



UNIVERSITAT POLITÈCNICA
DE CATALUNYA

The Japan (Toyota) industrial organization pattern in mixed-model-assembly-line

Hao Niu and Anna Maria Coves Moreno

IOC-DT-P-2005-14
Juliol 2005

Institut d'Organització i Control
de Sistemes Industrials



THE JAPAN (TOYOTA) INDUSTRIAL ORGANIZATION PATTERN IN MIXED-MODEL-ASSEMBLY-LINE*

Hao Niu and Anna M. Coves Moreno

Institut d' Organització i Control de Sistemes Industrials (IOC)

Universitat Politècnica de Catalunya (UPC)

Av. Diagonal 647 planta 11, 08028 Barcelona, Spain

Abstract

Mixed-model-assembly-lines (MMAL) are widely used in modern industries. There are two principal kinds of industrial organization patterns, which called the *United States Pattern* and the *Japan Patter*. *The Japan (Toyota) Pattern* will be studied in this article and the objective of this pattern is minimizing the stoppage and the idle time. Genetic algorithm is been used for solving the problem and more than 5000 examples have been designed. The results will shown the better parameters for the assembly line and the genetic algorithm.

Keywords: Mixed-model-assembly-lines (MMAL), the *Japan (Toyota) Pattern*, Stoppage time, idle time

1. Introduction

The concept of Autonomation in the Toyota production system gives “autonomy”, or responsibility, to the workers, allowing them to stop the conveyor whenever they fail to complete the operations within their work stations in a mixed-model-assembly-line. It supports JIT by never allowing defective units from the preceding process to flow into and disrupt a subsequent process [Zhao, X. B et al. 2000]. This is quite different from the industrial organization pattern of the USA, which employs utility workers to complete operations left undone by the primary workers [Tsai, L.1995]. Consequently, in the Toyota production system, the conveyor stoppage becomes a crucial criterion in the sequencing problems for these mixed-model assembly lines. In the paper of Zhao, X. B. et al [1996, 1999], they consider the goal of minimizing total conveyor stoppage time, but not the walking times of the workers. The paper of Zhao, X. B. et al [2000] considers both the goal of minimizing a total conveyor stoppage time and the walking times of the workers and a heuristic algorithm is presented, but this algorithm is not efficient enough for large range problems.

Since conveyor stoppages are frequent in many mixed model assembly lines, the goal of minimizing the total conveyor stoppage time will become more and more important. A better planned sequence for mixed models to be assembled on the conveyor can help to

* This research is supposed by the project of investigation of the Ministry of Science and Technology DPI2004 -03472 “Design and Equivalent of Assembly Line with Real Condition”, Spain.

improve the performance of the assembly line. With the consideration of the walking times of the workers, this paper uses the Genetic Algorithm for solving the sequencing problem.

The objective of this paper is to minimize the stoppage time of the assembly line whenever the operators can't finish their work within their stations. The mathematical formula is $Min ST = Min \sum_{k=1}^K \sum_{n=1}^M st_n^k$, where ST is the total stoppage time of the conveyor, K and M are the number of stations and the work-piece models respectively; t_n^k is the operation time of worker k for model n .

According to the classification of assembly line [Niu, H. et al 2005], the *Japan (Toyota) Pattern* assembly line problem discussed in this paper can be classified as follows:

About the products:

In this paper, the type of the products corresponds to the mixed-model-assembly-line; the launching interval discipline is fixed rate; and the position of the products is fixed.

About the assembly line:

In this paper, the layout of time is serial; the line is paced; the type of the station in this paper is left-side closed and right side open (except the last one); the length of the line is deterministic; this pattern is the *Japan (Toyota) Pattern* without consideration of the set-up time.

About the operator:

In this paper, the velocity of the operator is considered and can vary from operator to operator; idle time exists, but there is no utility time; the operation time is deterministic; the operator schedule is later start and the operator only works at one station.

About the objectives:

The objective of the pattern is to minimize the idle time.

The above considerations can be seen in Fig. 1.

2. Conditions of the Assembly Line

The condition of the assembly is: the assembly line moves from left to right, every station has two boundaries, and the two sides of each station are closed.

