▩ 응용논문

Jobshop환경에서 총처리시간을 최소화하기 위한 AGV의 할당규칙
Dispatching rule of automated guided vehicle to minimize makespan under jobshop condtion

최정상
Jung Sang Choi*
강인선
In Seon Kang**
박찬웅
Chan Woong Park***

Abstract

This research is concerned with jobshop scheduling problem for an advanced manufacturing system like flexible manufacturing which consists of two machine centers and a single automated guided vehicle(AGV). The objective is to develop and evaluate heuristic scheduling procedures that minimize makespan to be included travel time of AGV. A new heuristic algorithm is proposed and illustrates the proposed algorithm. The heuristic algorithm is implemented for various cases by SLAM II. The results show that the proposed algorithm provides better solutions in reduction ratio and frequency than the previous algorithm.

1. Introduction

Recent innovations in manufacturing systems such as the flexible manufacturing systems(FMS) are aimed at the realization of fully automated manufacuring. Recent years have seen the develpment of several automated computer-controlled devices for the purpose of increasing manufacturing productivity. Examples of these devices machining centers, 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.

* Department of Industrial Engineering, Chosun University
** Department of Industrial Engineering. Jeonju University
*** Department of Industrial Management Information System, Kyungwon College

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 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-objective, two-stage with one transporter and unlimited buffer[2],[3],[4],[5].

On the other side, there are relatively few reported research of job shop scheduling problem with material handling consideration. The general job shop scheduling problem is of perennial interest because of its direct revelance to practical manufacturing problems and because of the moderate level progress that has been obtained toward a computationally feasible solution. A multitude of approaches to the problem have been suggested. Among these are expert system methods[10], neural networks methods[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. Problem Formulation

2.1 System description

Figure 1 is a graphical representation of the system of interest in this paper. The figure shows a manufacturing shop consisting of two automated workstations. 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 a bi-directional track and this allows the shuttle to move in either direction.

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.

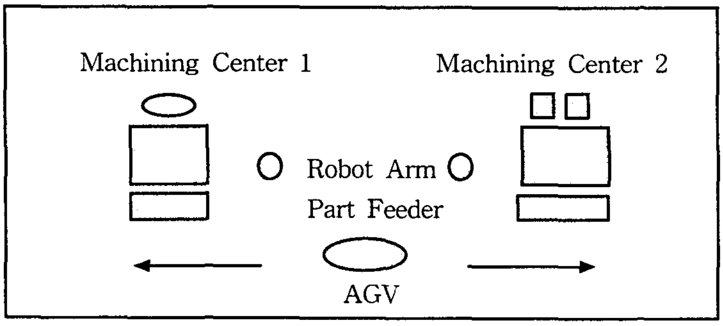


Fig. 1 A Two-Workstation FMS with a Bi-directional Shuttle Cart

Therefore, the objective of this research is to develop a production plan which would minimized the total completion time 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 material transfer between the stations and between the stations and the load/unload point is accomplished using a material handling system. In the paper, we present an algorithm to minimize the production makespan for jobs in a system as described.

3. Problem Formulation

In formulating the problem, the following 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

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

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 { S_{12} --> S_{10} -->S_{21}}

S_2 : set of all jobs that require processing on machine 2 { S_{21} --> S_{20} -->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

$A_{i,1}$: 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.

3.1 Mathematical model

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

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

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

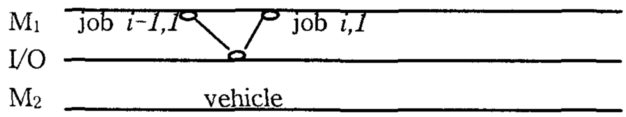


Fig. 2 Transporter move pattern when it revisits M_1 from M_0

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.

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

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

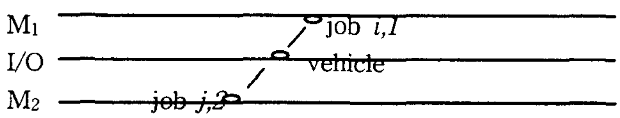


Fig. 3 Transporter moves from M_2 through the I/O point to M_1.

