}+\mu_1+\mu_2+\mu_1^{'}+\mu_3\right)P_{n_1,n_2,n_3,n_4,n_5}(t) + \lambda_1 P_{n_1-1,n_2,n_3,n_4,n_5}(t) +$$
$$\lambda_2 P_{n_1,n_2-1,n_3,n_4,n_5}(t) + \mu_1\left(n_1+1\right)p_{13}P_{n_1+1,n_2,n_3,n_4,n_5-1}(t) +$$
$$\mu_1\left(n_1+1\right)p_{12}P_{n_1+1,n_2-1,n_3,n_4,n_5}(t) + \mu_2\left(n_2+1\right)p_{23}P_{n_1,n_2+1,n_3,n_4,n_5-1}(t) +$$
$$\mu_2\left(n_2+1\right)p_{21}P_{n_1-1,n_2+1,n_3,n_4,n_5}(t) + \lambda_1^{'}P_{n_1,n_2,n_3-1,n_4,n_5}(t) + \lambda_2^{'}P_{n_1,n_2,n_3,n_4-1,n_5}(t)$$
$$+\mu_3\left(n_5+1\right)P_{n_1,n_2,n_3,n_4,n_5+1}(t) + \mu_1^{'}\left(n_3+1\right)P_{n_1,n_2,n_3+1,n_4,n_5-1}(t) + \mu_2^{'}\left(n_4+1\right)P_{n_1,n_2,n_3,n_4+1,n_5-1}(t).$$

The steady state equation $(t \to \infty)$ governing the model are depicted as

$$\left(\lambda_1+\lambda_2+\lambda_1^{'}+\lambda_2^{'}+\mu_1+\mu_2+\mu_1^{'}+\mu_2^{'}+\mu_3\right)P_{n_1,n_2,n_3,n_4,n_5} = \lambda_1 P_{n_1-1,n_2,n_3,n_4,n_5} +$$
$$\lambda_2 P_{n_1,n_2-1,n_3,n_4,n_5} + \mu_1\left(n_1+1\right)p_{13}P_{n_1+1,n_2,n_3,n_4,n_5-1} +$$

$$\mu_1(n_1+1)p_{12}P_{n_1+1,n_2-1,n_3,n_4,n_5} + \mu_2(n_2+1)p_{23}P_{n_1,n_2+1,n_3,n_4,n_5-1} +$$
$$\mu_2(n_2+1)p_{21}P_{n_1-1,n_2+1,n_3,n_4,n_5} + \lambda_1^{'}P_{n_1,n_2,n_3-1,n_4,n_5} + \lambda_2^{'}P_{n_1,n_2,n_3,n_4-1,n_5}$$
$$+\mu_3(n_5+1)P_{n_1,n_2,n_3,n_4,n_5+1} + \mu_1^{'}(n_3+1)P_{n_1,n_2,n_3+1,n_4,n_5-1} + \mu_2^{'}(n_4+1)P_{n_1,n_2,n_3,n_4+1,n_5-1}. \quad (1)$$

Let us define the generating function as

$$F(X,Y,Z,R,S) = \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty} P_{n_1,n_2,n_3,n_4,n_5} X^{n_1} y^{n_2} Z^{n_3} R^{n_4} S^{n_5}$$

where

$$|X| = |Y| = |Z| = |R| = |S| = 1.$$

Also we define partial generating functions as

$$F_{n_2,n_3,n_4,n_5}(X) = \sum_{n_1=0}^{\infty} P_{n_1,n_2,n_3,n_4,n_5} X^{n_1}$$

$$F_{n_3,n_4,n_5}(X,Y) = \sum_{n_2=0}^{\infty} P_{n_2,n_3,n_4,n_5}(X) Y^{n_2}$$

$$F_{n_4,n_5}(X,Y,Z) = \sum_{n_3=0}^{\infty} P_{n_3,n_4,n_5}(X,Y) Z^{n_3}$$

$$F_{n_5}(X,Y,Z,R) = \sum_{n_4=0}^{\infty} P_{n_4,n_5}(X,Y,Z) R^{n_4}$$

$$F(X,Y,Z,R,S) = \sum_{n_5=0}^{\infty} P_{n_5}(X,Y,Z,R) S^{n_5}$$

Now, on taking n_1, n_2, n_3, n_4, n_5 equal to zero one by one and then taking two of them pairwise, three of them at a time, four of them at a time and all of them; we get 32 equations. Now, on proceeding on the lines of Gupta *et al* [8] and following the standard technique, this after manipulation gives the final reduced result as:

$$F(X,Y,Z,R,S) = \frac{\begin{array}{c}\mu_1\left(1-\frac{S}{X}p_{13}-\frac{Y}{X}p_{12}\right)F(Y,Z,R,S) + \mu_2\left(1-\frac{S}{Y}p_{23}-\frac{X}{Y}p_{21}\right)F(X,Z,R,S) + \\ \mu_1^{'}\left(1-\frac{S}{Z}\right)F(X,Y,R,S) + \mu_2^{'}\left(1-\frac{S}{R}\right)F(X,Y,Z,S) + \mu_3\left(1-\frac{1}{S}\right)F(X,Y,Z,R)\end{array}}{\begin{array}{c}\lambda_1(1-X) + \lambda_2(1-Y) + \mu_1\left(1-\frac{Y}{X}p_{12}-\frac{S}{X}p_{13}\right) + \mu_2\left(1-\frac{S}{Y}p_{23}-\frac{X}{Y}p_{21}\right) + \lambda_1^{'}(1-Z) + \\ \lambda_2^{'}(1-R) + \mu_1^{'}\left(1-\frac{S}{Z}\right) + \mu_2^{'}\left(1-\frac{S}{R}\right) + \mu_3\left(1-\frac{1}{S}\right)\end{array}} \quad (2)$$

