Journal of the Korean Institute of Industrial Engineers
Vol. 17, No. 1, June, 1991

Scheduling for Mixed-Model Assembly Lines in JIT Production Systems†

In-Kyu Ro* and Joon-Seok Kim*

JIT 생산 시스템에서의 혼합모델 조립라인을 위한 일정계획

노인규* · 김준석*

Abstract

This study is concerned with thd scheduling problem for mixed-model assembly lines in Just-In-Time(JIT) production systems. The most important goal of the scheduling for the mixed-model assembly line in JIT production systems is to keep a constant rate of usage for every part used by the systems.

In this study, we develop two heuristic algorithms able to keep a constant rate of usage for every part used by the systems in the single-level and the multi-level. In the single-level, the new algorithm generates sequence schedule by backward tracking and prevents the destruction of sequence schedule which is the weakest point of Miltenburg's algorithms. The new algorithm gives better results in total variations than the Miltenburg's algorithms. In the multi-level, the new algorithm extends the concept of the single-level algorithm and shows more efficient results in total variations than Miltenburg and Sinnamon's algorithms.

1. Introduction

The number of manufacturing firms worldwise take a step to adopt a type of multi-product, small-lot-sized production. Considering the present business circumstances faced with various difficulties, it is strongly required to establish a production system which can do practically effective and flexible multi-product, small-lot-sized production[1, 6].

JIT production concept can simply be described as the production of one unit in a process to be incorporated just in time into a subsequent process [19]. Monden[8] explained JIT production as to

† 이 논문은 1990년도 교육부지원 해외연구비로 연구되었음.

* Department of Industrial Engineering, Hanyang University

produce necessary units in necessary quantities at the necessary time. The purpose of JIT production systems is to produce a unit in such a way that there is only one unit of work on process and minimum stock of finished goods in inventories. Therefore, unnecessary inventories a 'collection of troubles and bad causes' [17] will be eliminated. Based on this idea, only those goods sold should be produced and replaced[12]. JIT production systems employ a combination of several elements. These elements include smoothing of production providing for process flexibility and versatility, standardization of jobs, and utilization of an ordering and delivery system called kanban[4]. The smoothing of production is the most important condition for production by kanban and for minimizing slack time in regards to manpower, equipment, and work-in-process[13]. If the subsequent process withdraws parts in a fluctuating manner in regards to time or quantity under the production rule of JIT concept and there are many sequenced processes, the variance of the quantities withdrawn by each subsequent process may became large. In order to prevent such large variance in all production lines, an effort must be made to minimize the fluctuation of production in the final assembly line[15].

The JIT production systems often uses mixed-model assembly line for a multi-product, small-lot-sized production which helps satisfy various types of customer demands without holding large inventories. The effective utilization of mixed model assembly line requires that the following problems must be solved[9]:

1) Determination of line cycle times,

2) Determination of the number and sequence of stations on the line,

3) Line balancing,

4) Determination of the sequence scheduling for producing different products on the line.

This study deals with the fourth problem. Determination of the sequence schedule for producing different products on the line depends upon the goals of the company. There are two possible goals [13]:

1) Levelling the load(total assembly time) on each station on the line,

2) Keeping a constant rate of usage of every part used by the line.

Under the JIT production systems, the variations in production quantities or conveyance times at preceding process must be minimized. Also, their work-in-process must be minimized. To do so, the quantity used per hour for each part in the mixed-model assembly line must be kept as constant as possible. Therefore, in this study, the second goal is considered. Although both goals are important and need to be considered for all mixed-model assembly line, the second goal is considered to be more important for JIT production systems.

The problem which determines the sequence schedule of products in multi-level mixed-model assembly line is divided into two cases by part requirements. That is,

1) The Single-Level Problem(products with similar part requirements)

It is assumed that products require approximately the same number and mixing of parts. Thus, a constant rate of part usage can be achieved by considering only the demand rates for the products.

2) The Multi-Level Problem(products with different part requirements)

It is assumed that products require quite different number and mixing of parts. In this case, a constant rate of part usage can be achieved by considering both the demand rates for the products and those for the resulting parts.

