Scheduling for improving productivity of the automated manufacturing system

Jungsang Choi*and Gilsang Jang**

*Dept. of Industrial Engineering, Kangnam University, Kyonggi 449-702, Korea

** Dept. of Industrial Information Systems, Dongguk University, Kyongbuk 780-714 , Korea

Abstract

In this paper jobshop scheduling problem was considered on automated manufacturing systems with the closed loop and unidirectional material handling system. The objective of this research is to develop and evaluate heuristic scheduling procedures to improve productivity by minimizing makespan. Especially travel time of material handling system as well as processing time was considered in the proposed algorithms. A new heuristic algorithms are proposed and illustrates the proposed algorithm. The heuristic algorithms are implemented for various cases. The results show that the proposed algorithms provide better solutions in productivity, frequency, job waiting time and the number of waiting jobs than the random scheduling algorithm.

1. Introduction

The advanced manufacturing system such as flexible manufacturing systems aims at the realization of fully automated manufacturing. Many kinds of the automated facilities, controlled by computers, have been developed for this purpose, e.g., machining centers for machining, automated guided vehicles for transporting material, robot hands for loading/ unloading and inspection units [1].

Advanced manufacturing is a concept has been widely accepted for increasing manufacturing productivity and flexibility. Significant reduction in setup time, tooling needs, and work-in-process, simplified flow of parts and tools, and improved human relations have been the most quoted advantages in the application of advanced manufacturing systems [1][10]. Recent innovations in manufacturing systems such as the flexible manufacturing systems (FMS) are aimed at the realization of fully automated manufacturing. Recent years have seen the development of several automated computer-controlled devices for the purpose of increasing manufacturing productivity. Examples of these devices machining center, automated guided vehicles (AGV), industrial robots, and coordinate measuring machines.

As it has pointed clearly in the literature, these devices have the potential to improve productivity. However, such increased productivity may not be realized unless the operational issues are well planned and controlled. Of the many control problems present in a manufacturing system, job scheduling is, perhaps, one of the most important issues. However, the generation of consistently good schedules has proven to be extremely difficult, especially when the issues of material handling time are simultaneously considered.

Two machine flow-shop scheduling problem with transportation time of vehicle have been discussed in Maggu et al [8] with the assumption that the input and output buffers have infinite capacity. This implies that no blocking of machines occurs. Stern and Vinter [11] discuss a two-machine flow-shop model involving travel time found the problem to be NP hard. Kise et al [7] discussed the two-machine flow-shop scheduling problem with handling time consideration using vehicles. Their model assumed that there is no buffer for work-in-process. Choi [1] proposed a heuristic algorithm for minimizing completion time of jobs with handling time consideration for an n-machine system in which the machines are arranged around a unidirectional loop. He also described an procedures for solving a multi-objectives, two-stage with one transporter and unlimited buffer [2],[3],[4],[5]. However, there is relatively few reported research of job shop scheduling problem with material handling time consideration.

The general job shop scheduling problem is of perennial interest because of its direct relevance to practical manufacturing problems and because of the moderate level progress that has been obtained toward a computationally feasible solution. Many approaches to the problem have been suggested. Among these are expert system method [10], neural network method [4], heuristics [3], and combinatorial optimization [9][12]. In this paper the problem of scheduling multiple jobs through a flexible manufacturing system or general job shop with two machine stations is addressed. Details of the design and operation of the system are described in the next section.

2. System description

The figure 1 shows a manufacturing shop consisting of two automated workstations and the closed loop material handling system with unidirection. Each workstation has inbound and outbound buffer. When a job arrives at the workstation, it is loaded on the machine by a robot. Otherwise, the job stays in the buffer. Completed jobs also are unloaded from the machines to the output buffer by the robots.

All jobs enter and leave the system through an input/output (I/O) point. From the I/O point, inbound jobs are transferred to any of the workstations by a shuttle cart. The shuttle cart is also used to transfer jobs between the workstations. The cart operates on the closed loop track and this allows the shuttle to move in unidirection.

The system is an automated job shop. Therefore, jobs processed in the system do not have a uniform flow pattern. Some jobs flow from machine 1 to machine 2 while others flow from machine 2 to machine. A third class of jobs requires processing on only one machine to have their processing needs satisfied.

Given the production described, the problem of interest is to find a schedule that would minimize the makespan of jobs loaded into the system. It is assumed that all jobs considered for loading in a given scheduling instance are available at the start of the beginning of the scheduling session.

Therefore, the objective of this research is to develop a production schedulig which would improve productivity by minimizing makespan of a set of jobs released for production through a flexible manufacturing system during a given production instance.

The FMS is composed of two workstations and a closed loop and unidirectional material handling system for transport jobs between the workstations and input/output station. In the paper, we present dispatching rule of AGV and three algorithms to minimize the production makespan for improving productivity in an automated system as described.

3. Mathematical Modeling

In formulating the problem, the following

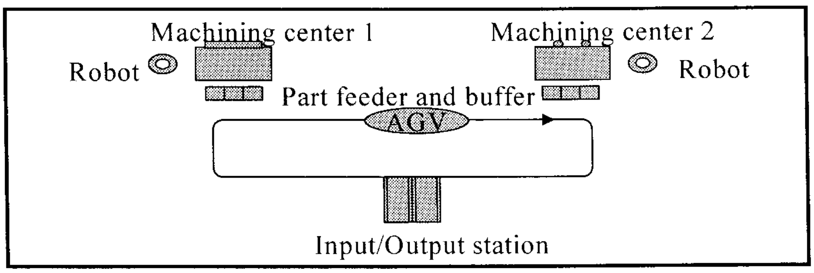


Figure 1 FMS with two-workstations and an uni-directional AGV

notations used throughout the paper are presented.

{1} : set of jobs that require processing on machine 1 only.

{2} : set of jobs that require processing on machine 2 only.

{12} : set of jobs that require processing on both machines 1 and 2 but starts from machine 1.

{21} : set of jobs that require processing on both machines 1 and 2 but starts from machine 2.

S_{12} : set of jobs that visit machine 1 followed by machine 2 sequenced according to Johnson's rule[6].

S_{10} : set of jobs that require processing on machine 1 only sequenced according to increasing order of processing time.

S_{21} : set of jobs that visit machine 2 followed machine 1 sequenced according to Johnson's rule.

S_{20} : set of jobs that require processing on machine 2 only sequenced according to increasing order of processing time.

S_1 : set of all jobs that require processing on machine 1 ordered according to Jackson's rule[5].

$\{S_{12} \rightarrow S_{10} \rightarrow S_{21}\}$

S_2 : set of jobs that require processing on machine 2 ordered according to Jackson's rule.