For convenience, let us denote

$F(Y,Z,R,S) = F_1$
$F(X,Z,R,S) = F_2$
$F(X,Y,R,S) = F_3$
$F(X,Y,Z,S) = F_4$
$F(X,Y,Z,R) = F_5$

Also *F(1,1,1,1,1)=1* , the total probability.

By taking *X =1* as $Y,Z,R,S \to 1$, *F(X, Y, Z, R, S)* is of $\frac{0}{0}$ indeterminate form.

Now, by differentiating numerator and denominator of (2) separately w.r.t X, we have

$$1 = \frac{\mu_1\left(p_{13}+p_{12}\right)F_1 + \mu_2(-p_{21})F_2}{-\lambda_1 + \mu_1\left(p_{12}+p_{13}\right) + \mu_2(-p_{21})}$$

$$\Rightarrow \mu_1 F_1 - \mu_2 p_{12} F_2 = -\lambda_1 + \mu_1 - \mu_2 p_{21} \qquad (\because p_{12}+p_{13}=1) \tag{3}$$

Similarly, by differentiating numerator and denominator of (2) separately w.r.t Y, by taking *Y=1 and* $X,Z,R,S \to 1$ we have

$$1 = \frac{\mu_1\left(-p_{12}\right)F_1 + \mu_2(p_{23}+p_{21})F_2}{-\lambda_2 + \mu_1\left(-p_{12}\right) + \mu_2(p_{23}+p_{21})}$$

$$\Rightarrow -\mu_1 p_{12} F_1 + \mu_2 F_2 = -\lambda_2 - p_{12}\mu_1 + \mu_2 \qquad (\because p_{23}+p_{21}=1) \tag{4}$$

Again, by the differentiating numerator and denominator of (2) separately w.r.t Z, by taking Z=1 *and* $X,Y,R,S \to 1$ we have

$$1 = \frac{\mu_1^{'} F_3}{-\lambda_1^{'} + \mu_1^{'}} \Rightarrow \mu_1^{'} F_3 = -\lambda_1^{'} + \mu_1^{'} \tag{5}$$

Again, by differentiating the numerator and denominator of (2) separately w.r.t R, by taking R=1 *and* $X,Y,Z,S \to 1$ we have

$$1 = \frac{\mu_2^{'} F_4}{-\lambda_2^{'} + \mu_2^{'}} \Rightarrow \mu_2^{'} F_4 = -\lambda_2^{'} + \mu_2^{'} \tag{6}$$

Again, by differentiating the numerator and denominator of **(2)** separately w.r.t S, by taking S=1 *and* $X,Y,Z,R,S \to 1$ we have

$$1 = \frac{-\mu_1 p_{13} F_1 - \mu_2 p_{23} F_2 + \mu_1^{'}(-F_3) + \mu_2^{'}(-F_4) + \mu_3(F_5)}{\mu_1(-p_{13}) + \mu_2(-p_{23}) + \mu_1^{'}(-1) + \mu_2^{'}(-1) + \mu_3}$$

$$\Rightarrow -\mu_1 P_{13} F_1 - \mu_2 P_{23} F_2 - \mu_1^{'} F_3 - \mu_2^{'} F_4 + \mu_3^{'} F_5 = -p_{13}\mu_1 - p_{23}\mu_2 - \mu_1^{'} - \mu_2^{'} + \mu_3 \tag{7}$$

By multiplying (4) with p_{21} and adding to (3), we get

$$F_1(\mu_1 - \mu_1 p_{12} p_{21}) = -\lambda_1 + \mu_1(1 - p_{12}p_{21}) - \lambda_2 p_{21}$$

$$\Rightarrow F_1 = 1 - \frac{\lambda_1 + \lambda_2 p_{21}}{(1 - p_{12} p_{21})\mu_1} \tag{8}$$

$$F_3 = 1 - \frac{\lambda_1^{'}}{\mu_1^{'}} \quad \text{(Using (5))} \tag{9}$$

$$F_4 = 1 - \frac{\lambda_2^{'}}{\mu_2^{'}} \quad \text{(Using (6))} \tag{10}$$

$$F_5 = 1 - \frac{\lambda_3^{'}}{\mu_3^{'}} \quad \text{(Using (7))} \tag{11}$$

By multiplying (3) with p_{12} and adding to (4), we get

$$\mu_2(1 - p_{12}p_{21})F_2 = -\lambda_2 - \lambda_1 p_{12} + \mu_2(p_{12} - p_{21}) + \mu_1(1 - p_{21}p_{12})$$

$$F_2 = 1 - \frac{\lambda_2 + \lambda_1 p_{12}}{(1 - p_{12}p_{21})\mu_2} \tag{12}$$