Monden[1983] described a Goal Chasing Method which is able to keep a constant consumption rate of each part for the mixed-model assembly line composed of final assembly line and sub-assembly line. Miltenburg[1989] developed a number of heuristic algorithms which are able to keep a constant rate of usage for all parts in the singel-level. Miltengurg's algorithms often generated the destruciton of sequence schedule. Therefore, an additional procedure for rescheduling must be included in Miltenburg's algorithms. Park[1989] developed several algorithms which are able to keep a constant consumption rate for all parts used by the multi-level mixed-model assembly line in the cases of part mixed-usage and no-part-mixed-usage. Park's algorithm could not perfectly prevent the destruction of sequence schedule. Miltenburg and Sinnamon[1989] developed heuristic algorithms which are able to keep a constant rate of usage for all parts in the multi-level mixed-model assembly line. Miltenburg and Sinnamon's algorithms could not effectively reflect the status of future stages.

A four-level mixed-model assembly line is considered in this study. Figure 1 shows the schematic of a four-level mixed-model assembly line. Raw-materials(R. M.) or purchased parts and fabricated into components(COMP) combined into sub-assemblies. Sub-assemblies(SAL) are assembled into products(PRO) on a mixed-model final assembly line.

The objective of this study is to develop the two new heuristic algorithms for the single-level and the multi-level scheduling problems. The one attempts to prevent the destruction of sequence in the single-level, the other will extend the concept of the single-level algorithm to the multi-level scheduling problem.

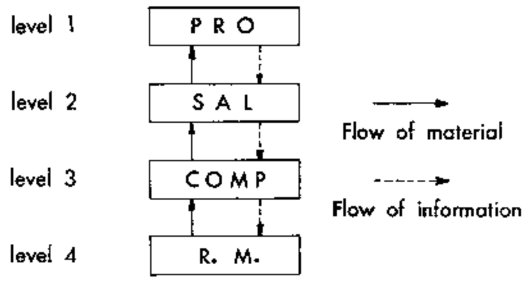

Fig 1. The four-level mixed-model assembly line

2. The Single-Level Problem

2-1. Mathematical Model

The notations used in the single-level are as follows :

k : stage number($k=1, 2, \cdots, D_T$)

d : demand for product i($i=1, 2, \cdots, n$)

D_T : total demand of all product $DT=\sum_{i=1}^{n} di$

r_i : desired proportion of product i. $r_i=d_i/D_T$

x_{ik} : total production of product i over stages 1 to k

s_{ik} = 1 : product i being produced during stage k

0 : otherwise

z_{ik} : desired number of product i($i=1, 2, \cdots, n$). $z_{ik}=(k \cdot d_{di})/D_T$

m_{ik} : the nearest integer point to z_{ik}. $|m_{ik}-z_{ik}| \leq 1/2$

M_k : set of m_{ik}.

$M_k = (m_{1k}\ m_{2k}\ m_{3k}\ \cdots\ m_{nk})$

Z_k : set of z_{ik}.

$Z_k = (z_{1k}\ z_{2k}\ z_{3k}\ \cdots\ z_{nk})$

X_k : set of x_{ik}.

$X_k = (x_{1k}\ x_{2k}\ x_{3k}\ \cdots\ x_{nk})$

k_{mk} : the sum of m_{ik} for product i in stage k

$K_{mk} = \sum_{i=1}^{n} m_{ik}$

q_i : value of m_{ik} at the latest base stage k

In the JIT production systems, only one product can be produced during each stage. Thus,

$\sum_{i=1}^{n} s_{ik} = 1$, for all k

Therefore, X_{ik} and k are as follows :

$x_{ik} = \sum_{j=1}^{k} s_{ij}$, $k = \sum_{i=1}^{n} x_{ik}$

The objective function can be one of the following equations[9].

$$\text{Min} \sum_{k=1}^{D_T} \sum_{i=1}^{n} (x_{ik}/k - r_i)^2 \quad \cdots\cdots (1)$$

$$\text{Min} \sum_{k=1}^{D_T} \sum_{i=1}^{n} (x_{ik} - kr_i)^2 \quad \cdots\cdots (2)$$

$$\text{Min} \sum_{k=1}^{D_T} \sum_{i=1}^{n} | x_{ik}/k - r_i | \quad \cdots\cdots (3)$$

$$\text{Min} \sum_{k=1}^{D_T} \sum_{i=1}^{n} | x_{ik} - kr_i | \quad \cdots\cdots (4)$$

These objective functions seek to minimize the variation of the actual production from the desired production. The objective functions (1) and (3) try to keep the actual proportions of the production mix(x_{ik}/k) close to the desired proportions(r_i) at all times(k). The objective functions (2) and (4) attempt to keep the actual number of units produced(x_{ik}) close to the desired number of units(kr_i) at all times(k). Both of the objective function (2) and (4) are considered to be reasonable and adequate to the single-level scheduling problem. In this study, the objective function (2) is adopted.