The expression $B_{i,1}$ 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 M1 from M0 before it undertakes any other movement, then

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

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

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

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

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

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

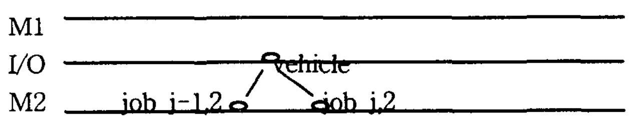


Fig. 4 Transporter move pattern when it revisits M_2 from M_0 before making some other move from M_2.

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 $i = 1$, $W_{i-k,1} = W_{0,1}$, $J_{i-k,1} = J_{0,1}$, and $C_{i-k,1} = C_{0,1}$.

$$\begin{aligned} D_{j,o} &= A_{i,1} + YW_{i-k,1} + T_{1o}, \quad 1 \le k < i \\ &= C_{i-k,1} + J_{i-k,1} + T_{1o}, \quad 1 \le k < i \\ \textit{where } W_{i-k,1} &= Max\left\{ 0,\ C_{i-k,1} - A_{i,1}, \quad 1 \le k < i \right\} \\ J_{i-k,1} &= Max\left\{ 0,\ A_{i,1} - C_{i-k,1}, \quad 1 \le k < i \right\} \\ Y &= \begin{cases} 0, & \textit{if } W_{i-k,1} > T_{1o} + T_{o1}, \quad 1 \le k < i \\ 1, & \textit{otherwise} \end{cases} \end{aligned} \tag{11}$$

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

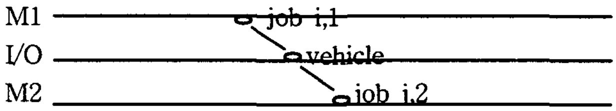


Fig. 5 Transporter move pattern as it travels to M_2 from M_1 with a pickup of load j at M_0

The expression for the start time, $B_{j,2}$, 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 M_o and returns to M_2 before making any other move, then

$$\begin{aligned} B_{j,2} &= Max\left\{ C_{j-1,2},\ A_{j-1,2} + YW_{j-p,2} + T_{2o} + T_{o2}, \quad 1 \le p < j \ne 1 \right\} \\ &= Max\left\{ C_{j-1,2},\ C_{j-p,2} + J_{j-p,2} + T_{2o} + T_{o2}, \quad 1 \le p < j \ne 1 \right\} \end{aligned} \tag{12}$$

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

$$\begin{aligned} B_{j,2} &= Max\left\{ C_{j-1,2},\ A_{i,1} + YW_{i-k,1} + T_{1o} + T_{o2}, \quad 1 \le k < i \right\} \\ &= Max\left\{ C_{j-1,2},\ C_{i-k,1} + J_{i-k,1} + T_{1o} + T_{o2}, \quad 1 \le k < i \right\} \end{aligned} \tag{13}$$

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

$$C_{j,2} = B_{j,2} + P_{j,2} \tag{14}$$

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

$$D_{j-p,2} = C_{j-p,2} + J_{j-p,2}, \quad 1 \le p < j \ne 1 \tag{15}$$

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

$$D_{j-p,2} = A_{j,2} + W_{j-p,2}, \quad 1 \le p < j \ne 1 \tag{16}$$

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).

$$\begin{aligned} C_{\max} &= Max\left\{ A_{n,o},\ A_{m,o} \right\} \\ &= Max\left\{ C_{n,1} + J_{n,1} + T_{1o},\ C_{m,2} + J_{m,2} + T_{2o} \right\} \\ &= Max\left\{ A_{n,1} + W_{n,1} + T_{1o},\ A_{m,2} + W_{m,2} + T_{2o} \right\} \end{aligned} \tag{17}$$

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

$$Min\ Z = C_{\max} \tag{18}$$

4. Dispatching rule of AGV

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.

a) 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, $0 \leq R_1 < n.$

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

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

Step 2. If , vehicle delivers job $i+1$ from M_o to M_1 Set $i=i+1,\ R_1 = R_1 - 1$ and 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 M_1 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.

b) Part A : The method of assigning AGV at M_1.