We also have the following conditions:

The assembly line consists of K stations, and the line moves with velocity of Vc .

There are M kinds of different models which are manufactured from the first station to the last on this assembly line.

There is only one operator at every station.

Whenever one operator can't finish his work within his station, he stops the assembly line and continues to work on this work-piece until he finishes it, and then he restarts the assembly line.

The cycle time or the launch interval time t_c is less than the maximum assembly time of the work-piece of the conveyor.

When one operator finishes his work-piece within his workstation, he walks upstream to the left boundary of his station with the velocity of V_o .

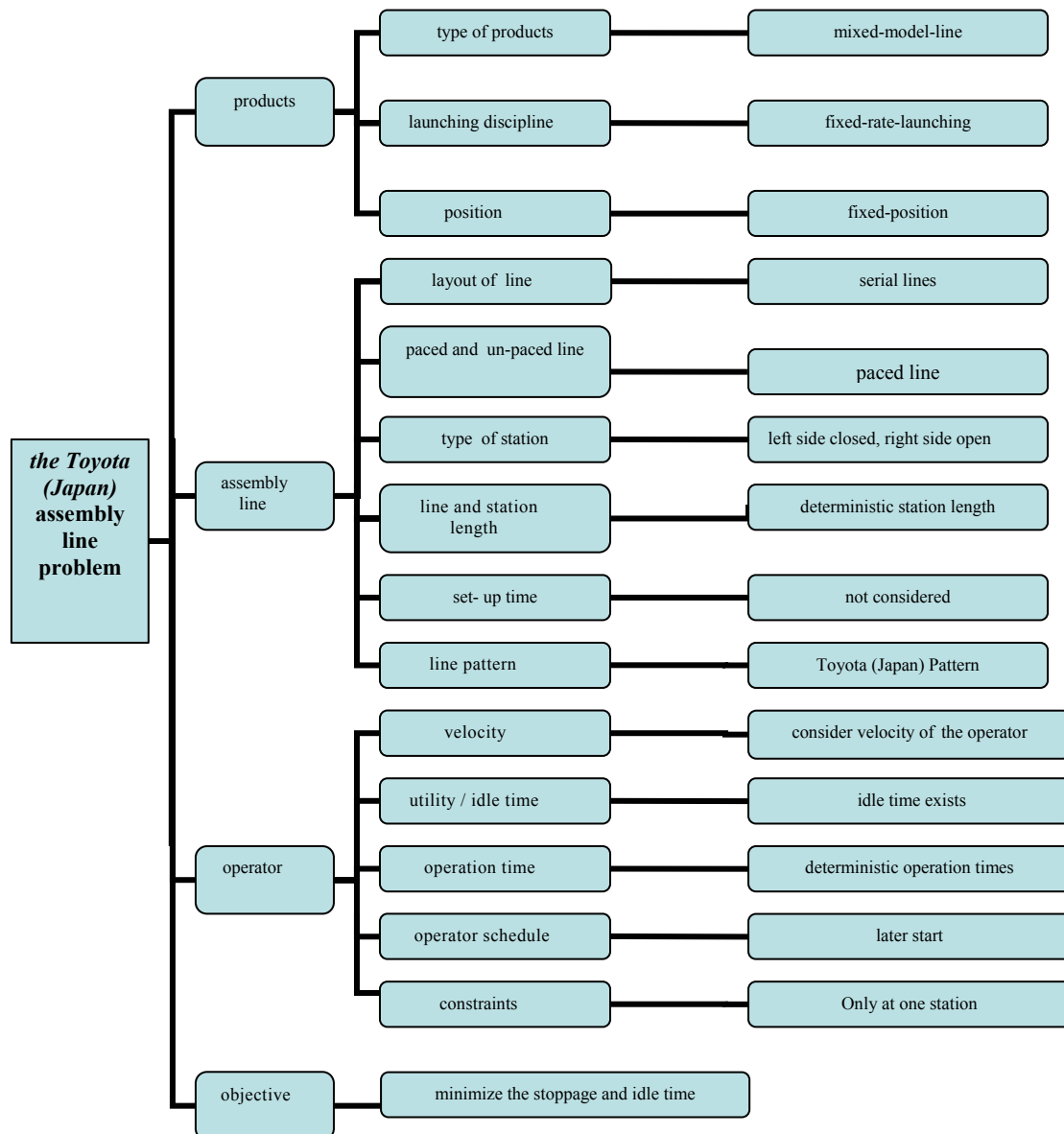


Fig 1 Japan (Toyota) Pattern Assembly Line Problem Classification

3. Preliminaries and Notation

For discussing the Japan (Toyota) Pattern, the following notations are used:

t_m^k	Operation time by worker k for model m ($1, \dots, M$)
$\pi(n)$	The n th unit in sequence $\pi = \{\pi(1), \dots, \pi(n)\}$
L^k	Length of work station k
p_n^k	Starting position from the upstream boundary of work station k for the n th unit in the sequence
f_n^k	Completing position from the upstream boundary of work station k for the n th unit in the sequence
st_n^k	Assembly line stoppage time caused by worker k for the n th units in the sequence.
it_n^k	Idle time of worker k reaching the upstream boundary of work station after completing the operation for the n th unit in the sequence.
$t_{-p_n^k}$	Time the worker k begins the n th unit in the sequence.
$t_{-f_n^k}$	Time the worker k finishes the n th unit in the sequence.
$t_{-fw_n^k}$	Time the operator meets the $(n+1)$ th when he has finished the n th unit or the time when he arrives at the left bounder of the station k even though the $(n+1)$ th hasn't yet arrived.
S	Conveyor stoppage time.
t_{fs}	Time when the conveyor stops.
f_{labs}^k	Right boundary position of station k , based on the left boundary of the first station.
f_{sabs}^n	Position of the n th unit from the left boundary of the first station when the conveyor is stopped.
L_{labs}^k	Right side position of station k from the left boundary of the first station.
L_{sabs}^n	Position of the n th unit from the left boundary of the first station when the conveyor stoppage happens.

$ST(\pi)$ Total conveyor stoppage time of the sequence π .

$IT(\pi)$ Total idle time of sequence of π .

For all the notations above, $K, t_c, Vc, Vo, M, L^k, \pi(n), t_n^k$ are the input variables.

4. Algorithm and Process of the Japan (Toyota) Pattern

Fig. 2 shows an example of the movement of workers on a conveyor. The light lines with arrowheads represent the flows of the work-piece, while the heavy and dotted lines with arrowheads represent the movement of the workers. In the figure, the Vct_c means the flowing distance of every work-piece on the conveyor in one cycle, which is the launch interval distance. In this example, conveyor stoppage occurs with a time interval of length S when the second worker is operating for the 9th work-piece, and the third worker becomes idle upon reaching the upstream boundary of the work station after completing the operations for the 3rd work-piece.