The single-level scheduling problem can be formulated as follows :

$$\text{Min} \sum_{k=1}^{D_T} \sum_{i=1}^{n} (x_{ik} - kr_i)^2$$

$$\text{s.t.} \sum_{i=1}^{n} x_{ik} = k \quad (k = 1, 2, \cdots, D_T) \quad \cdots\cdots (5)$$

x_{ik} is nonnegative integer

Constraint (5) means only one product can be produced during each stage.

2-2. Development of Heuristic Algorithm

In the single-level, we can achieve a constant rate of part usage by considering only the demand rates for the products.

The sequence schedule by Miltenburg's algorithm is frequently destroyed because it does not consider the status of future stages. Therefore, it is necessary for another procedure to resolve the destruction of sequence schedule, and hence many addtional calculations are required.

As only one product can be assembled during one stage, k products must be produced during k stages. The sum of the nearest integer point(k_{mk}) in k stage must be the same as production quantity (k) during k stages. If k_{mk} is different from k, m_{ik} should be adjusted.

$k - k_{mk} = 0 \rightarrow$ Adjustment is not needed.
$k - k_{mk} < 0 \rightarrow$ Decrement of m_{ik} is required.
$k - k_{mk} > 0 \rightarrow$ Increment of m_{ik} is required.

Because adjustment of m_{ik} must be performed without considering the status of future stages, the sequence schedule generated by Miltenburg's algorithm is frequently destroyed. In an attempt to resolve these problems backward tracking from the base stage which has the value of $k - k_{mk}$ is

0. The single-level scheduling algorithm is proposed as following.

Heuristic Algorithm for The Single-Level

Step 1. (Initialization)
$k=1$.

Step 2. (Determination of the base stage k)
(1) Find the nearest integer m_{ik} to each z_{ik}. $|m_{ik}-z_{ik}| \leq 1/2$.
(2) Calculate k_{mk}.

Step 3. $k-k_{mk}=0$ $j=0$, Go to Step 5.
$k-k_{mk} \neq 0$ $l=l+1$, Go to Step 4.

Step 4. (Adjustment of m_{ik} by the value of $k-k_{mk}$)
(1) If $k-k_{mk}<0$ then
1) Find m_{ik} with the largest value of $m_{ik}-z_{ik}$ among $m_{ik}>q_i$.
2) Decrease the value of this m_{ik}; $m_{ik}=m_{ik}-1$.
Else if $k-k_{mk}>0$ then
1) Find m_{ik} with the smallest value of $m_{ik}-z_{ik}$.
2) Increase the value of this m_{ik}; $m_{ik}=m_{ik}+1$.
End if
(2) Go To Step 8.

Step 5. (Backward tracking for determining sequence schedule)
$l=0$ Go to Step 7.
$l \neq 0$ $j=j+1$, Go To Step 6.

Step 6. (1) If m_{ik} with $m_{ik}<m_{i(k-j)}$ exists then
1) $m_{i(k-j)}=m_i(k-j)-1$.
2) Find $m_{t(k-j)}$ with the smallest $m_{t(k-j)}-z_{t(k-j)}$. $(i \neq t)$
3) $m_{t(k-j)}=m_{t(k-j)}+1$.
End if
(2) $l=l-1$; $j=j+1$: Go To Step 5.

Step 7. Schedule product i with $m_{i(k-j+1)}>m_{i(k-j)}$ at stage $k-j+1$. $j=j-1$. Repeat Step 7 until $j=0$.

Step 8. (Investigation of stopping condition)
$k<D_T$ $k=k+1$, Go To Step 2.
$k=D_T$ Stop.