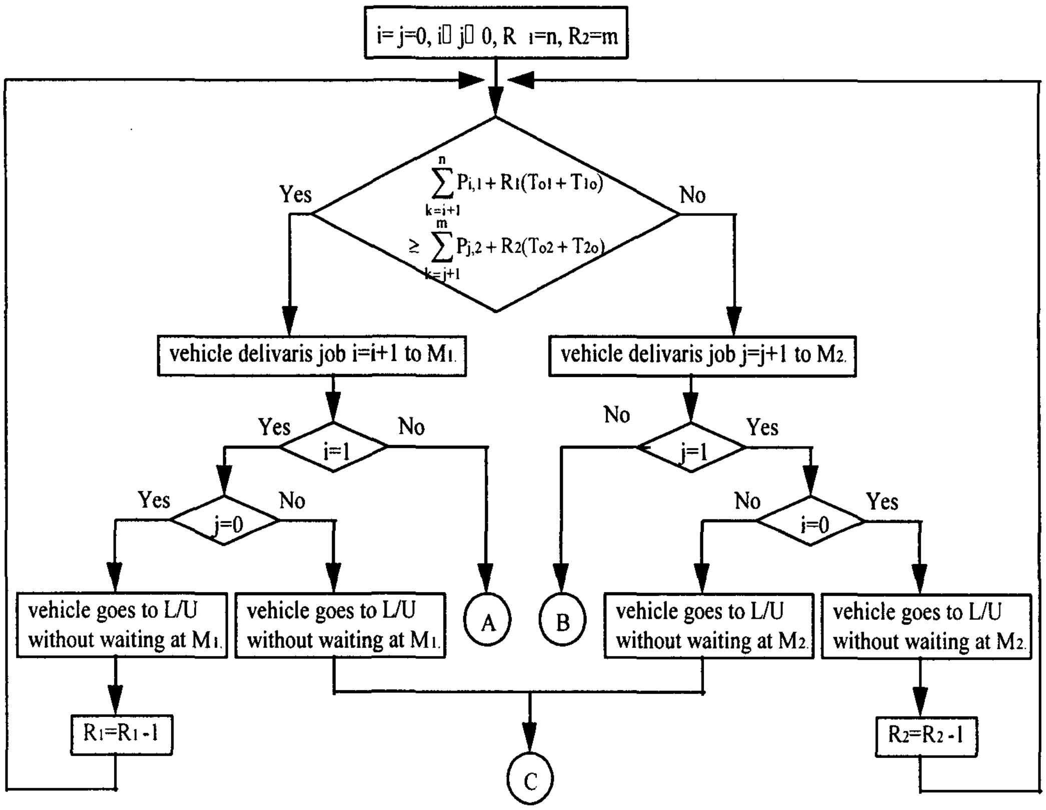
F_1 : the number of jobs completed wait for being picked up at M_1 after having completed processing, $0 \leq F_1 < n.$

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.



i□ : a job that has completed processing at M1 and turned back to loading/unloading station(L/U) (i□ 1,2,3,...,n).
j□ : a job that has completed processing at M2 and turned back to loading/unloading station(L/U).(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. ($0 \leq R_2 \leq m$)

Fig. 6. Flow chart for initialization to assign vehicle.