$\{S_{21} \rightarrow S_{20} \rightarrow S_{12}\}$

i : job index in S_1 (i = 1,2,3,...,n)

j : job index in S_2 (j = 1,2,3,...,m)

n_{12} : number of jobs in set {12}

M_h : machine h, h= 0,1,2, (M_0 means loading/unloading station)

$P_{i,1}$: processing time of job i at M_1

$P_{j,2}$: processing time of job j at M_2

$B_{i,1}$: beginning time of job i at M_1

$B_{j,2}$: beginning time of job j at M_2

$C_{i,1}$: completion time of job i at M_1

$C_{j,2}$: completion time of job i at M_2

T_{mn} : travel time of vehicle from machine m(or loading/unloading station) to machine n (or loading /unloading station)

$A_{i,1}$: arrival time of vehicle with job i at M_1

$A_{j,2}$: arrival time of vehicle with job j at M_2

$W_{i,1}$: waiting time of vehicle to load job i which has finished processing at M_1

$W_{j,2}$: waiting time of vehicle to load job j which has finished processing at M_2

$D_{i,1}$: departure time of vehicle loaded job i which has finished processing at M_1

$D_{j,2}$: departure time of vehicle loaded job j which has finished processing at M_2

$J_{i,1}$: waiting time of job i which has finished processing at M_1 to be picked up by vehicle

$J_{j,2}$: waiting time of job j which has finished processing at M_2 to be picked up by vehicle

C_{max} : makespan

In developing the model, the following assumptions are made:

1. A machine can process only one job at a time.
2. A machine cannot interrupt processing once it has started an operation (No preemption.).
3. The shuttle or vehicle can transfer only one job at a time.
4. Job pickup time is included in travel time of vehicle.
5. The buffer at the I/O point has an infinite capacity.
6. Jobs arriving at a station are processed according to first come-first serve (FCFS) rule. They are also picked from the output buffer of machines according to the same rule.
7. When the shuttle is ready for a new assignment, the next task to assign the vehicle determined at the vehicle ready time.

Because no actual processing of jobs take place at the load/unload point and given the assumption that all jobs are ready at time zero, this implies that $B_{i0} = A_{i0} = D_{i0} = C_{i0}$ if i=1. Therefore if i=1 and no job had been delivered to M_2, then $B_{1,1}=A_{1,1}=T_{01}$. For all other subsequent jobs processed on machine M_1, the start time on the machine is as expressed in equation (1).

$$\begin{aligned} B_{i,1} &= Max\left\{ C_{i-1,1},\ \ A_{i,1}\right\} \\ &= Max\left\{ C_{i-1,1},\ \ D_{i,o}+T_{o1}\right\} \end{aligned} \tag{1}$$

If the transporter departs M_1 and to M_0 to pickup a load and then returns to M_1 without following the load pickup, then the departure time of the new load i (i.e., $D_{i,o}$) from the I/O point to M_1 is as given in equation (2) below. The variable k in the equations (2) should take the maximum value possible that would not cause the FCFS rule for load pickup from a machine station to be violated. $D_{i,o}=0$ if i=1 and no job had been delivered to M_2. Otherwise

$$\begin{aligned} D_{i,o} &= A_{i-1,1} + YW_{i-k,1} + T_{1o}, \quad {\scriptstyle 1\le k \langle i \ne 1} \\ &= C_{i-k,1} + J_{i-k,1} + T_{1o}, \quad {\scriptstyle 1\le k \langle i \ne 1} \\ \textit{where}\ W_{i-k,1} &= Max\left\{ 0,\ C_{i-k,1} - A_{i-1,1}, \quad {\scriptstyle 1\le k \langle i \ne 1}\right\} \\ J_{i-k,1} &= Max\left\{ 0,\ A_{i-1,1} - C_{i-k,1}, \quad {\scriptstyle 1\le k \langle i \ne 1}\right\} \\ Y &= \begin{cases} 0, & \textit{if}\ \ W_{i-k,1} \rangle\, T_{1o} + T_{o1}, \quad {\scriptstyle 1\le k \langle i \ne 1} \\ 1, & \textit{otherwise} \end{cases} \end{aligned} \tag{2}$$

Graphically, the movement of the transporter is as illustrated in Figure 2.

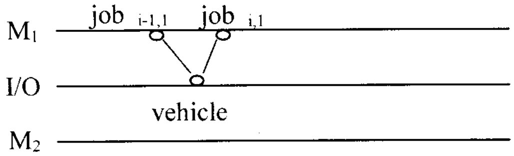


Figure 2: Transporter moves pattern when it revisits M_1 from M_0

On the other hand, if the vehicle visits M_1 from M_2 through M_0 and picks up load i from the I/O point, then the departure time of load i, $D_{i,0}$ from the I/O point is as given in equation (3) below. The variable p in the equations (3) should take the maximum value possible that would not cause the FCFS rule for load pickup from a machine to be violated. Note that when j=1 in equation (3), $W_{j-p,2}$ = $W_{0,2}$, $J_{j-p,2}$ =$J_{0,2}$ and $C_{j-p,2}$ =$C_{0,2}$ are zero.

$$
\begin{aligned}
D_{i,o} &= A_{j,2} + YW_{j-p,2} + T_{2o}, \quad 1 \le p \langle j \\
&= C_{j-p,2} + J_{j-p,2} + T_{2o}, \quad 1 \le p \langle j \\
\textit{where } W_{j-p,2} &= Max\left\{ 0,\ C_{j-p,2} - A_{j,2}, \quad 1 \le p \langle j \right\} \\
J_{j-p,2} &= Max\left\{ 0,\ A_{j,2} - C_{j-p,2}, \quad 1 \le p \langle j \right\} \\
Y &= \begin{cases} 0, & \textit{if } W_{j-p,2} \rangle T_{2o} + T_{o2}, \quad 1 \le p \langle j \\ 1, & \textit{otherwise} \end{cases}
\end{aligned}
\tag{3}
$$

Graphically, this is also equivalent to the movement of vehicle shown in Figure 3.

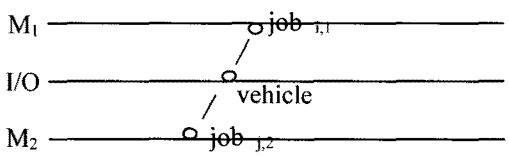


Figure 3: Transporter moves from M_2 through the I/O point to M_1.