2-3. Numerical Example

There are 3 products($n=3$) with demands $D=(6, 6, 1)$ to be assembled on a mixed-model assembly line. Therefore, the vector fo demand ratios is $r=(6/13, 6/13, 1/13)$. Miltenburg's algorithm gives the following partial schedule shown in Table 1. The sequence schedule is destroyed at stages 6 and 8. At both stages 6 and 8, the numeric -3 of the sequence schedule represents the production of -1 unit of product 3. Production completed in the previous stages cannot be canceled. Thus, the sequence schedule is destroyed. Therefore, Miltenburgs algorithm requires another procedure for rescheduling.

The steps for the new algorithm at stage $k=2$ in numerical example are illistrated as follows :

At stage 2 :

Step 2. $Z_2=(12/13, 12/13, 2/13)$, $M_2=(1, 1, 0)$ and $k_{m2}=2$

Step 3. $k-k_{m2}=0$(stage 2 is base stage), $j=0$

Table 1. Sequence schedule by Miltenburg's algorithm

k	Xk	Sequence Schedule	Variation
5	2 2 1	3	.5680
6	3 3 0	1, 2, -3	.3195
7	3 3 1	3	.3195
8	4 4 0	1, 2, -3	.5680

Step 5. $l=1$, $j=j+1=1$

Step 6. $m_{l2}<m_{l1}$ does not exist. $l=l-1=0$ and $j=j+1=2$

Step 5. $l=0$

Step 7. $m_{11}>m_{10}$, product 1 is scheduled at stage 1.

$X_1=(1,0,0)$, $j=j-1=1$

$m_{22}>m_{21}$, product 2 is scheduled at stage 2.

$X_2=(1,1,0)$, $j=j-1=0$

Step 8. $k=2<D_T=13$, $k=3$

Table 2 gives the complete sequence schedule by the new algorithm. In Table 2, the rest of sequence schedule can be completed in the same manner with stage 2. The sequence schedule which is generated by the new algorithm is not destroyed as shown in Table 2. The result shows that the number of calculations of sequence schedule are reduced in comparison with Miltenburg's algorithm because the new algorithm can prevent the destruction of the sequence schedule.

Table 2. Sequence schedule by the new algorithm in the single-level

k	Xk	Seauence Schedule	Variation
1	1 0 0	1	.5088
2	1 1 0	2	.0355
3	2 1 0	1	.5799
4	2 2 0	2	.1420
5	3 2 0	1	.7219
6	3 3 0	2	.3195
7	3 3 1	3	.3195
8	3 4 1	2	.7219
9	4 4 1	1	.1420
10	4 5 1	2	.5799
11	5 5 1	1	.0355
12	5 6 1	2	.5089
13	6 6 1	1	.0000

3. The Multi-Level Problem

3-1. Mathematical Model

The notations used in the multi-level are as follows :

k : stage number($k=1,2,\cdots,DT_1$)

j : level number(1=final-assembly, 2=sub-assembly, 3=component, 4=raw-material)

n_j : number of outputs at level $j(j=1,2,3,4)$

d_{i1} : demand for product $i(i=1,2,\cdots,n_1)$

t_{ijl} : number of units of output i at level j used to produce one unit of product $l(i=1,2,\cdots,n_j;\ j=2,3,4;\ l=1,2,\cdots,n_1)$

d_{ij} : demand for output i at level j

$$d_{ij}=\sum_{h=1}^{n_1} t_{hji}\cdot d_{h1}$$

DT_j : total demand for production at level j

$$DT_j=\sum_{i=1}^{n_1} d_{ij}$$

r_{ij} : proportion of level j production devoted to output i. $r_{ij}=d_{ij}/DT_j$.

x_{i1k} : number of units of product i produced during stages $1,2,\cdots,k$.

x_{ijk} : number of units of output i at level j produced during stages $1,2,\cdots,k$. Set $x_{ij0}=0$

$$x_{ijk}=\sum_{h=1}^{n_1} t_{hji}\cdot x_{h1k}$$

XT_{jk} : total production at level j during stages $1,2,\cdots,k$.

$$XT_{jk}=\sum_{i=1}^{n_j} x_{ijk}$$

z_{i1k} : desired number of product $i(i=1,2,\cdots,n_1)$. $z_{i1k}=(k \cdot d_{i1})/DT_1$

m_{i1k} : the nearest integer point to $z_{i1}k$. $|m_{i1k}-Z_{i1k}| \leq 1/2$

X_k : set of x_{i1k}. $X_k=(x_{11k}\ x_{21k}\ x_{31k}\ \cdots\ x_{n1k})$

M_k : set of m_{i1k}. $M_k=(m_{11k}\ m_{21k}\ m_{31k}\ \cdots\ m_{n1k})$

Z_k : set of z_{i1k}. $Z_k=(z_{11k}\ z_{21k}\ z_{31k}\ \cdots\ z_{n1k})$

N_k : set of feasible sequence schedules at stage k.