4.1 Calculated Equation (one part of the equations are adopted from Zhao, X. B. et al [2000])

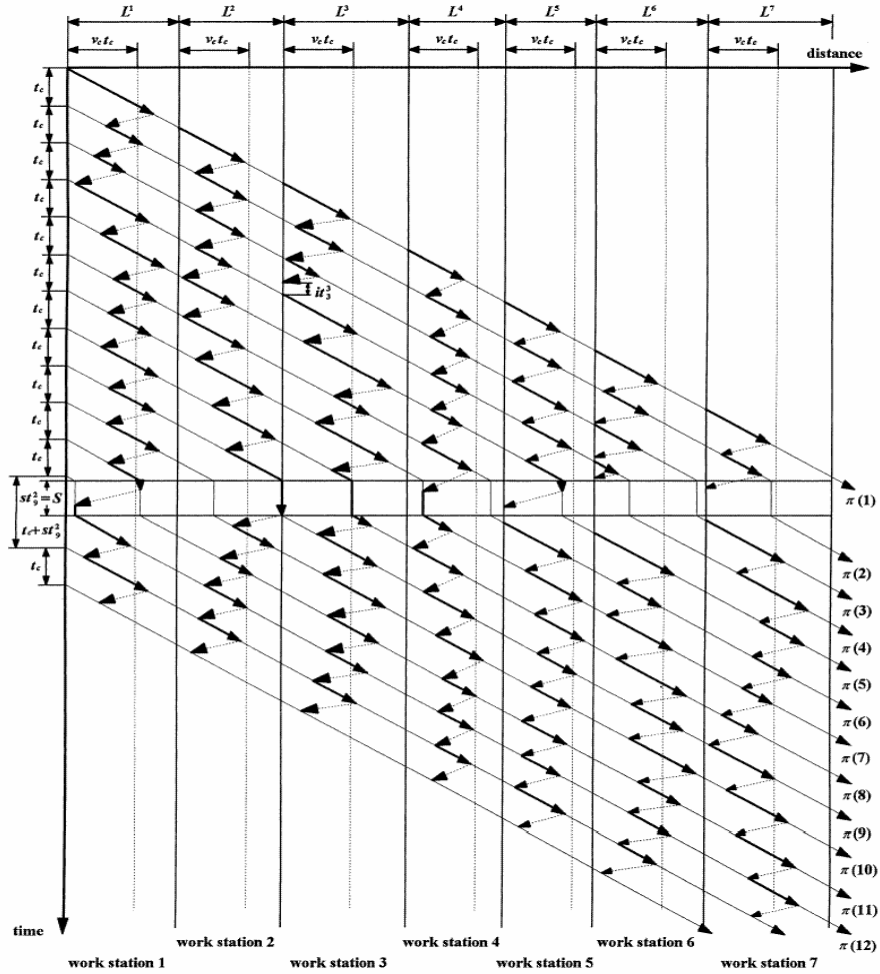


Fig. 2. One Conveyor and Worker Movement Diagram (Adopted from Zhao, X. B. et al [2000])

Before the first conveyor stoppage occurs, the assembly line moves according to the following equations:

$$f_n^k = p_n^k + V_c \cdot t_{\pi(n)}^k \quad (1)$$

Where $t_{\pi(n)}^k$ denotes the operation time by worker k for the n th unit in the sequence π .

$$st_n^k = 0 \quad (2)$$

$$it_n^k = \left(\frac{1}{v_c} + \frac{1}{v_k} \right) * \max(d_k - f_n^k, 0) \quad (3)$$

$$p_{n+1}^k = \max(f_n^k - d_k, 0) \quad (4)$$

$$t_{-}f_n^k = t_{-}p_n^k + t_{\pi(n)}^k \quad (5)$$

On the condition that idle time exists, that is $f_n^k - d_k < 0$, the equations of the assembly line are as follows:

$$t_{-}fw_n^k = t_{-}f_n^k + \frac{f_n^k}{v_k} \quad (6)$$

$$t_{-}p_{n+1}^k = t_{-}fw_n^k + it_n^k \quad (7)$$

On the condition that idle time does not exist, that is $f_n^k - d_k \geq 0$, the equations are as follows:

$$t_{-}fw_n^k = t_{-}f_n^k + \frac{d_k}{v_k} \quad (8)$$

$$t_{-}p_{n+1}^k = t_{-}fw_n^k \quad (9)$$

For the equations above, d_k is the walking distance of the worker k from a completing position to the next starting position when neither idle time nor conveyor stoppage occurs.

The cycle time is t_c , and the conveyor speed V_c , so the distance between the two work-pieces is $t_c \cdot V_c$. If the worker's upstream speed is V_o , the time required to access the approaching work-piece is $\frac{d_k}{v_k}$. Similarly, the time for a work-piece moving downstream

to meet the operator is $\frac{t_c \cdot v_c - d_k}{v_c}$. Therefore we have $\frac{d_k}{v_k} = \frac{t_c \cdot v_c - d_k}{v_c}$ or $d_k = \frac{v_c \cdot v_k}{v_c + v_k} t_c$.