The expression Bi,1 depends on the move pattern of the vehicle as shown in the equation (2) and (3). Therefore, equations(4) and (5) can be obtained by substituting the equations (2) and (3) into equation (1). If the vehicle revisits M_1 from M_0 before it undertakes any other movement, then

$$
\begin{aligned}
B_{i,1} &= Max\left\{ C_{i-1,1},\quad A_{i-1,1} + YW_{i-k,1} + T_{1o} + T_{o1}, \quad 1 \le k \langle i \ne 1 \right\} \\
&= Max\left\{ C_{i-1,1},\quad C_{i-k,1} + J_{i-k,1} + T_{1o} + T_{o1}, \quad 1 \le k \langle i \ne 1 \right\}
\end{aligned}
\tag{4}
$$

On the other hand, if the vehicle visits M_1 from M_2 without delay at M_0, then

$$
\begin{aligned}
B_{i,1} &= Max\left\{ C_{i-1,1},\quad A_{j,2} + YW_{j-p,2} + T_{2o} + T_{o1}, \quad 1 \le p \langle j \right\} \\
&= Max\left\{ C_{i-1,1},\quad C_{j-p,2} + J_{j-p,2} + T_{2o} + T_{o1}, \quad 1 \le p \langle j \right\}
\end{aligned}
\tag{5}
$$

Therefore, the completion time $C_{i,1}$ of job i at M_1 is as given in equation (6).

$$
C_{i,1} = B_{i,1} + P_{i,1} \tag{6}
$$

The departure time $D_{i-k,1}$ of job i-k after it has been processed at M_1 given that job i-k has already completed processing before the vehicle arrives at M_1 is as given in equation (7)

$$
D_{i-k,1} = C_{i-k,1} + J_{i-k,1}, \quad 1 \le k \langle i \ne 1 \tag{7}
$$

On the other hand, if the vehicle has to wait at waits M_1 to pickup job i-k, then

$$
D_{i-k,1} = A_{i,1} + W_{i-k,1}, \quad 1 \le k \langle i \ne 1 \tag{8}
$$

Like the equations developed for job i at M_1, we can formulate similar equations for job j at M_2. If the first job delivery of a job j =1 for processing is to machine 2, then

$D_{1,0} = 0$, $C_{0,2} = 0$, and $B_{1,2} = A_{1,2} = T_{0,2}$

For all other subsequent jobs processed on machine M_2, the processing start time $B_{j,2}$ of job j at M_2 is given in equation (9).

$$B_{j,2} = Max\left\{ C_{j-1,2},\quad A_{j,2} \right\}$$
$$= Max\left\{ C_{j-1,2},\quad D_{j,o} + T_{o2} \right\} \tag{9}$$

When the vehicle leaves M_2 to M_0, picks up a load j from M_0 and revisits M_2 from M_0 before it undertakes any other movement, the vehicle departure time from M_0, given by $D_{j,0}$ is as expressed in equation (10). Again, the variable p in the equations (10) should take the maximum value possible that would not cause the FCFS rule for load pickup from machine 2 to be violated. $D_{j,o}=0$ if j=1 and no job had been delivered to M_1. Otherwise

$$D_{j,o} = A_{j-1,2} + YW_{j-p,2} + T_{2o}, \quad 1 \le p \langle j \ne 1$$
$$= C_{j-p,2} + J_{j-p,2} + T_{2o}, \quad 1 \le p \langle j \ne 1$$
$$where\ W_{j-p,2} = Max\left\{ 0,\ C_{j-p,2} - A_{j-1,2},\quad 1 \le p \langle j \ne 1 \right\}$$
$$J_{j-p,2} = Max\left\{ 0,\ A_{j-1,2} - C_{j-p,2},\quad 1 \le p \langle j \ne 1 \right\}$$
$$Y = \begin{cases} 0, & if\ W_{j-p,2} \rangle T_{2o} + T_{o2}, \quad 1 \le p \langle j \ne 1 \\ 1, & otherwise \end{cases} \tag{10}$$

Graphically, the move pattern of the vehicle is as illustrated in Figure 4 below.

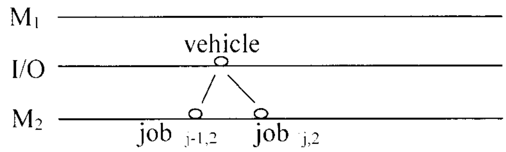


Figure 4: Transporter moves pattern when it revisits M_2 from M_0 before making some other moves from M_2.

On the other hand, when the vehicle travels from M_1 to M_0, picks up a load j from M_0 to M_2, then the departure time, $D_{j,o}$, of load j from M_0 is as given in equation (11). The variable k in equations (11) should take the maximum value possible that would not cause the FCFS rule for load pickup from M_2 to be violated. In equation (11), for i =1, $W_{i-k,1} = W_{0,1}$, $J_{i-k,1} = J0,1$, and $C_{i-k,1} = C_{0,1}$.

Graphically, the movement pattern of the transporter is as illustrated in Figure 5.

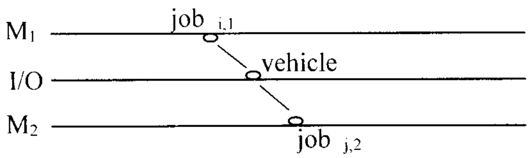


Figure 5: Transporter moves pattern as it travels to M_2 from M_1 with a pickup of load j at M_0

$$D_{j,o} = A_{i,1} + YW_{i-k,1} + T_{1o}, \quad 1 \le k \langle i$$
$$= C_{i-k,1} + J_{i-k,1} + T_{1o}, \quad 1 \le k \langle i$$

$$where \quad W_{i-k,1} = Max\{\ 0,\ C_{i-k,1} - A_{i,1},\ \ {}_{1 \le k \langle i}\}$$
$$J_{i-k,1} = Max\{\ 0,\ A_{i,1} - C_{i-k,1},\ \ {}_{1 \le k \langle i}\}$$
$$Y = \begin{cases} 0, & if \quad W_{i-k,1} \rangle\ T_{1o} + T_{o1}, \quad {}_{1 \le k \langle i} \\ 1, & otherwise \end{cases} \qquad (11)$$

The expression for the start time, Bj,2, for load j on machine M_2 depends on the movement pattern of the vehicle as described in the equation (10) and (11). Therefore we can express $B_{j,2}$, as given in equations (12) and (13) by substituting the equations (10) and (11) into (9).