k_{mk} : the sum f m_{i1k} for product i in stage k $k_{mk}=\sum_{i=1}^{n_1} m_{i1k}$

V_{ik} : the variation when product i is produced at stage k $V_{ik}=\sum_{j=1}^{4}\beta_{ijk}$

$$\beta_{i1k} = (x_{11(k-1)}-kr_{11})^2+\cdots+(x_{i1(k-1)}-kr_{i1})^2+\cdots+(x_{n1(k-1)}-kr_{n1})^2$$

$$\beta_{ijk} = \sum_{h=1}^{n_j}[(x_{hj(k-1)}+t_{hji})^2 - (XT_{j(k-1)}+\alpha_{ji})r_{hj}]^2$$

for $j \neq i$ where $\alpha_{ji}=\sum_{h=1}^{n_j} t_{hji}$

The objective function can be one of the following equations[10].

$$\text{Min} \sum_{k=1}^{DT_1}\sum_{j=1}^{4}\sum_{i=1}^{n_1}(x_{ijk}-XT_{jk}\cdot r_{ij})^2 \quad \cdots\cdots (6)$$

$$\text{Min} \sum_{k=1}^{DT_1}\sum_{j=1}^{4}\sum_{i=1}^{n_1}|x_{ijk}-XT_{jk}\cdot r_{ij}| \quad \cdots\cdots (7)$$

$$\text{Min} \sum_{k=1}^{DT_1}\sum_{j=1}^{4}\sum_{i=1}^{n_1}(x_{ijk}/XT_{jk}-r_{ij})^2 \quad \cdots\cdots (8)$$

$$\text{Min} \sum_{k=1}^{DT_1}\sum_{j=1}^{4}\sum_{i=1}^{n_1}|x_{ijk}/XT_{jk}-r_{ij}| \quad \cdots\cdots (9)$$

$$\text{Min} \sum_{k=1}^{DT_1}\sum_{j=1}^{4}\sum_{i=1}^{n_1}(x_{ijk}-d_{ij}\cdot k/DT_1)^2 \quad \cdots\cdots (10)$$

The object functions (6) and (7) will keep the actual number of units produced(x_{ijk}) close to the desired number of units($XT_{jk}\cdot r_{ij}$) at all times. The objective functions (8) and (9) ensure that the product mix(x_{ijk}/XT_{jk}) is close to the desired proportions(r_{ij}) at all times. Since the stage length is fixed, we may view k/DT_1 as a measure of the elapsed time, so that the objective function (10) is minimized when the production of each individual output proceeds at a nearly constant rate. The objective functions (6) and (7) are very similar. In this study, the objective function (6) is adopted. To compare the objective functions (6) and (8), it is useful to consider the expressions $x_{ijk}-XT_{jk}\cdot r_{ij}$ and $x_{ijk}/XT_{jk}-r_{ij}$. The former necessitates the approximation of a fraction by an integer and so should be of the order of 1. The latter is of the order of $1/XT_{jk}$. Since XT_{jk} increases as k does, we can see the latter terms will contribute less to the overall deviation in the schedule than the earlier terms. The division of responsibility between levels in the objective function (6) is more clearly defined when it is compared with the objective function (10).

The multi-level scheduling problem can be formulated as follows :

$$\text{Min} \sum_{k=1}^{DT_1}\sum_{j=1}^{4}\sum_{i=1}^{n_1}(x_{ijk}-XT_{jk}\cdot r_{ij})^2$$

s.t. $XT_{1k}=k \quad k=1,2,\cdots,DT_1$ $\cdots\cdots$ (11)

$0 \leq x_{i1k}-x_{i1(k-1)} \leq 1$ $\cdots\cdots$ (12)

$(i=1,2,\cdots,n_1;\ k=1,2,\cdots,DT_1)$

x_{i1k} is nonnegative integer.