4.2 The station at where the worker stops the assembly line

Suppose that the first conveyor stoppage is caused by worker l when he is operating on the n th unit, that is $p_n^l + v_c t_{\pi(n)}^l > L^l$. When the whole conveyor is stopped, the unit flow and the worker movement diagram is influenced. It is assumed that the conveyor stoppage lasts for a time interval of length S . For $k=l$, we have the following equations:

$$f_n^k = L^k \quad (10)$$

$$st_n^k = \frac{1}{v_c} (p_n^k + v_c \cdot t_{\pi(n)}^k - L^k) = S \quad (11)$$

$$it_n^k = 0 \quad (12)$$

$$p_{n+1}^k = L^k - d_k \quad (13)$$

$$t_{-f_n^k} = t_{fs} + S \quad (14)$$

$$t_{-fw_n^k} = t_{-f_n^k} + \frac{d_k}{v_k} \quad (15)$$

$$t_{-p_{n+1}^k} = t_{-fw_n^k} \quad (16)$$

4.3 The Stations which are Not Stopped

For $K \neq l$, the starting position, the completing position and the idle time of worker k are determined accordingly for the following cases of worker k at the moment of the conveyor stoppage.

4.3.1 The Operator is in Walking State

In the following two cases the worker is walking upstream to the left boundary of his station after finishing the work-piece.

Case 1:

Worker k is in walking state after finishing the operations for the n_l th unit and the (n_l+1) th unit hasn't yet entered the work station. In our example there are two cases of this happening. Case 1 corresponds to the state at work station 7 in Fig. 2.

- Case 1a:

Worker k has to wait for the (n_l+1) th unit at the left boundary of his station. The equations are:

$$it_{n_l}^k = \left(\frac{1}{v_c} + \frac{1}{v_k} \right) (d_k - f_{n_l}^k) + S \quad (17)$$

$$p_{n_l+1}^k = 0 \quad (18)$$

$$t_{-}fw_{n_l}^k = \frac{f_{n_l}^k}{v_k} + t_{-}f_{n_l}^k \quad (19)$$

$$t_{-}p_{n_l+1}^k = t_{-}fw_{n_l}^k + it_{n_l}^k \quad (20)$$

• Case 1b:

Worker k does not need to wait for the (n_l+1) th unit at the left boundary of his station. The equations are:

$$it_{n_l}^k = 0 \quad (21)$$

$$p_{n_l+1}^k = \frac{1}{v_k + v_c} \left\{ v_c \cdot v_k \cdot (t_{-}f_{n_l}^k - t_{fs} - S) + v_c \cdot (L_{labs}^{k-1} + f_{n_l}^k) + v_k \cdot L_{sabs}^{n_l+1} \right\} - L_{labs}^{k-1} \quad (22)$$

$$t_{-}fw_{n_l}^k = \frac{1}{v_k + v_c} \left\{ v_k t_{-}f_{n_l}^k + v_c \cdot (t_{fs} + S) + L_{labs}^{k-1} - L_{sabs}^{n_l+1} + f_{n_l}^k \right\} \quad (23)$$

$$t_{-}p_{n_l+1}^k = t_{-}fw_{n_l}^k \quad (24)$$

The equation (22) and (23) can be deduced from the following methods and diagrams:

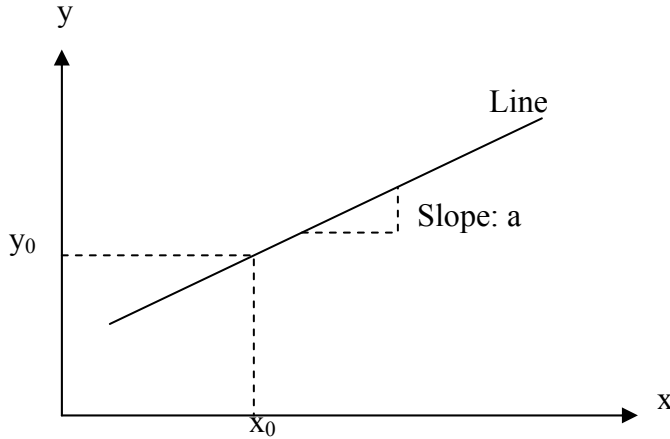


Fig 3 Explanation One of the Deduction of the Equation of Case1b.