Now by putting the values of F_1, F_2, F_3, F_4 in (8), we get

$$F_5 = 1 - \left[\frac{\lambda_1^{'} + \lambda_2^{'}}{\mu_3} + \frac{(\lambda_1 + \lambda_2 p_{12})p_{13} + (\lambda_2 + \lambda_1 p_{12})p_{23}}{\mu_3(1 - p_{12}p_{21})}\right] \tag{13}$$

By using the values of F_1, F_2, F_3, F_4 and F_5, the joint probability is given by

$$P_{n_1,n_2,n_3,n_4,n_5} = \rho_1^{n_1}\rho_2^{n_2}\rho_3^{n_3}\rho_4^{n_4}\rho_5^{n_5}(1-\rho_1)(1-\rho_2)(1-\rho_3)(1-\rho_4)(1-\rho_5)$$

Where $\rho_1 = 1 - F_1$, $\rho_2 = 1 - F_2$, $\rho_3 = 1 - F_3$, $\rho_4 = 1 - F_4$, $\rho_5 = 1 - F_5$.

Further the solution in a steady state condition exist if $\rho_1, \rho_2, \rho_3, \rho_4, \rho_5 < 1$.

3.1 Mean Queue Length

Average number of the customer (L)

$$= \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty}(n_1+n_2+n_3+n_4+n_5)P_{n_1,n_2,n_3,n_4,n_5}$$

$$= \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty} n_1 P_{n_1,n_2,n_3,n_4,n_5} + \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty} n_2 P_{n_1,n_2,n_3,n_4,n_5} + ------$$

$$+ \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty} n_5 P_{n_1,n_2,n_3,n_4,n_5}$$

Therefore

$L = L_1 + L_2 + L_3 + L_4 + L_5$

Further,

$$L_1 = \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty} n_1 P_{n_1,n_2,n_3,n_4,n_5}$$

$$= \sum_{n_1=0}^{\infty}\sum_{n_2=0}^{\infty}\sum_{n_3=0}^{\infty}\sum_{n_4=0}^{\infty}\sum_{n_5=0}^{\infty} n_1 \rho_1^{n_1}\rho_2^{n_2}\rho_3^{n_3}\rho_4^{n_4}\rho_5^{n_5}(1-\rho_1)(1-\rho_2)(1-\rho_3)(1-\rho_4)(1-\rho_5)$$

$$= (1-\rho_1)(1-\rho_2)(1-\rho_3)(1-\rho_4)(1-\rho_5)\sum_{n_1=0}^{\infty} n_1\rho_1^{n_1}\sum_{n_2=0}^{\infty}\rho_2^{n_2}\sum_{n_3=0}^{\infty}\rho_3^{n_3}\sum_{n_4=0}^{\infty}\rho_4^{n_4}\sum_{n_5=0}^{\infty}\rho_5^{n_5}$$

$$= \frac{\rho_1}{1-\rho_1}.$$

Similarly

$$L_2 = \frac{\rho_2}{1-\rho_2},\ L_3 = \frac{\rho_3}{1-\rho_3}, L_4 = \frac{\rho_4}{1-\rho_4}, L_5 = \frac{\rho_5}{1-\rho_5}.$$

Therefore, mean queue length = L

$$= L_1 + L_2 + L_3 + L_4 + L_5 = = \frac{\rho_1}{1-\rho_1} + \frac{\rho_2}{1-\rho_2} + \frac{\rho_3}{1-\rho_3} + \frac{\rho_4}{1-\rho_4} + \frac{\rho_5}{1-\rho_5}$$

3.2 Average Waiting Time

The average waiting time and the average number of items waiting for a service in a service system are important measurements for a manager. Little's Law relates these two metrics via the average rate of arrivals to the system. This fundamental law has found numerous uses in operations management and managerial decision making. Little's Law says that, under steady state conditions, the average number of items in a queuing system equals the average rate at which items arrive multiplied by the average time that an item spends in the system. Let
L = Average number of items in the queuing system,
W = Average waiting time in the system for an item, and
A = Average number of items arriving per unit time.
By Little's formula, we have $L = W\lambda$; or $W = \frac{L}{\lambda}$.

4 Assumptions, Theorem and Algorithm

The following assumptions are made for developing the proposed algorithm.

1. We assume that the arrival rate in the queue network follows position distribution.
2. Each job/customer is processed on the machines M_1 and M_2 in the same order and pre-emission is not allowed, .i.e. once a job is started on a machine, the process on that machine cannot be stopped unless job is completed.
3. For the existence of the steady state behavior the following conditions hold good:

$$(i)\rho_1 = \frac{\left(\lambda_1 + \lambda_2 p_{21}\right)}{\mu_1\left(1 - p_{12}p_{21}\right)} < 1$$

$$(ii)\rho_2 = \frac{\left(\lambda_2 + \lambda_1 p_{12}\right)}{\mu_2\left(1 - p_{12}p_{21}\right)} < 1$$

$$(iii)\rho_3 = \frac{\lambda_1^{'}}{\mu_1^{'}} < 1$$

$$(iv)\rho_4 = \frac{\lambda_2^{'}}{\mu_2^{'}} < 1$$

$$(vi)\rho_5 = [\ \frac{\lambda_1^{'} + \lambda_2^{'}}{\mu_3} + \frac{p_{13}\left(\lambda_1 + \lambda_2 p_{21}\right) + p_{23}\left(\lambda_2 + \lambda_1 p_{12}\right)}{\mu_3\left(1 - p_{12}p_{21}\right)}\ ] < 1$$