Constraint (11) ensures that exactly k products are scheduled during k stages. Constraint (12) ensures that it is not possible to schedule less than zero units, more than one units, or fraction of a unit of any product. That is, for each product i, either one unit scheduled in a given stage or it is not scheduled at all.

3-2. Heuristic Algorithm

In the multi-level, we can achieve a constant rate of part usage by considering both the demand rates for products and those of resulting parts.

If all possible sequences are investigated, we can determine optimal sequence schedule. However, this total enumeration method does not work well for large problems[20]. Therefore, we will investigate other scheduling algorithms. In determining of sequence schedule of products, the condition of present stage will affect the status of future stages. Therefore, we can obtain better results with considering the status of future stages.

In this study, a new algorithm for the multi-level extends the concept of the base stage in the single-level algorithm. At the stage which satisfies the condition of the base stage, the sequence schedule is determined by considering the present stage and the next two stages. And at the stage which does not satisfy the condition of the base stage, the sequence schedule is determined by considering the present stage and the next stage. The multi-level scheduling algorithm is proposed as following.

Heuristic Algorithm for The Multi-Level

Step 1. (Initialization)
    $k=1$.

Step 2. (Determination of the base stage k)
    (1) Find the nearest integer m_{ilk} to each z_{ilk}. $|m_{ilk}-z_{ilk}| \leq 1/2$.
    (2) Calculate k_{mk}.
    (3) If $k_{mk}=k$ then
        Go To Step 3.
      Else
        Go To Step 4.
      End if

Step 3. (Investigation of the existence of the feasible sequence schedule among N_k equivalent to M_k of the base stage k)
    If the feasible sequence schedule equivalent to M_k of the base stage k exists then
      Go To Step 5.
    Else
      Go To Step 4.
    End if

Step 4. (Determination of the sequence schedule for stage k by considering stage k and stage k+1)
    (1) Calculate the variation(V_{ik}) when product i is scheduled in stage k.
    (2) Calculate the variation($V_{j(k+1)}$) when product j is scheduled in stage k+1 under the status of producing product i in stage k.
    (3) Calculate $V_{ik}+V_{j(k+1)}$
    (4) Schedule the product i with the lowest variation($V_{ik}+V_{j(k+1)}$) for stage k.
    (5) $k=k+1$: Go To Step 6.

Step 5. (Determination of the sequence schedule for stage k by considering stage k, stage k+1 and stage k+2)
    (1) Calculate the variation(V_{ik}) when product i is scheduled in stage k.
    (2) Calculate the variation($V_{j(k+1)}$) when product j is scheduled in stage k+1 under the status of producing product i in stage k.
    (3) Calculate the variation($V_{l(k+2)}$ when product l is scheduled in stage k+2 under the status of producing product i in stage k and product j in stage k+1.
    (4) Calculate $V_{ik}+V_{j(k+1)}+V_{l(k+2)}$
    (5) Schedule the product i with the lowest variation
    ($V_{ik}+V_{j(k+1)}+V_{l(k+2)}$) for stage k.
    (6) $k=k+1$.

Step 6. (Investigation of stopping condition)

If $k \leq DT_1$ then
Go To Step 2.
Else
Stop.
End if

3-3. Numerical Example

There are 3 products with demands D=(6, 6, 1) to be assembled on a mixed-model assembly. Table 3 gives the part requirements of each product in each level.

The steps for the new algorithm at stage k=2 in numerical example are illustrated as follows:

At stage 2:

Step 2. $Z_2=(12/13, 12/13, 2/13)$, $M_2=(1, 1, 0)$ and $k_{m2}=2$, $k-k_{m2}=0$

Step 3. $N_2=\{(1, 1, 0), (0, 2, 0), (0, 1, 1)\}$

A feasible sequence schedule equal to M_2 of the base stage 2 exists.