For the case where the vehicle visits M_0 from M_2, picks up a load from Mo and returns to M_2 before making any other move, then

$$B_{j,2} = Max\left\{C_{j-1,2},\ \ A_{j-1,2} + YW_{j-p,2} + T_{2o} + T_{o2},\ \ {}_{1 \le p \langle j \ne 1}\right\}$$
$$= Max\left\{C_{j-1,2},\ \ C_{j-p,2} + J_{j-p,2} + T_{2o} + T_{o2},\ \ {}_{1 \le p \langle j \ne 1}\right\} \qquad (12)$$

For the case where the vehicle moves from M_1 to M_2 through M_0, then

$$B_{j,2} = Max\left\{C_{j-1,2},\ \ A_{i,1} + YW_{i-k,1} + T_{1o} + T_{o2},\ \ {}_{1 \le k \langle i}\right\}$$
$$= Max\left\{C_{j-1,2},\ \ C_{i-k,1} + J_{i-k,1} + T_{1o} + T_{o2},\ \ {}_{1 \le k \langle i}\right\} \qquad (13)$$

Therefore, the completion time $C_{j,2}$ of job j at M_2 can be expressed as in equation (14).

$$C_{j,2} = B_{j,2} + P_{j,2} \qquad (14)$$

The departure time $D_{j-p,2}$ of job j-p that was processed at M_2, if it (i.e., job j-p) has already completed processing before the vehicle arrives at M_2, then

$$D_{j-p,2} = C_{j-p,2} + J_{j-p,2}\,, \qquad {}_{1 \le p \langle j \ne 1} \qquad (15)$$

Otherwise, if the vehicle waits to pickup job j-p at M_2, then

$$D_{j-p,2} = A_{j,2} + W_{j-p,2}\,, \qquad {}_{1 \le p \langle j \ne 1} \qquad (16)$$

Finally, given that job n and m were the last to be processed on machines M_1 and M_2 respectively, then the total completion time for all jobs is as given in equation (17).

$$C_{\max} = Max\left\{A_{n,o},\ \ A_{m,o}\right\}$$
$$= Max\left\{C_{n,1} + J_{n,1} + T_{1o},\ \ C_{m,2} + J_{m,2} + T_{2o}\right\}$$
$$= Max\left\{A_{n,1} + W_{n,1} + T_{1o},\ \ A_{m,2} + W_{m,2} + T_{2o}\right\} \qquad (17)$$

Therefore, the problem of minimizing the completion time of the jobs breaks down to the expression given in equation (18).

$$Min\ Z = C_{\max} \qquad (18)$$

4. The proposed algorithms

Johnson's rule provides the optimal

solution to minimize the makespan in a flow shop with two machine, Jackson's rule extended Johnson's rule to provide the optimal sequence in job shops with two machine, too. However, if the travel time of job between machines is considered, both rules fail to not guarantee the optimal solution. To illustrate the point, consider a problem as shown below, where the arrows indicate the direction of flow of the jobs.

Job		Processing Time M_1		Processing Time M_2
	1	16	—	12
	2	23	—	21
	3	14	—	11

If $T_{12} = T_{21} = 0$, the model is the classical two machine flowshop problem. Therefore Johnson's rule provides optimal solution in sequence 2→1→3 and has a makespan of 67. However, if it is assumed that $T_{12} = T_{21} = 10$ time units, the sequence 2→1→3 has a makespan of 84. Since there is a sequence 3→2→1 that provides a makespan of 80, then Johnson's rule does not provide the optimal solution in the two machine flowshop problem with vehicle travel time consideration

Similarly, Jackson's rule provides optimal sequence in job shops with two machines when travel times are not considered. However, if the travel times of vehicle are considered, Jackson's rule does not guarantee the optimal solution any more. This is illustrated by the problem below.

Job		Processing Time M_1		Processing Time M_2
	1	25	—	17
	2	18	—	15
	3	21	—	28
	4	21	—	24
	5	14	—	17

When travel time is negligible, the above problem is the classical two machine job shop problem and using Jackson's rule provides optimal sequence 3→1→2→4→5 on machine 1 and sequence 4→5→3→1→2 on machine 2 and has a makespan of 101. However, if travel time is considered and $T_{10} + T_{02} = T_{20} + T_{01} = 10$ time units, Jackson's rule provides a solution with a makespan of 134. We can find a better solution in sequence 1→2→3→4→5(or 1→3 →2→4→5) at machine 1 and 4→5→1→2→3(or 4→5→1→3→2) at machine 2. These sequences produce a makespan of 132. Thus, Jackson's rule does not guarantee an optimal solution in the two machines job shop problem when travel times are considered.

In addition to the job sequence, the method of controlling and routing the vehicle also affects the quality of solution obtained. Therefore, both the job sequence on machines and the travel pattern of the transporter have to be simultaneously considered in seeking a solution to the two machines job shop problem with travel time consideration. For this reason, it is necessary to develop a new and efficient algorithm in seeking a solution to the job shop problem with travel time consideration. We present below, three such algorithms developed as part of this research.

4.1 Job sequencing method

1) algorithm #1

a) Schedule the jobs in {1} in increasing order of their processing times to give the sequence S_{10}.

b) Schedule the jobs in{2} in increasing order of their processing times to give the sequence S_{20}.

c) Schedule the jobs in {12} according to the following steps.

Step 1. Calculate the initial job waiting time at machine 1.

$J_{i,1}$=Max[0, $(T_{01}+T_{10}+T_{02}+T_{20})-P_{i,1}$],

i=1, 2, 3,...., n_{12}

Step 2. First, sequence the jobs with positive Ji,1 in decreasing order of $J_{i,1}$, and then, schedule the other jobs according to Johnson's rule. Let the sequence produce be S_{12}

d) Schedule the jobs in{21} according to the same procedures as S_{12}, recognizing that M_2 is now the first machine and M_1 the second machine. Let the sequence be S_{21}

e) The combined job sequence on machine 1 becomes $S_{12} \rightarrow S_{10} \rightarrow S_{21}$ and on machine 2, the sequence is $S_{21} \rightarrow S_{20} \rightarrow S_{12}$.

2) algorithm #2:

a) Schedule the jobs in {1} and {2}according to the procedures described in Algorithm 1.

b) Schedule the jobs in{12} according to the following steps:

Step 1. Generate a schedule S according to Johnson's rule and calculate the makespan Cmax of a sequence S. S[k]=a sequence that places k-th job of S at the first position and leaving the rest of the sequence unaltered, where k=1, 2, 3,...,n_{12}.

Set k=1, S[k] = S, and Cmax[k] = Cmax.

Step 2. Set k=k+1.

Calculate the initial waiting time of each job at machine 1.

$J_{k,1}$ = Max[0, $(T_{o1}+T_{1o}+T_{o2}+T_{2o})-P_{k,1}$], where k=2, 3, 4,.., n_{12} and k means the k-th job of the sequence S.

Step 3. If $J_{k,10}$, then obtain the new sequence S[k] by moving the k-th job to the first position of S and calculate its makespan Cmax[k], otherwise Cmax [k]= M > Cmax[1]. If k=n12, go to step 4, otherwise go to step 2.

Step 4. If Cmax[k] = Min [Cmax [k], k =1,2,3,···,n_{12}] then S [k]is the sequence for S_{12}.

c) Schedule the jobs in{21} according to the same procedures as S_{12}, recognizing that M_2 is now the first machine and M_1 the second machine. Let the sequence be S_{21}.

d) The combined job sequence on machine 1 becomes $S_{12}S_{10}S_{21}$ and on machine 2, the sequence is $S_{21}\ S_{20}S_{12}$.