4.1 Theorem

Consider flowshop consisting of n jobs and two machines A and B. All jobs are to be processed on these machines according to the order AB and each machine can handle only one job at a time and each job i has transportation time t_i from machine A to machine B and they are known prior to making scheduling decisions. An optimal ordering of jobs to minimize total elapsed time is given by the following rule: job i proceeds job i+1 if $\min\left\{A_{i,A} + t_i, A_{i+1,B} + t_{i+1}\right\} < \min\left\{A_{i+,A} + t_{i+1}, A_{i,B} + t_i\right\}$.

Proof. Let S and S' be the sequences of jobs given by

S = $J_1 - J_2 - J_3$ -…- $J_{i-1} - J_i - J_{i+1}$ -…- J_n
S' = $J_1 - J_2 - J_3$ -…- $J_{i-1} - J_{i+1} - J_i$ -…- J_n

Let $\left(A_{p,x}, A_{p,x}^{'}\right)$ and $\left(C_{p,x}, C_{p,x}^{'}\right)$ denote the processing times and completion time of p^{th} job on machine x in the processes of sequence $\left(S, S^{'}\right)$ respectively. Let $\left(t_p, t_p^{'}\right)$ denotes the transportation times of p^{th} job from machine A to the machine B in the processes of sequence $\left(S, S^{'}\right)$ respectively.

By definition, we have, $C_{p,B} = \max\left(C_{p,A} + t_p, C_{p-1,B}\right) + A_{p,B}$ (1)

Now, sequence S is preferable to S' for n jobs if

$$C_{n,B} < C_{n,B}^{'} \tag{2}$$

.i.e. $\max\left(C_{n,A} + t_n, C_{n-1,B}\right) + A_{n,B} < \max\left(C^{'}_{n,A} + t^{'}_n, C^{'}_{n-1,B}\right) + A^{'}_{n,B}$

Now, $C_{n,A} + t_n = \sum_{i=1}^{n} A_{i,A} + t_n = C^{'}_{n,A} + t_n\ C_{n,A} + t_n = \sum_{i=1}^{n} A_{i,A} + t_n = C^{'}_{n,A} + t_n$

Also, $A_{n,B} = A'_{n,B}$

Inequality (2) will hold if $C_{n-1,B} < C'_{n-1,B}$ (3)

Continuing in this way, one can easily get

$C_{p,B} < C'_{p,B}$, $(p = i+1, i+2, i+3, ------, n)$

and

$$C_{i+1,B} < C'_{i+1,B} \quad (4)$$

Now

$$C_{i+1,B} = \max\left\{C_{i+1,A} + t_{i+1}, C_{i,B}\right\} + A_{i+1,B} = \max\left\{C_{i+1,A} + t_{i+1}, \max\left\{C_{i,A} + t_i, C_{i-1,B}\right\} + A_{i,B}\right\} + A_{i+1,B}$$
$$= \max\left\{C_{i+1,A} + t_{i+1}, \max\left\{C_{i,A} + t_i + A_{i,B}, C_{i-1,B} + A_{i,B}\right\}\right\} + A_{i+1,B}$$
$$= \max\left\{C_{i+1,A} + t_{i+1} + A_{i+1,B}, C_{i,A} + t_i + A_{i,B} + A_{i+1,B}, C_{i-1,B} + A_{i,B} + A_{i+1,B}\right\}$$
$$= \max\left\{(C_{i-1,A} + A_{i,A} + A_{i+1,A} + t_{i+1} + A_{i+1,B}), (C_{i-1,A} + A_{i,A} + t_i + A_{i,B} + A_{i+1,B}), (C_{i-1,B} + A_{i,B} + A_{i+1,B})\right\} \quad (5)$$

Similarly,

$$C'_{i+1,B} = \max\left\{(C'_{i-1,A} + A'_{i,A} + A'_{i+1,A} + t'_{i+1} + A'_{i+1,B}), (C'_{i-1,A} + A'_{i,A} + t'_i + A'_{i,B} + A'_{i+1,B}),\right.$$
$$\left.(C'_{i-1,B} + A'_{i,B} + A'_{i+1,B})\right\} \quad (6)$$

Further, on comparing sequences S and S', we have

$$C_{i-1,A} = C'_{i-1,A}; C_{i-1,B} = C'_{i-1,B}; A_{i,x} = A'_{i+1,x}; t_i = t'_{i+1}; A_{i+1,x} = A'_{i,x}; t_{i+1} = t'_i \quad (7)$$