Step 5. $V_{12}=7.229$
$V_{13}=18.962$
$\min(V_{14}, V_{24}, V_{34})=V_{24}=28.918$,

Table 3. The part requirement of each output in each level

		product (j=1)		
		1	2	3
sub-assembly (j=2)	1	1	0	0
	2	1	1	1
	3	0	0	4
component (j=3)	1	1	0	4
	2	2	1	1
	3	1	1	1
	4	0	0	16
raw-material (j=4)	1	1	0	20
	2	2	1	17
	3	2	1	5

$V_{13}+V_{24}=47.880$
$V_{23}=15.595$
$\min(V_{14}, V_{24}, V_{34})=V_{24}=28.478$,
$V_{23}+V_{24}=44.073$
$V_{33}=180.736>44.073$ … fathomed
$\min(V_{12}+V_{13}+V_{24}, V_{12}+V_{23}+V_{24})=$
$V_{12}+V_{13}+V_{24}=51.302$

$V_{22}=9.034$
$V_{13}=15.595$
$\min(V_{14}, V_{24}, V_{34})=V_{24}=28.478$,
$V_{13}+V_{24}=44.073$
$V_{23}=20.326$
$\min(V_{14}, V_{24}, V_{34})=V_{14}=28.478$,
$V_{23}+V_{14}=48.804$
$V_{33}=196.009>44.073$ … fathomed
$\min(V_{22}+V_{13}+V_{24}, V_{22}+V_{23}+V_{14})=$
$V_{22}+V_{13}+V_{24}=53.107$

$V_{32}=225.876>51.302$ … fathomed

$\min(V_{12}+V_{13}+V_{24}, V_{22}+V_{13}+V_{24})=$
$V_{12}+V_{13}+V_{24}=51.302$

Product 1 is scheduled at stage 2. $X_2=(1, 1, 0)$. k=3

Table 4 gives the complete sequence schedule by the new algorithm. In table 4, the rest of sequence schedule can be figured out in the same manner with stage 2.

4. Evaluation of Algorithms

4-1. The Single-Level

In an attempt to evaluate the new algorithm for the single-level, we compare it with the Miltenburg's heuristic algorithm 1 and 2. The unmber of products are given to 3, 4 and 5. The demands for products are randomly changed from 1 to 100.

Table 4. Sequence schedule by the new algorithm in the multi-level

k	Xk	Sequence Schedule	Variation
1	0 1 0	2	2.258
2	1 1 0	1	7.229
3	1 2 0	2	15.595
4	1 3 0	2	28.478
5	2 3 0	1	43.391
6	3 3 0	1	65.065
7	3 3 1	3	65.065
8	4 3 1	1	43.391
9	5 3 1	1	28.478
10	5 4 1	2	15.595
11	5 5 1	2	7.229
12	6 5 2	1	2.258
13	6 6 1	2	0.000

In each case which is divided by the number of products, 20 problems are considered to compare each algorithm. Table 5 gives the results of the comparison. Total variations of each case are the averages of total variations of 20 problems. Not only the new algorithm can prevent the destruction of sequence schedule, but also shows better results in total variations than Miltenburg's algorithms.

4-2. The Multi-Level

In order to evaluate the new algorithm for the multi-level, we compare it with Miltenburg and Sinnamon's heuristic algorithm 1 and 2. The number of products are given to 3, 4 and 5. The demands for products are randomly changed from 1 to 10 and the part requirements for the level 2, 3 and 4 are randomly changed 1 to 5. In each case which is divided by number of products, 20 problems are also considered to compare the new algorithm with Miltenburg and Sinnamon's algorithms. Table 6 gives the results of the comparison. Total variations of each case are the average of total variations of 20 problems. The new algorithm shows much better results in total variations than Miltenburg and Sinnamon's algorithms.

5. Conclusion

This study is concerned with the scheduling problem for mixed-model assembly lines in JIT production systems. The most important goal of the scheduling for the mixed-model assembly line in JIT production systems is to keep a constant rate of usage for every part used by the system.

In this study, we develop two heuristic algorithms which are able to keep a constant rate of usage for every part used by the system in the single-level and the multi-level. In the single-level, the new algorithm generates sequence schedule by backward tracking and prevents the destruction of sequence schedule which is the weakest point of Miltenburg's algorithms. The new algorithm gives better results in total variations than the Mil-

Table 5. Total variation of each algorithm in the single level

number of products	new algorithm	Miltenburg's algorithm 1	Miltenburg's algorithm 2
3	47.57	49.67	47.40
4	85.80	89.55	86.12
5	137.20	144.26	137.27

Table 6. Total variations of each algorithm in the multi level

number of products	new algorithm	Miltenburg and Sinnamon's algorithm 1	Miltenburg and Sinnamon's algorithm 2
3	249.52	329.09	258.28
4	274.15	333.65	280.24
5	280.03	406.79	300.05

tenburg's algorithms. In the multi-level, the new algorithm extends the concept of the single-level algorithm and shows more efficient results in total variations than Miltenburg and Sinnamon's algorithms.