3) algorithm #3:

a) Schedule the jobs in {1} and {2} according to the same procedures described in Algorithm1.

b) Schedule the jobs in{12} according to the procedure described below.

Step 1. Let $A_{i,1}$ = Max[(T_{01} + T_{10} +T_{02} +T_{20}), $P_{i,1}$] and set $B_{i,2}$= $P_{i,2}$. Generate a schedule S according to Johnson's rule and Calculate the makespan Cmax of schedule S.

S[k]= a sequence that places the k-th job of S at the first position and leaving the rest of the sequence unchanged, k=1, 2, 3,...,n_{12}.

Set k=1, S[k] = S, and Cmax[k] = Cmax .

Step 2, 3, 4 are the same as in Algorithm #2.

c) Schedule the jobs in{21} according to the same procedures as S_{12}, recognizing that M_2 is now the first machine and M_1 the second machine. Let the sequence be S_{21}.

d) The combined job sequence on machine 1 becomes $S_{12}S_{10}S_{21}$ and on machine 2, the sequence is $S_{21}S_{20}S_{12}$.

4.2 Dispatching rule of vehicle

To accelerate the flow of jobs through the system, decision on which workstation to next route the transporter is made dynamically. To initiate the process, it is assumed that the first load transfer from the I/O point to a machine is sent to the station that has more jobs to be processed. Let Q_2 be the queue size at the I/O point of M_2. We present the rules for deciding on what next the transporter has to do next, assuming it is located at station 1.

1) Initialization

i': a job that has completed processing at M_1 and turned back to I/O, i'=1,2,3,..., n.

j': a job that has completed processing at M_2 and turned back to I/O, j'=1,2,3,..., m.

R_1: the number of jobs that require processing at M_1, $0 \leq R_1 \leq n$.

R_2: the number of jobs that require processing at M_2, $0 \leq R_2 \leq m$.

Step 1. Set i=j=0, i'=j'=0, R_1=n, R_2=m.

Step 2. If

$$\sum_{k=i+1}^{n} P_{i,1} + R_1(T_{o1} + T_{1o}) \geq \sum_{k=j+1}^{m} P_{j,2} + R_2(T_{o2} + T_{2o}),$$

vehicle delivers job i+1 from M_o to M_1 Set i=i+1, R_1=R_1-1and go to Step 3.

Otherwise delivers job j+1 from M_o to M_2, and set j=j+1, R_2=R_2-1 and go to step 4.

Step 3. 1) If $i+1=1$ and $j=0$, then vehicle goes to Mo without waiting at M1 and set $i=i+1$,
$R_1=R_1-1$. Go to Step 2.
2) If $i+1=1$ and $j\neq 0$, then vehicle goes to M_o without waiting at M_1. Go to Part C.
3) If $i+1\neq 1$, go to Part A.

Step 4. 1) If $j+1=1$ and $i=0$, then vehicle goes to M_o without waiting at M_2 and set $j=j+1$,
$R_2=R_2-1$. Go to Step 2.
2) If $j+1=1$ and $i\neq 0$, then vehicle goes to M_o without waiting at M_1. Go to Part C.
3) If $j+1\neq 1$, go to Part B.

Figure 6 shows the procedures for initialization to assign vehicle in graph.

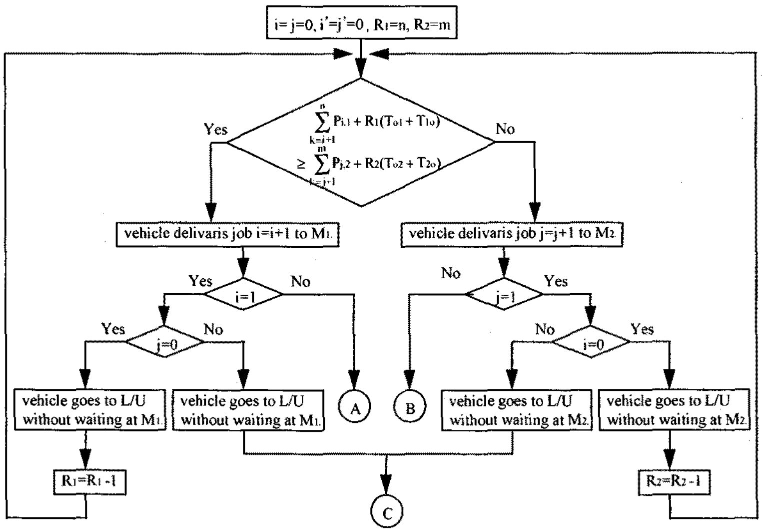
2) Part A : The method of assigning vehicle at machine 1.

F1 : the number of jobs completed wait for being picked up at M_1 after having completed processing, $0\leq F_1<n$.

Step 1. If $F_1=0$, then go to step 2. Otherwise vehicle picks up the completed job according to FCFS. Set $i'=i'+1$ and go to Part C.

Step 2. If $W_{i-k,1}<T_{1o}+T_{o1}$, then go to step 3. Otherwise go to step 4.

Step 3. If $j'=m$, then vehicle waits for pick up i-k at M_1, goes to I/O. Go to



i': a job that has completed processing at M_1 and turned back to loading/unloading station(L/U) ($i'=1,2,3,\ldots,n$).
j': a job that has completed processing at M_2 and turned back to loading/unloading station(L/U) ($j'=1,2,3,\ldots,m$).
R_1: the number of jobs that require processing at M_1. ($0\leq R_1\leq n$)
R_2: the number of jobs that require processing at M_2. ($0\leq R_2\leq m$)

Figure 6. Flow chart for initialization to assign vehicle.

Part C. Otherwise go to step 5.

Step 4. If j'=m, then vehicle waits for pick up at M_1 and goes to I/O.
Set i'=i'+1 and go to Part C.
Otherwise go to step 2 of Part C.

Step 5. If $W_{i-k,1}<T_{1o}+T_{o2}$, then vehicle waits for pick up i-k and goes to I/O. Go to Part C.
Otherwise go to step 6.

Step 6. Vehicle goes to I/O without waiting for pick up i-k . Go to Part C.