The equation of the line above is:

$$y - y_0 = a \cdot (x - x_0) \quad (25)$$

In Fig 4, line A:

$$\begin{aligned}
x &= t, \\
y &= y, \\
x_0 &= t_{fs} + S, \\
y_0 &= L_{sabs}^{n+1}, \\
a &= Vc.
\end{aligned}$$

So line A:

$$y - L_{sabs}^{n+1} = v_c \{t - (t_{fs} + S)\} \quad (26)$$

In Line B,

$$\begin{aligned}
x &= t, \\
y &= y, \\
x_0 &= t - f_n^k, \\
y_0 &= L_{labs}^{k-1} + f_n^k, \\
a &= -Vo.
\end{aligned}$$

So line B:

$$y - (L_{labs}^{k-1} + f_n^k) = -v_k \cdot (t - t - f_n^k). \quad (27)$$

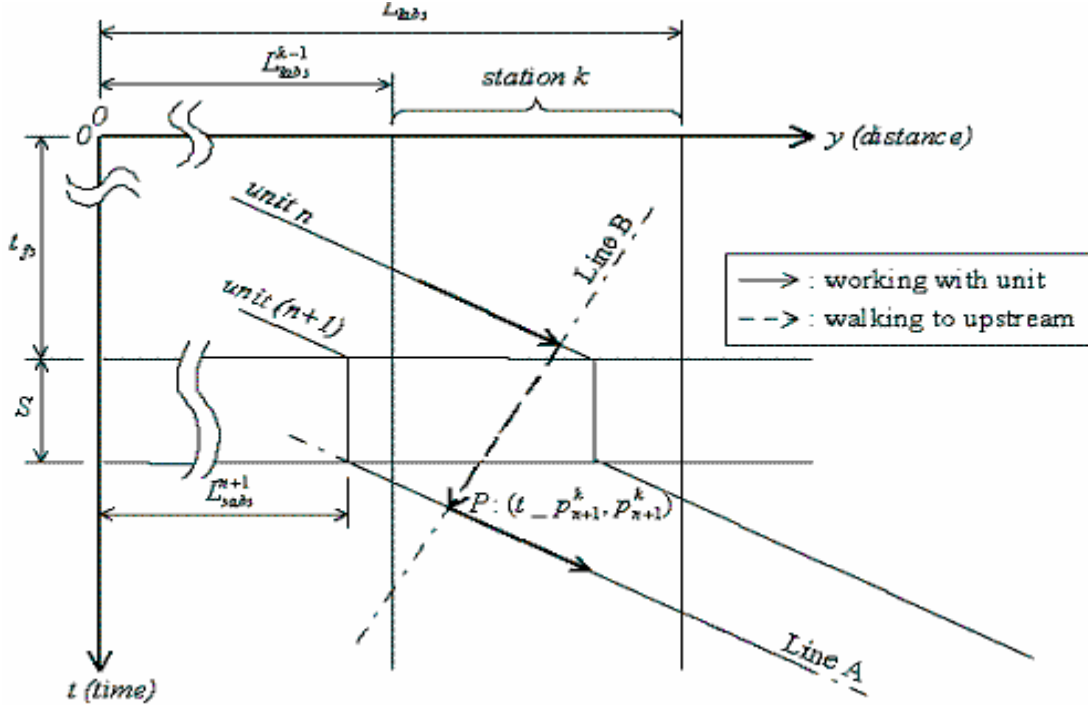


Fig 4. Explanation Two

In the Fig.4, the coordinate of point P is:

$$(t - P_{n+1}^k, P_{n+1}^k) = (t - f_{w_n}^k, P_{n+1}^k) = (t, y - L_{labs}^{k-1}).$$

According to equation 27, there is

$$y = v_c \{ t - (t_{fs} + S) \} + L_{sabs}^{n+1} \quad (28)$$

Take (28) to (27), there is

$$v_c \{ t - (t_{fs} + S) \} + L_{sabs}^{n+1} - (L_{labs}^{k-1} + f_n^k) = -v_k \cdot (t - t_{fn}^k)$$

So

$$t = \frac{1}{v_k + v_c} \{ v_k t_{fn}^k + v_c \cdot (t_{fs} + S) + L_{labs}^{k-1} - L_{sabs}^{n+1} + f_n^k \}$$

$$y = \frac{1}{v_k + v_c} \{ v_c \cdot v_k (t_{fn}^k - t_{fs} - S) + v_c \cdot (L_{labs}^{k-1} + f_n^k) + v_k \cdot L_{sabs}^{n+1} \}$$