References

[1] Ansari, A. and B. Modaress, "JIT Purchasing as a Quality and Productivity Centre," *Int. J. Prod. Res.*, Vol. 26, No. 1, pp. 19−26, 1988.

[2] Burns, L.D. and C.F. Daganzo, "Assembly Line Job Sequencing Principles," *Int. J. Prod. Res.*, Vol. 25, No. 1, pp. 71−79, 1987.

[3] Dal-El, E.M. and R.F. Cother, "Assembly Line Sequencing for Model Mix," *Int. J. Prod. Res.*, Vol. 13, No. 5, November, pp. 463−477, 1975.

[4] Ebrahimpour, M and R. Shonberger, "The Japanese Just-In-Time/ Total Quality Control Production System : Potential for Developing Countries," *Int. J. Prod. Res.*, Vol. 22, No. 3, pp. 421−430, 1984.

[5] Goodwin, J.S. and J.K. Weeks, "Evaluating Scheduling Policies in a Multi-Level Assembly System," *Int. J. Prod. Res.*, Vol. 24, No. 2, pp. 247−257, 1986.

[6] Ham, I.Y., K. Hitomi and T. Yoshida, *Group Technology : Application to Production Management*, Kluwer-Nijhoff Publishing, a member of the Kluwer Academic Publisher Group, 1985.

[7] Kilbridge, M.D. and L. Wester, "Heuristic Line Balancing : A Case," *Journal of Industrial Engineering*, May-June, pp. 139−149, 1962.

[8] Nellemann, D.O. and L.F. Smith, "Just-In-Time vs. Just-In-Case Production/Inventory Systems : Concept Borrowed Back form Japan," *Pro. and Inv. Mgt.*, 2nd Quarter, pp. 12−21, 1982.

[9] Miltenburg, J., "Level Schedules for Mixed-Model Assembly Lines in Just-In-Time Production Systems," *Management Science*, Vol. 35, No. 2, Fegruary, pp. 192−207, 1989.

[10] Miltenburg. J. and G. Sinnamon, "Scheduling Mixed-Model Multi-Level Just-In-Time Production Systems," *Int. J. Prod. Res.*, Vol. 27, No. 9, pp. 1487−1509, 1989.

[11] Monden, Y., "Smoothed Production Lets Toyata Adapt to Demand Changes and Reduce Inventory," *Industrial Engineering*, August, pp. 42−51, 1981.

[12] Monden, Y., "What Makes The Toyota Production System Really Tick ? " *Industrial Engineering, January, pp. 36−43, 1981.*

[13] Monden, Y., *Toyota Production System*, Institute of Industrial Engineering Press, Norcross, GA, 1983.

[14] Okamura, K. and H. Yamashina, "A Heuristic Algorithm for the Assembly Line Model-Mix Sequencing Problem to Minimize the Risk of Stopping the Conveyor," *Int. J. Prod. Res.*, Vol. 17, No. 3, pp. 233—247, 1979.

[15] Park, C. W., A Study on Sequencing Schedule for Mixed-Model Assembly Line in Just-In-Time Production System, M. S. Thesis, Hanyang University, 1989.

[16] Sarker, B. R. and R. D. Haris, "The Effect of Imbalance in a Just-In-Time Production Systems : A Simulation Study," *Int. J. Prod. Res.*, Vol. 26, No. 1, pp. 1—18, 1988.

[17] Shonberger, R., *Japanese Manufacturing Techniques : Nine Hidden Lessons in Simplicity*, The Free Press, New York, 1982.

[18] Thomopoulos, N. T., "Mixed-Model Line Balancing with Smoothed Station Assignments," *Management Science*, Vol. 16, No. 9, May, pp. 593—603, 1970.

[19] Willis, T. H. and W. C. Suter Jr., "The Five M's of Manufacturing : JIT Conversion Life Cycle," *Pro. and Inv. Mgt.*, 1st Quarter, pp. 53—57, 1989.

[20] Zanakis, S. H. and J. R. Evans, "Heuristic 'Optimization' : Why, When, and How to Use It," *Interfaces*, Vol. 11, No. 5, October, pp. 84—91, 1981.