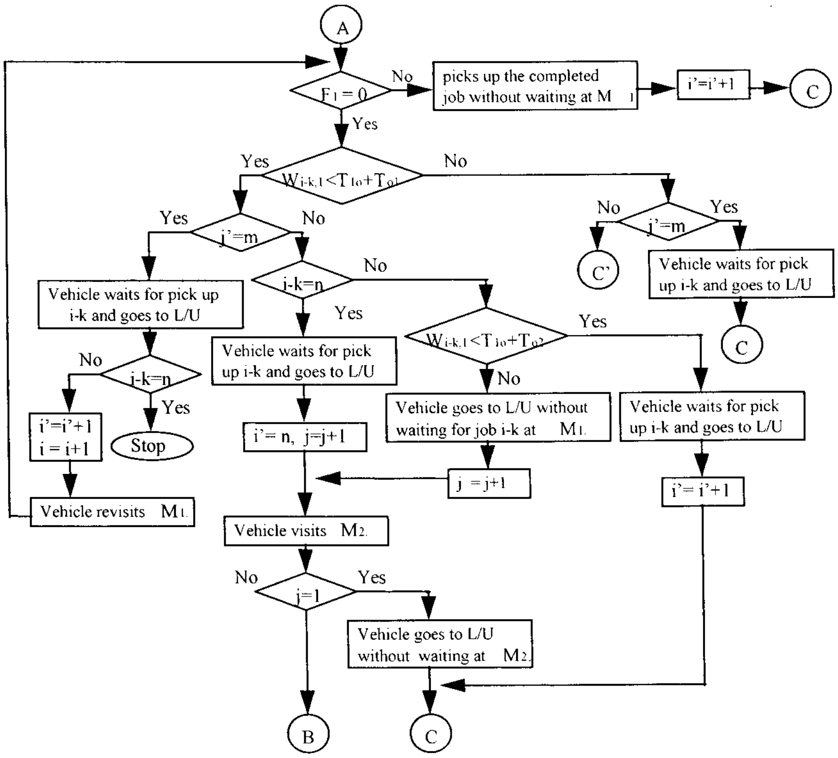
Figure 7 shows the method for assigning vehicle at machine 1 in graph.

3) Part B : The method of assigning vehicle at machine 2.

F2 : the number of jobs completed wait for being picked up at M_2 after having completed processing, $0 \leq F_2 < m$.

Step 1. If $F_2=0$, then go to step 2.
Otherwise vehicle picks up the completed job according to FCFS.
Set j'=j'+1 and go to Part C.

Step 2. If $W_{j-p,2}<T_{20}+T_{02}$, then go to step 3. Otherwise go to step 4.



F1 : the number of jobs waiting for pick up after having completed processing at $M_1(0 \leq F_1 < n)$.

Figure 7. Flow chart for assigning vehicle at machine 1.

Step 3. If i'=n, then vehicle waits for pick up j-p at M_2, goes to I/O. Go to Part C. Otherwise go to step 5.

Step 4. If i'=n, then vehicle waits for pick up at M_2 and goes to I/O. Go to Part C.

Otherwise go to step 2 of Part 2.

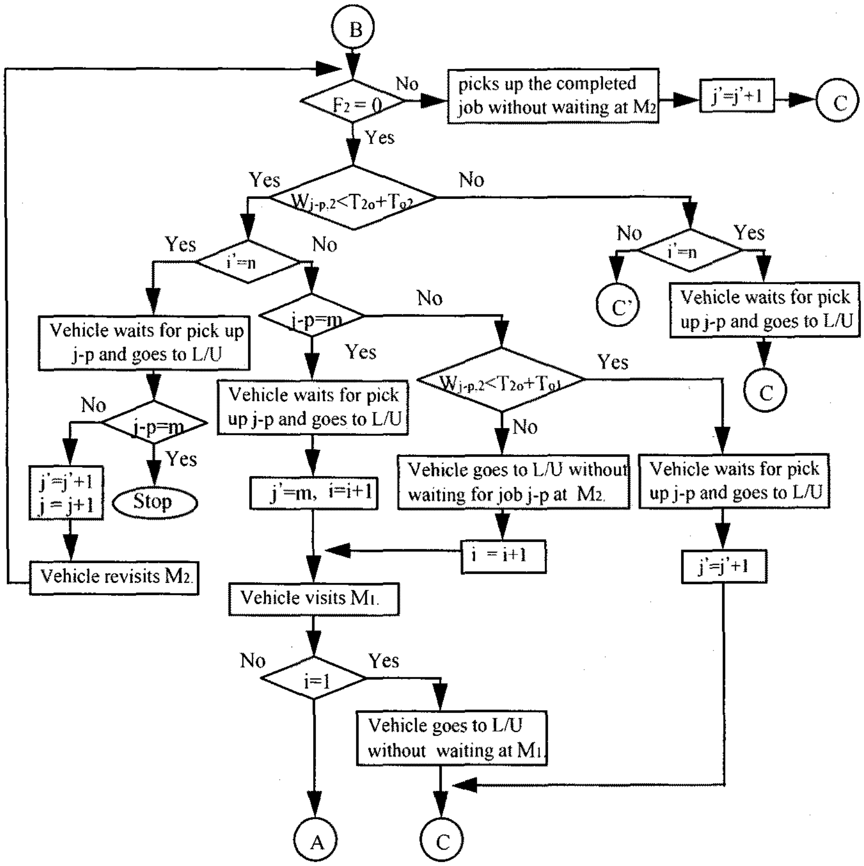
Step 5. If $W_{j-p,2}<T_{20}+T_{01}$, then vehicle waits for pick up j-p and goes to I/O. Set j'=j'+1 and go to Part C.

Otherwise go to step 6.

Step 6. Vehicle goes to I/O without waiting for pick up j-p. Go to Part C.

Figure 8 shows the method for assigning vehicle at machine 2 in graph.

4) Part C : The method of assigning vehicle at I/O station



F_2 : the number of jobs waiting for pick up after having completed processing at M_2 ($0 \le F_2 < m$).

Figure 8. Flow chart for assigning vehicle at machine 2

Step 1. If i'=n and j'=m, then stop.

If i'=n and j'≠m, then set j=j+1 and vehicle visits M_2. Go to Part B.

If i'≠n and j'=m, then set i=i+1 and vehicle visits M_1. Go to Part A. Otherwise go to step 2.

Step 2. If $C_{i-k,1}-D_o<T_{o1}$ and $C_{i-k,1}-D_o<T_{o2}$, then set j=j+1 and vehicle visits M_2. Go to Part A.

If $C_{i-k,1}-D_o<T_{o1}$ and $C_{i-k,1}-D_o\geq T_{o2}$, then set i=i+1, vehicle visits M_1 and go to Part A, or set j=j+1, vehicle visits M_2 and go to Part B arbitrarily.

If $C_{i-k,1}-D_o\geq T_{o1}$ and $C_{i-k,1}-D_o<T_{o2}$, then set i=i+1 and vehicle visits M_1. Go to Part A.

If $C_{i-k,1}-D_o\geq T_{o1}$ and $C_{i-k,1}-D_o\geq T_{o2}$, then go to step 3.

Step 3. If $C_{i-k}<C_{j-p,2}$, then set i=i+1 and vehicle visits M_1. Go to Part A.