On using results (5), (6) and (7), the result (4) can be written as

$$\max\left\{(C_{i-1,A} + A_{i,A} + A_{i+1,A} + t_{i+1} + A_{i+1,B}), (C_{i-1,A} + A_{i,A} + t_i + A_{i,B} + A_{i+1,B}), (C_{i-1,B} + A_{i,B} + A_{i+1,B})\right\}$$
$$< \max\left\{(C_{i-1,A} + A_{i+1,A} + A_{i,A} + t_i + A_{i,B}), (C_{i-1,A} + A_{i+1,A} + t_{i+1} + A_{i+1,B} + A_{i,B}), (C_{i-1,B} + A_{i+1,B} + A_{i,B})\right\}$$

Or $\max\left\{C_{i-1,A} + A_{i,A} + A_{i+1,A} + t_{i+1} + A_{i+1,B}, C_{i-1,A} + A_{i,A} + t_i + A_{i,B} + A_{i+1,B}\right\}$
$< \max\left\{C_{i-1,A} + A_{i+1,A} + A_{i,A} + t_i + A_{i,B}, C_{i-1,A} + A_{i+1,A} + t_{i+1} + A_{i+1,B} + A_{i,B}\right\}$

On subtracting $C_{i-1,A} + A_{i,A} + A_{i+1,A} + t_i + t_{i+1} + A_{i,B} + A_{i+1,B}$ from each term, we have

$$\max\left\{-t_i - A_{i,B}, -t_{i+1} - A_{i+1,A}\right\} < \max\left\{-t_{i+1} - A_{i+1,B}, -t_i - A_{i,A}\right\}$$

Or $\quad \min\left\{t_i + A_{i,B}, t_{i+1} + A_{i+1,A}\right\} > \min\left\{t_{i+1} + A_{i+1,B}, t_i + A_{i,A}\right\}$

Or $\quad \min\left\{A_{i,A} + t_i, A_{i+1,B} + t_{i+1}\right\} < \min\left\{A_{i+1,A} + t_{i+1}, A_{i,B} + t_i\right\}$

Hence, the required result is verified.

4.2 Algorithm

The following algorithm gives the procedure to determine the optimal sequence of the jobs to minimize the idle time for the machines A and B when the completion time (waiting time + service time) of the jobs coming out of Phase I is the setup times for the machine A.

Step 1. Find the mean queue length on the lines of Gupta *et al.* [8] using the formula

$$L = \frac{\rho_1}{1-\rho_1} + \frac{\rho_2}{1-\rho_2} + \frac{\rho_3}{1-\rho_3} + \frac{\rho_4}{1-\rho_4} + \frac{\rho_5}{1-\rho_5}.$$

Here,

$$\rho_1 = \frac{(\lambda_1 + \lambda_2 p_{21})}{\mu_1 (1 - p_{12} p_{21})},\ \rho_2 = \frac{(\lambda_2 + \lambda_1 p_{12})}{\mu_2 (1 - p_{12} p_{21})},\ \rho_3 = \frac{\lambda_1^{'}}{\mu_1^{'}},\ \rho_4 = \frac{\lambda_2^{'}}{\mu_2^{'}},$$

$$\rho_5 = [\ \frac{\lambda_1^{'} + \lambda_2^{'}}{\mu_3} + \frac{p_{13}(\lambda_1 + \lambda_2 p_{21}) + p_{23}(\lambda_2 + \lambda_1 p_{12})}{\mu_3 (1 - p_{12} p_{21})}\ ];$$

where λ_i is the mean arrival rate, μ_i is the mean service rate and p_{ij} are the probabilities.

Step 2. Find the average waiting time of the customers on the line of Little's [3] using relation $E(w) = \frac{L}{\lambda}$, where $\lambda = \lambda_1 + \lambda_2 + \lambda_1^{'} + \lambda_2^{'}$

Step 3. Find the completion time(C) of jobs/customers coming out of Phase I, .i.e. when processed thought the network of queues Q_1, Q_2, Q_3, Q_4 and Q_5 by using the formula

$$C = E(W) + \frac{1}{\mu_1 p_{12} + \mu_1 p_{13} + \mu_2 p_{21} + \mu_2 p_{23} + \mu_3 + \mu_1^{'} + \mu_2^{'}}.$$

Step 4. The completion time C of the customers / jobs through the network of queues Q_1, Q_2, Q_3, Q_4 and Q_5 will form the setup time for machine A. Define the two machines A and B with processing time $A_{i,A}^{'} = A_{i,A} + C$ and $A_{i,B}$.

Step 5. If t_i be the transportation time of i^{th} job from machine A to machine B. Introduce to fictitious machines G_i and H_i with processing times $G_i = A_{i,A}^{'} + t_i$ and $H_i = A_{i,B} + t_i$

Step 6. Apply modified Johnson's procedure to find the optimal sequence(s) with minimum elapsed time using theorem 4.1.

Step 7. Prepare In-Out tables for the optimal sequence(s) obtained in step 6. The sequence S_k having minimum total elapsed time will be the optimal sequence for the given problem.

5 Numerical Illustration

Consider twelve customers / jobs are processed through the network of queues Q_1, Q_2, Q_3, Q_4 and Q_5 with the servers S_1, S_2 and S_3, The server S_1 consists of two biserial service servers S_{11} and S_{12}. The server S_2 contains two parallel servers S_{21} and S_{22}. Server S_3 is commonly linked in series with each of two servers S_1 and S_2. The number of the customers, mean arrival rate, mean service rate and associated probabilities are given as in table 1.