Therefore the coordinate of point P is found by the following equations and also the equations of (22) and (23).

$$\left\{ \begin{array}{l} p_{n+1}^k = \frac{1}{v_k + v_c} \cdot \{ v_c \cdot v_k \cdot (t_{fn}^k - t_{fs} - S) + v_c \cdot (L_{labs}^{k-1} + f_n^k) + v_k \cdot L_{sabs}^{n+1} \} - L_{labs}^{k-1} \end{array} \right. \quad (29)$$

$$\left\{ \begin{array}{l} t_{fn}^k = t - p_{n+1}^k = \frac{1}{v_k + v_c} \cdot \{ v_k t_{fn}^k + v_c \cdot (t_{fs} + S) + L_{labs}^{k-1} - L_{sabs}^{n+1} + f_n^k \} \end{array} \right. \quad (30)$$

Case 2:

Worker k is in walking state after finishing the operations for the n_2 th unit with the remaining walking time to be $\tilde{t}_{\pi(n_2)}^k$, and the (n_2+1) th unit has entered the work station. Also there are two different cases. Case 2 corresponds to the state at work station 4 in Fig. 2.

- Case 2a:

Worker k begins to operate on the (n_2+1) th unit before the conveyor stoppage time ends.

The equations are:

$$it_{n_2}^k = 0 \quad (31)$$

$$p_{n_2+1}^k = f_{n_2}^k - d_k - \frac{v_c \cdot v_k}{v_c + v_k} \cdot \tilde{t}_{\pi(n_2)}^k \quad (32)$$

$$t_{\pi(n_2+1)}^k \leftarrow t_{\pi(n_2+1)}^k - (S - \tilde{t}_{\pi(n_2)}^k) \quad (33)$$

$$t_{-}f_{n_2}^k = t_{-}f_{n_2}^k + \frac{f_{n_2}^k - p_{n_2+1}^k}{v_k} \quad (34)$$

$$t_{-}p_{n_2+1}^k = t_{fs} + S \quad (35)$$

- Case 2b:

Worker k begins to operate on the (n_2+1) th unit after the conveyor stoppage time ends.

The equations are:

$$it_{n_2}^k = 0 \quad (36)$$

$$p_{n_2+1}^k = \frac{1}{v_k + v_c} \cdot \left\{ v_c \cdot v_k (t_{-}f_{n_2}^k - t_{fs} - S) + v_c \cdot (L_{labs}^{k-1} + f_{n_2}^k) + v_k \cdot L_{sabs}^{n_2+1} \right\} - L_{labs}^{k-1} \quad (37)$$

$$t_{-}f_{n_2}^k = \frac{1}{v_k + v_c} \cdot \left\{ v_k t_{-}f_{n_2}^k + v_c \cdot (t_{fs} + S) + L_{labs}^{k-1} - L_{sabs}^{n_2+1} + f_{n_2}^k \right\} \quad (38)$$

$$t_{-}p_{n_2+1}^k = t_{-}f_{n_2}^k \quad (39)$$

4.3.2 The Worker is in Operating State

Case 3:

Worker k is in operating state for the n_3 rd unit with the remaining operating time to be $\tilde{t}_{\pi(n_3)}^k \geq S$. This case occurs at work station 3 in Fig. 2.

The equations are:

$$p_{n_3}^k = p_{n_3}^k + v_c \cdot (t_{\pi(n_3)}^k - \hat{t}_{\pi(n_3)}^k) \quad (40)$$

$$t_{\pi(n_3)}^k = \hat{t}_{\pi(n_3)}^k - S \quad (41)$$

$$t_{-}p_{n_3}^k = t_{fs} + S \quad (42)$$

Case 4:

Worker k is in operating state for the n_4 th unit with the remaining operating time to be $\tilde{t}_{\pi(n_4)}^k < S$, and the (n_4+1) th unit hasn't yet entered the work station. There are case 4a and case 4b. Case 4 occurs at work station 5 in Fig. 2.