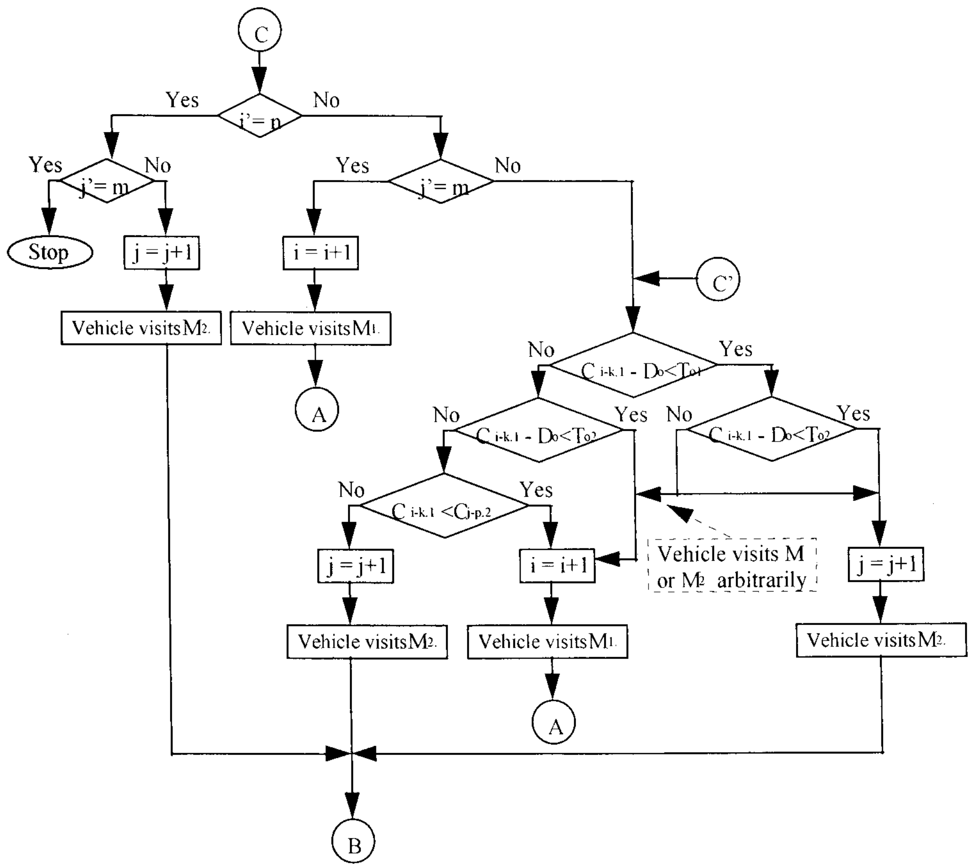


Figure 9. Flow chart for assigning vehicle at I/O

Table 1. Comparative analysis by the proposed heuristics (I)

Jobs	Improving ratio of productivity			Frequency test		
	algorithm #1	algorithm #2	algorithm #3	algorithm #1	algorithm #2	algorithm #3
5	5.9	6.5	7.8	100	100	100
10	10.6	11.7	12.5	100	100	100
20	12.3	13.6	15.4	100	100	100
30	14.8	15.9	18.8	100	100	100
50	15.7	17.7	20.5	100	100	100
70	16.4	18.4	21.2	100	100	100
100	16.8	18.8	20.8	100	100	100

Table 2. Comparative analysis by the proposed heuristics(II)

Jobs	Reduction ratio of waiting jobs at M_1 and M_2						reduction ratio of job waiting time at M_1 and M_2					
	algorithm #1		algorithm #2		algorithm #3		algorithm #1		algorithm #2		algorithm #3	
	M_1	M_2	M_1	M_2	M_1	M_2	M_1	M_2	M_1	M_2	M_1	M_2
5	14.6	10.7	14.8	11.4	15.2	11.8	23.5	8.4	27.3	10.0	28.5	10.4
10	13.4	8.1	13.5	8.5	16.5	8.7	17.8	5.5	20.1	6.1	21.6	6.6
20	7.2	6.0	8.0	6.3	8.6	6.9	12.6	3.0	14.7	3.5	15.3	3.8
30	6.3	4.3	6.4	4.9	6.5	5.0	7.3	2.0	8.0	2.2	8.5	2.3
50	4.0	3.2	4.3	3.5	4.3	3.6	4.7	2.1	5.4	2.4	5.8	2.4
70	3.1	2.4	3.2	2.4	3.2	2.7	2.5	2.0	3.0	2.2	3.1	2.2
100	2.6	2.3	2.6	2.3	2.8	2.7	2.2	1.5	2.4	1.7	2.5	1.7

Otherwise set j=j+1 and vehicle visits M_2. Go to Part B.

Figure 9 shows the method for assigning vehicle at I/O in graph.

5. Evaluation of the proposed algorithms

We proposed three heuristics for scheduling in an automated manufacturing systems. To evaluate the proposed algorithms in this paper, the various problems are implemented by simulation in computer and the results are compared with the random algorithm. Table 1 and 2 present the results of the reduction ratio compare with random algorithm for 7 cases based on randomized completely block design. The problem sizes have been varied anywhere from 3 to 100(5, 10, 20, 30, 50, 70, 100). The proposed heuristic #3 gave better solutions copare with random and algorithm #1, #2. The algorithm 3 has consistently determined a lesser makespan the random algorithm and other two heuristics in figure 10.

The results of frequency test for all size problems are also presented in figure 11. For these, each problem size was simulated 100 times. The proposed algorithm #3 provides better solution than random algorithm in n all problem sizes.

As the coefficient of variation of processing time increase, improving ratio of the productivity also increase from 5.8% to 17.1 % as shown in figure 12 below.

As we can see in figure 13 and 14, reduction ratio of job waiting time and the number of waiting jobs to be processed at machine centers M_1 and M_2 became higher as the number of jobs is few.

Based upon the results obtained it can be concluded that the proposed algorithm has outperformed the random in all of the problems solved with regard to improving productivity by minimizing the makespan and frequency.

The improvement in performance of the proposed algorithms over random procedures is more pronounced with medium and large size problems than small size problems.