Table 1 The detail classification of the linkage model

S. No.	No. of Customers	Mean Arrival Rate	Mean Service Rate	Probabilities
1	$n_1 = 2$	$\lambda_1 = 6$	$\mu_1 = 15$	$p_{12} = 0.6$
2	$n_2 = 3$	$\lambda_2 = 4$	$\mu_2 = 18$	$p_{13} = 0.4$
3	$n_3 = 4$	$\lambda_1' = 2$	$\mu_1' = 8$	$p_{21} = 0.4$
4	$n_4 = 2$	$\lambda_2' = 5$	$\mu_2' = 10$	$p_{23} = 0.6$
5	$n_5 = 11$		$\mu_3 = 20$	

After getting service at Phase I jobs/customers are to be served at the machines M_1 and M_2 with processing time $M_1$and M_2 respectively as given in table 2.

Table 2 The machines M_1 and M_2 with processing times

Jobs	1	2	3	4	5	6	7	8	9	10	11
$M_1(A_{i1})$	5	3	5	7	6	4	3	6	8	2	3
t_i	2	1	3	2	1	2	1	2	3	1	2
$M_2(A_{i2})$	7	4	6	8	8	5	7	4	4	5	4

The objective is to find an optimal sequence of the jobs / customers to minimize the makespan in this Queue-Scheduling linkage system by considering the first phase service into account.

Solution: We have

$$\rho_1 = \frac{\lambda_1 + \lambda_2 p_{21}}{(1 - p_{12} p_{21}) \mu_1} = 0.666667$$

$$\rho_2 = \frac{\lambda_2 + \lambda_1 p_{12}}{(1 - p_{12} p_{21}) \mu_2} = 0.55556$$

$$\rho_3 = \frac{\lambda_1'}{\mu_1'} = 0.25$$

$$\rho_4 = \frac{\lambda_2'}{\mu_2'} = 0.25$$

$$\rho_5 = \frac{\lambda_1^{'} + \lambda_2^{'}}{\mu_3} + \frac{p_{13}\left(\lambda_1 + \lambda_2 p_{21}\right) + p_{23}\left(\lambda_2 + \lambda_1 p_{12}\right)}{\mu_3\left(1 - p_{12}p_{21}\right)} = 0.85$$

Mean Queue Length = Average number of Jobs / Customers =

$$L = \frac{\rho_1}{1-\rho_1} + \frac{\rho_2}{1-\rho_2} + \frac{\rho_3}{1-\rho_3} + \frac{\rho_4}{1-\rho_4} + \frac{\rho_5}{1-\rho_5} = 10.25 \text{ units.}$$

Average waiting time of the jobs / customers = $E(w) = \frac{L}{\lambda}$ =0.602020941 units.

The total completion time of Jobs / Customers when processed through queue network in Phase I $C = E(W) + \frac{1}{\mu_1 p_{12} + \mu_1 p_{13} + \mu_2 p_{21} + \mu_2 p_{23} + \mu_3 + \mu_1^{'} + \mu_2^{'}}$.= 0.617026 units.

By taking the completion time C =0.617026 as the setup time, when jobs / customers came for processing with machine M_1. The new reduced problem with processing times $A_{i1}^{'} = A_{i1} + C$ and A_{i2} on machine M_1 and M_2 is as shown in table 3.

Table 3 The processing times $A_{i1}^{'}$ and A_{i2} on machine M_1 and M_2 is

Jobs	$A_{i1}^{'}$	t_i	A_{i2}
1	5.617026	2	7
2	3.617026	1	4
3	5.617026	3	6
4	7.617026	2	8
5	6.617026	1	8
6	4.617026	2	5
7	3.617026	1	7
8	6.617026	2	4
9	8.617026	3	4
10	2.617026	1	5
11	3.617026	2	4

The two fictitious machines with processing times G_i and H_i by considering transportation time of the jobs from machine M_1 to M_2 are as follows

Table 4 The processing times G_i and H_i on machine M_1 and M_2

Jobs	G_i	H_i
1	7.617026	9
2	4.617026	5
3	8.617026	9
4	9.617026	10
5	7.617026	9
6	6.617026	7
7	4.617026	8
8	8.617026	6
9	11.617026	7
10	3.617026	6
11	5.617026	6

Using modified Johnson's algorithm as verified by the theorem proved in section, the optimal sequence of jobs processing is

S = 10 – 7 – 2 – 11 – 6 – 1 – 5 – 3 – 4 – 9 – 8.

The In-Out flow table for the sequence S is.

Table 5 The In-Out flow table for the sequence S

Jobs	Machine M_1	Machine M_2
10	0.00000 – 2.61703	3.61703 – 8.61703
7	2.61703 – 6.23405	8.61703 – 16.617
2	6.23405 – 9.85108	16.617 – 21.617
11	9.85108 – 13.4681	21.617 – 27.617
6	13.4681 – 18.0851	27.617 – 34.617
1	18.0851 – 23.7022	34.617 – 43.617
5	23.7022 – 30.3192	43.617 – 52.617
3	30.3192 – 35.9362	52.617 – 61.617
4	35.9362 – 43.5532	61.617 - 71.617
9	43.5532 – 52.1703	71.617 – 78.617
8	52.1703 – 58.7873	78.617 – 84.617

Therefore, the total minimum elapsed time for sequence S of jobs is 84.617 units, average waiting time for the jobs is 0.60202941 units and mean queue length is10.25 units.