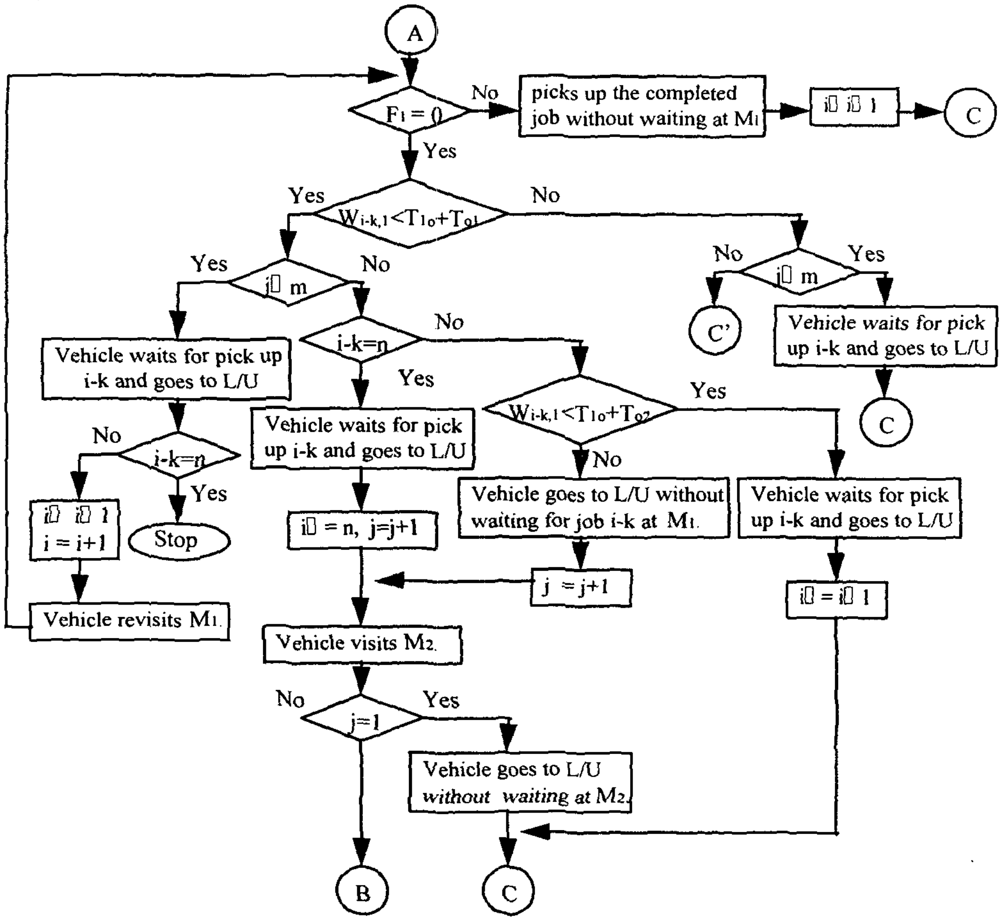
Step 3. If $j'=m$, then vehicle waits for pick up $i-k$ at M_1, goes to I/O. Go to Part C. Otherwise go to step 5.

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.



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

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

c) Part B : The Method of Assigning Vehicle at M_2.

F_2 : the number of jobs completed wait for being picked up at M2 after having completed processing, $0 \le F_2 < m$.

Step 1. If $F_2=0$, then go to step 2.

Otherwise vehicle picks up the completed job according to FCFS.