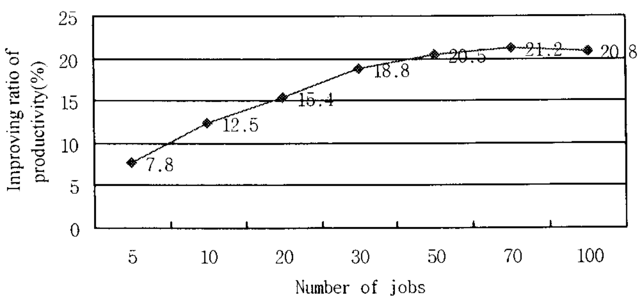


Figure 10. Improving ratio of productivity

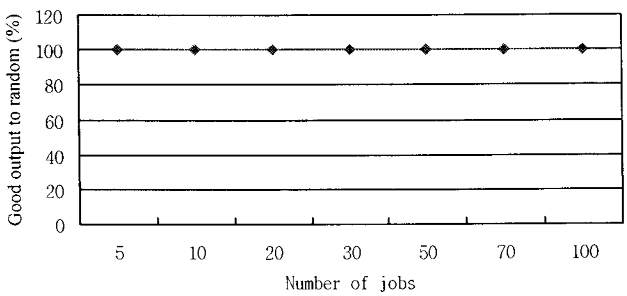


Figure 11. Frequency test of each case

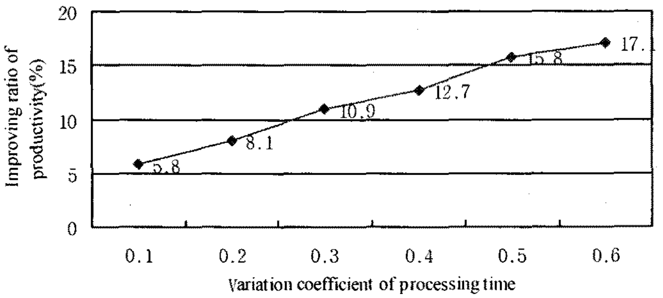


Figure 12. Improving ratio of productivity by variation of processing time

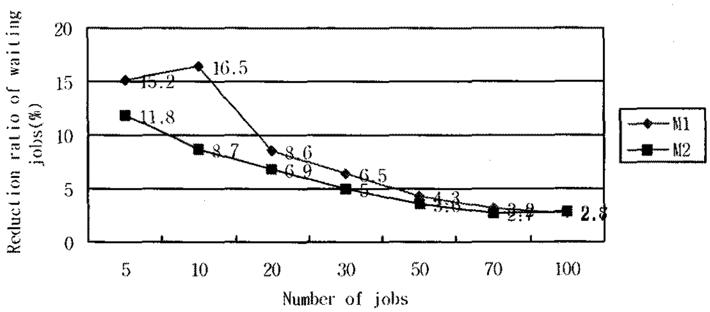


Figure 13. Reduction ratio of waiting jobs at M_1 and M_2

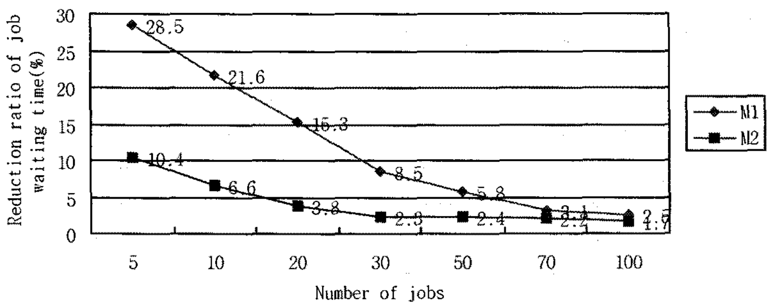


Figure 14. Reduction ratio of job waiting time at M_1 and M_2

6. Conclusions

A new heuristic algorithms have been proposed for solving jobshop scheduling problems with material handling time in automated manufacturing system.

The steps associated with this heuristic are suited for a scheduling procedures as it curtails substantially the enumerations needed to be performed to determine an optimal or near optimal solution for improving productivity by minimizing the makespan. For example, a case has six jobs to be processed. This example has 6! or 720 different schedules, in order to find the optimal solution.

However, if the number of jobs increase, it may be impossible to find the optimal solution within the limited feasible computation time as the enumerations of the given jobs increase geometrically. Therefore, the simulation has been performed to the various cases that have five through one hundred jobs. A complete schedule for this problem has been determined by the proposed heuristics after generating only 800 different schedules.

Thus curtailing substantially on the enumerations are performed. A lesser makespan has been evaluated for the schedule determined by the proposed heuristic algorithm than the random. Furthermore, a comparative analysis based on a randomized complete block design has been conducted to compare the performances of the proposed algorithms and the random procedures. The results obtained show a superior performance by the new algorithm proposed in this paper over the random heuristics on all problem sizes attempted.

In the future, it is necessary to study on job control mehtod under stochastic circumstance and with multiple performace measures.

Acknowledgement

This study was partially supported by research funds from Kangnam University, 2001.

7. References

1. Choi, J.S. and Ro, I.G., "An heuristic algorithm in FMS with an unidirectional transporter," J. Korean Institute of Industrial Engineers 18, 24-31(1988).
2. Choi, J.S. and Ro, I.G., "An scheduling algorithm in FMC with travel time of AGV," J. Korean Academy of Production & Operations Management 15, 65-72 (1994).
3. Choi, J.S. and Ko, N.Y., "Optimized approach for an advanced manufacturing systems with no temporary buffer," Int.

J. of Mechatronics, Vol. 2, 85-290 (1998).

4. Choi, J.S. and Ko, N.Y., "Scheduling procedures in FMS," Int. J. of Information and Communications, Vol. 2, 403-408 (2000)
5. Choi, J.S. and Lee, C.S., "Operation sequence and tool selection in advanced manufacturing systems," Int. J. of Computers in Industry, be appeared (2001).
6. Hoitomt, D., et al., "Scheduling jobs with simple precedence constraints on parallel machines," IEEE contr. Sys. Mag., 687-693(1990).
7. Jeffrey, W. and Rosner, R., "Optimization algorithms - simulated anneling and neural network processing," Astrophysical J., part 1, Vol. 310, 213-221(1986).
8. Jackson, J., "An extention of Johnson's results on jobshop scheduling," Naval Research Logistics Quarterly 3, 201-203 (1956).
9. Johnson, S.M., "Optimal two- and three stage production with setup times included," Naval Research Logistics Quarterly 1, 61-68(1954).
10. Kise, H., Shioyama, T. and Ibaraki, T., "Automated two-machine flow-shop scheduling," IIE Transactions 23, 10-16 (1991).
11. Maggu, P.L. and Dass, G., "On 2×n sequencing problem with transportation time of jobs," Pure and Applied Mathematika Science 12, 1-6(1980).
12. Musser, K.L., Dhingra, J.S. and Blankenship, G.L., "Optimization based job shop scheduling," IEEE Transactions on Automatic Control 38, 808-813(1998).
13. Ow, P.S. and Smith S.F., "Viewing scheduling as an opportunistic problem solving process," Annals Operations Research: Approaches to intelligent decision support, Jeroslow, R. G. Ed., Baltzer scientific publishing co.(1997).
14. Stern, H.I. and Vitner, G., "Scheduling part in a combined production-transportation work cell," J. Operational Research. Society 49, 625-632(1999).
15. Vancheeswaren, R. and Townsend, M.A., "Two-stage heuristic procedure for scheduling job shops," J. Manufacturing Systems 19, 315-325 (1999).