6 Conclusions

The present paper establishes linkage between the queue network comprised of three servers S_1, S_2, S_3 with a two stage flowshop scheduling system consisting machines M_1 and M_2. The server S_1 consists of two biserial service servers S_{11} and S_{12}. The server S_2 contains two parallel servers S_{21}, S_{22} and S_{23}. Server S_3 is commonly linked in series with each of two servers S_1 and S_2 for completion of first phase service demanded either at a subsystem S_1 or S_2.The objective of the model is to minimize the total elapsed time. A heuristic algorithm by considering the completion time of jobs in Phase I as setup time for the machine M_1 in Phase II is discussed. The study may further be extended by generalizing the number of machines and by introducing various parameters like setup time, Breakdown Interval, Job Block Criteria, etc.

References

1. Johnson, S. M., (1954). Optimal two & three stage production schedules with set up times includes. Nav. Res. Log. Quart., 1, 61-68.
2. Jackson, R. R. P., (1954). Queuing system with phase type service. O.R. Quat., 5, 109-120.
3. Little John, D. C., (1965). A proof of queuing formula: "$L = \lambda W$". Operation Research, 13, 400-412.
4. Maggu, P. L., Das, G., (1985). Elements of advance production scheduling, United Publishers and Periodical Distributors, New Delhi.
5. Maggu, P. L., (1970). Phase type service queue with two channels in Biserial, J.OP. Res. Soc Japan, 13(1).
6. Singh, T. P., (1986). On some networks of queuing & scheduling system, Ph.D., Thesis, Garhwal University, Shrinagar, Garhwal.

7. Singh, T. P., Vinod, K., Rajinder, K., (2005). On transient behaviour of a queuing network with parallel biserial queues. JMASS, 1(2), 68-75.
8. Gupta D., Singh, T.P., Rajinder, K., (2007). Analysis of a network queue model comprised of biserial and parallel channel linked with a common server. Ultra Science, 19(2) M, 407-418.
9. Vinod, K., Singh, T. P., Rajinder, K., (2007). Steady state behaviour of a queue model comprised of two subsystems with biserial linked with common channel. Reflection des ERA., 1(2), 135-152.
10. Maggu, P.L., Singh, T. P., Vinod, K., (2007). A note on serial queuing & scheduling linkage, PAMS, LXV(1), 117-118.
11. Singh, T. P., Vinod, K., (2007). On Linkage of queues in semi-series to a flowshop scheduling system. Int. Agrkult. Stat. Sci., 3(2), 571-580.
12. Singh, T. P., Vinod, K., Rajinder, K., (2008). Linkage scheduling system with a serial queue-network. Lingaya's Journal of professional studies, 2(1), 25-30.
13. Singh, T. P., Vinod, K., (2009). On linkage of a scheduling system with biserial queue network. Arya Bhatta Journal of Mathematics & Informatics, 1(1), 71- 76.
14. Gupta, D., Sharma, S., Sharma, S., (2012). On linkage of a flow shop scheduling model including job block criteria with a parallel biserial queue network. Computer Engineering and Intelligent Systems, 3(2), 17-28.

Appendix

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>
#include<process.h>
#include<math.h>

int n[4],u[5],L[4];
int j[50],j1[50],m1;
float p[4];
float r[5];
float g[50],h[50],a[50],b[50],t1[50],g1[50],h1[50];
float a1,b1,a2,b2,a3,b3,a4,b4,b5,c1,c2,P,Q,V,W,M,z1,z2,z3,x;
float q1,q2,q3,z,f,c;
void main()
{

        clrscr();
for( int i=1;i<=4;i++)
        {
        cout<<"Enter the number of customers and  Mean Arrival Rate for Channel S"<<i<<":";
        cin>>n[i]>>L[i];
        }
m1=n[1]+n[2]+n[3]+n[4];
for(int d=1;d<=5;d++)
        {
        cout<<"\nEnter the Mean Service Rate for the Channel S"<<d<<":";
        cin>>u[d];
        }
for(int k=1;k<=4;k++)
        {
```