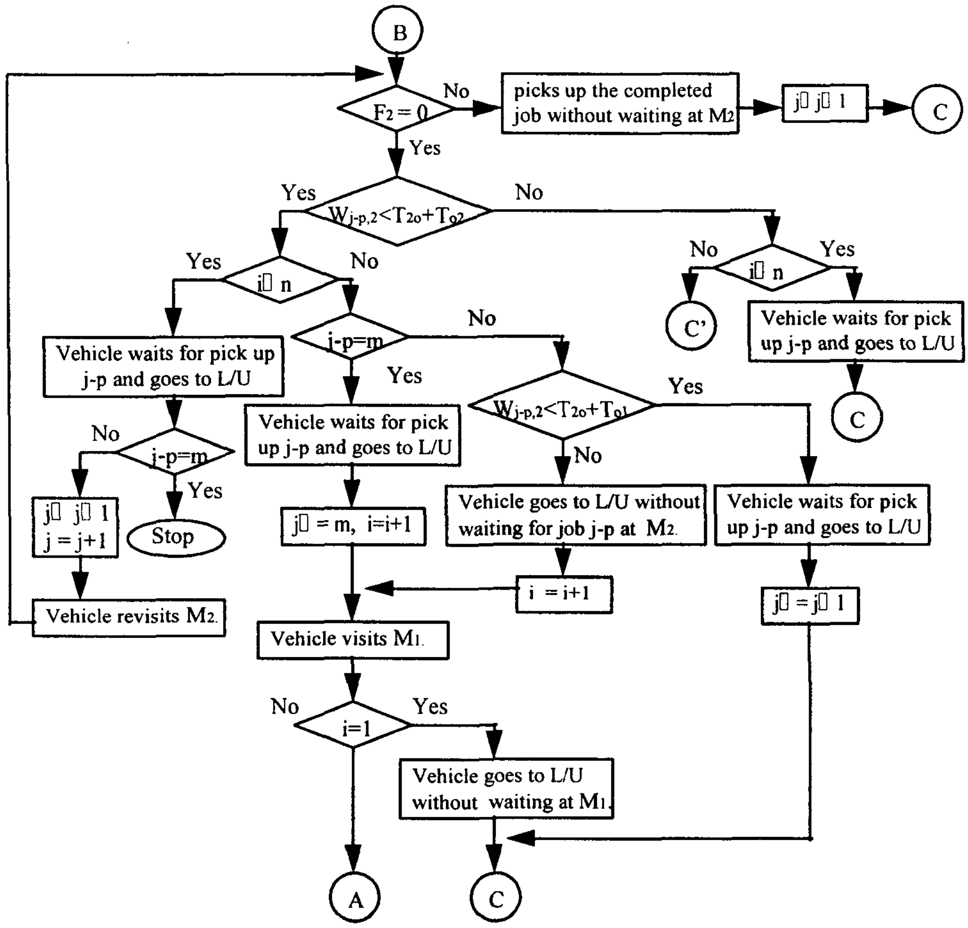
Set $j'=j'+1$ and go to Part C.

Step 2. If $W_{j-p,2}<T_{2o}+T_{o2}$, then go to step 3. Otherwise go to step 4.

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.



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

Fig. 8. Flow chart for assigning vehicle at machine 2

Step 5. If $W_{j-p,2}<T_{2o}+T_{o1}$, 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.

5. Evaluation the developed algorithms

To evaluate the proposed algorithm in this paper, the various problems are implemented by simulation in computer and the results are compared with the random algorithm. Table 1 presents the results of the reduction ratio compare with random algorithm for 8 cases based on randomized completely block design. The problem sizes have been varied anywhere from 3 to 100. The proposed heuristic has consistently determined a lesser makespan the random algorithm.

The results of frequency test for all size problems are also presented in Table 1. For these, each problem size was simulated 100 times. The proposed algorithm provides better solution than random algorithm in n all problem sizes.

Table 1. Reduction of makespan

No. of jobs	Reduction ratio of makespan	Frequency test
3	7.5	100
5	9.7	100
10	11.6	100
20	13.8	100
30	16.4	100
40	19.7	100
50	22.3	100
70	26.4	100
100	29.9	100

Table 2. Results by variation of processing time

Coefficient of variation	Reduction ratio of makespan
0.1	5.8
0.2	8.1
0.3	10.9
0.4	13.6
0.5	17.8
0.6	21.3
0.7	28.2

Like being shown in table 1, as the coefficient of variation of processing time increase, the reduction ratio of makespan also increase. 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 minimizing the makespan and frequency. The improvement in performance of the proposed algorithm over the previous heuristic is more pronounced with medium and large size problems than small size problems.

6. Conclusions

In this paper, a new heuristic algorithm has been proposed for solving jobshop scheduling problems with material handling time in advanced 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 minimizing the makespan. For example, a cases 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, simulation has been performed by considering the various cases that has four through one hundred jobs and three through twenty machining centers.

A complete schedule for this problem has been determined by the proposed heuristic after having generated only 800 different schedules. Thus curtailing substantially on the enumerations needed to be 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 algorithm and the previous heuristic. The results obtained show a superior performance by the new algorithm proposed in this paper over the random heuristic on all problem sizes attempted.

Acknowledgement

This study was partially supported by research funds from Chosun University, 1999.

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, 285-290(1998).

[4] Choi, J.S. and Ko, N.Y., Scheduling procedures in FMS, Int. J. of Information and Communications, Vol2, 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 annealing 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 2n 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. Manufacuring Systems 19, 315-325 (1999).