```
        cout<<"\nEnter the value of probability p"<<k<<":";
        cin>>p[k];
        }
for(i=1;i<=m1;i++)
        {
        j[i]=i;
        cout<<"\nEnter the processing time  of "<<i<<" job for machine A : ";
        cin>>a[i];
        cout<<"\nEnter the transportation time of "<<i<<" job form machine A to machine B:
";
        cin>>t1[i];
        cout<<"\nEnter the processing time  of "<<i<<" job for machine B : ";
        cin>>b[i];
        }
        a1=L[1]+L[2]*p[3];
        b1=(1-p[1]*p[3])*u[1];
        r[1]=a1/b1;
        a2=L[2]+L[1]*p[1];
        b2=(1-p[1]*p[3])*u[2];
        r[2]=a2/b2;
        a3=L[3];a4=L[4];b3=u[3];b4=u[4],b5=u[5];
        r[3]=a3/b3;
        r[4]=a4/b4;
        c2=(1-p[1]*p[3])*b5;
        z1=(a3+a4)/b5;
        z2= a1*p[2]/c2;
        z3= a2*p[4]/c2;
        r[5]=z1+z2+z3;
        M=L[1]+L[2]+L[3]+L[4];
for(i=1;i<=5;i++)
        {
        cout<<"r["<<i<<"]\t\t"<<r[i]<<"\n";
        }
        for(i=1;i<=5;i++)
        {
        if(r[i]>1)
        {
                cout<<"Steady state condition does not holds good for"<<r[i]<<"...\nExitting";
                getch();
                exit(0);
        }
        }
        Q =(r[1]/(1-r[1]))+(r[2]/(1-r[2]))+(r[3]/(1-r[3]))+(r[4]/(1-r[4]))+(r[5]/(1-r[5]));
        cout<<"\nThe mean queue length is :"<<Q<<"\n";
        W=Q/M;
        cout<<"\nAverage waiting time for the customer is:"<<W<<"\n";
        z=u[1]*p[1]+u[1]*p[2]+u[2]*p[3]+u[2]*p[4]+u[3]+u[4]+u[5];
        f=1/z;
```

```
        c= W+f;
        cout<<"\n\nTotal completetion time of Jobs / Customers through Queue Network in Phase 1 :"<<c;
        for(i=1;i<=m1;i++)
        {
                g1[i]=a[i]+c;
                h1[i]=b[i];
        }
        for(i=1;i<=m1;i++)
        {
                g[i]=g1[i]+t1[i];
                h[i]=h1[i]+t1[i];
        }
for(i=1;i<=m1;i++)
        {
        cout<<"\n\n"<<j[i]<<"\t"<<g1[i]<<"\t\t\t"<<t1[i]<<"\t\t"<<h1[i];
        cout<<endl;
        }
float mingh[16];
char ch[16];
for(i=1;i<=m1;i++)
        {
        if(g[i]<h[i])
                {
                        mingh[i]=g[i];
                        ch[i]='g';
                }
        else
                {
                        mingh[i]=h[i];
                        ch[i]='h';
                }
        }
        for(i=1;i<=m1;i++)
        {
        cout<<endl<<mingh[i]<<"\t"<<ch[i];
        }
        for(i=1;i<=m1;i++)
        {
                for(int k=1;k<=m1;k++)

                                if(mingh[i]<mingh[k])
                                        {
                                                float temp=mingh[i]; int temp1=j[i]; char d=ch[i];

                                                mingh[i]=mingh[k]; j[i]=j[k]; ch[i]=ch[k];
                                                mingh[k]=temp; j[k]=temp1; ch[k]=d;
                                                }
```

```
          }
            for(i=1;i<=m1;i++)
                    {
                    cout<<endl<<endl<<j[i]<<"\t"<<mingh[i]<<"\t"<<ch[i]<<"\n";
                    }

// calculate  scheduling
float sbeta[16];

int t=1,s=0;
for(i=1;i<=m1;i++)
          {
          if(ch[i]=='h')
          {
           sbeta[(m1-s)]=j[i];
           s++;
          }
else      if(ch[i]=='g')
          {
          sbeta[t]=j[i];
          t++;
          }
          }
int arr1[16], m=1;
          cout<<endl<<endl<<"Job Scheduling:"<<"\t";
for(i=1;i<=m1;i++)
          {
          cout<<sbeta[i]<<" ";
          arr1[m]=sbeta[i];
          m++;
          }
//calculating total computation sequence
  float macha[50], machb[50],maxv1[50];
          float time=0.0;
          macha[1]=time+g1[arr1[1]];
for(i=2;i<=m1;i++)
          {macha[i]=macha[i-1]+g1[arr1[i]];}
machb[1]=macha[1]+h1[arr1[1]]+t1[arr1[1]];
          for(i=2;i<=m1;i++)
          {
if((machb[i-1])>(macha[i]))
          maxv1[i]=machb[i-1];
else
          maxv1[i]=macha[i];machb[i]=maxv1[i]+h1[arr1[i]]+t1[arr1[i]];
          }
//displaying solution
cout<<"\n\n\n\n\n\t\t\t    #####THE SOLUTION##### ";
cout<<"\n\n\t***************************************************************";
```

```
cout<<"\n\n\n\t    Optimal Sequence is : ";
for(i=1;i<=m1;i++)
        {cout<<" "<<arr1[i];}
cout<<endl<<endl<<"In-Out Table is:"<<endl<<endl;
cout<<"Jobs"<<"\t"<<"Machine M1"<<"\t"<<"\t"<<"Machine M2" <<"\t"<<endl;
cout<<arr1[1]<<"\t"<<time<<"--"<<macha[1]<<"        \t"<<"\t"<<t1[arr1[1]]+macha[1]<<"--
"<<machb[1]<<" \t"<<endl;
for(i=2;i<=m1;i++)
        {
cout<<arr1[i]<<"\t"<<macha[i-1]<<"--"<<macha[i]<<"                 "<<"\t"<<maxv1[i]<<"--
"<<machb[i]<<" "<<"\t"<<endl;
        }
cout<<"\n\n\nTotal Elapsed Time (T) = "<<machb[m1];
cout<<"\n\n\t*****************************************************************";
getch();
}
```