The equations are:

$$f_{n_4}^k = p_{n_4}^k + v_c \cdot (t_{\pi(n_4)}^k - \hat{t}_{\pi(n_4)}^k) \quad (43)$$

$$st_{n_4}^k = 0 \quad (44)$$

$$it_{n_4}^k = \left(\frac{1}{v_c} + \frac{1}{v_k} \right) \max \left\{ d_k - f_{n_4}^k + \frac{v_c \cdot v_k}{v_c + v_k} \cdot (S - \hat{t}_{\pi(n_4)}^k), 0 \right\} \quad (45)$$

$$p_{n_4+1}^k = \max \left\{ f_{n_4}^k - \frac{v_c \cdot v_k}{v_c + v_k} \cdot (S - \hat{t}_{\pi(n_4)}^k) - d_k, 0 \right\} \quad (46)$$

$$t_{-}f_{n_4}^k = t_{f_s} + \hat{t}_{\pi(n_4)}^k \quad (47)$$

- Case 4a:

Worker k has to wait for the (n_4+1) th unit at the left boundary, that is: $p_{n_4+1}^k = 0$.

The equations are:

$$t_{-}fw_{n_4}^k = t_{-}f_{n_4}^k + \frac{f_{n_4}^k}{v_k} \quad (48)$$

$$t_{-}p_{n_4+1}^k = t_{-}fw_{n_4}^k + it_{n_4}^k \quad (49)$$

- Case 4b:

Worker k can begin to operate on the (n_4+1) th unit before arriving at the left boundary, that is: $p_{n_4+1}^k > 0$.

The equations are:

$$t_{-}fw_{n_4}^k = t_{-}f_{n_4}^k + \frac{f_{n_4}^k - p_{n_4+1}^k}{v_k} \quad (50)$$

$$t_{-}p_{n_4+1}^k = t_{-}fw_{n_4}^k + it_{n_4}^k \quad (51)$$

Case 5:

Worker k is in operating state for the n_5 th unit with the remaining operating time to be $\tilde{t}_{\pi(n_5)}^k < S$, and the (n_5+1) th unit has entered the work station. There are two cases, 5a and 5b. Case 5 occurs at work station 1 in Fig. 2.

The equations are:

$$f_{n_5}^k = p_{n_5}^k + v_c \cdot (t_{\pi(n_5)}^k - \hat{t}_{\pi(n_5)}^k) \quad (52)$$

$$st_{n_5}^k = 0 \quad (53)$$

$$it_{n_5}^k = 0 \quad (54)$$

$$p_{n_5+1}^k = f_{n_5}^k - v_c \cdot t_c + \frac{v_c \cdot v_k}{v_c + v_k} \cdot \max\{\theta_{n_5}^k, 0\} \quad (55)$$

Where

$$t_{\pi(n_5+1)}^k = t_{\pi(n_5+1)}^k - \max\{-\theta_{n_5}^k, 0\}$$

$$t_{-f_{n_5}}^k = t_{fs} + \hat{t}_{\pi(n_5)}^k$$

$$\theta_{n_5}^k = \frac{v_c}{v_k} \cdot t_c - (S - \hat{t}_{\pi(n_5)}^k)$$

- Case 5a:

Worker k can begin to operate on the (n_5+1) th unit before the conveyor stoppage ends, that is $p_{n_5+1}^k = f_{n_5}^k - v_c t_c$.

The equations are:

$$t_{-fw_{n_5}}^k = t_{-f_{n_5}}^k + \frac{v_c t_c}{v_k} \quad (56)$$

$$t_{-p_{n_5}}^k = t_{fs} + S \quad (57)$$

- Case 5b:

Worker k begins to operate on the (n_5+1) th unit after the conveyor stoppage ends, that is $p_{n_5+1}^k > f_{n_5}^k - v_c t_c$.

The equations are:

$$t_{-fw_{n_5}}^k = t_{-f_{n_5}}^k + \frac{f_{n_5}^k - p_{n_5+1}^k}{v_k} \quad (58)$$

$$t_{-p_{n_5+1}}^k = t_{-fw_{n_5}}^k \quad (59)$$

4.3.3 The Worker is in Idle State

Case 6:

Worker k is in idle state upon reaching the upstream boundary of the work station after completing the operations for the n_6 th unit. This case occurs at work station 6 in Fig.2. The equations are:

$$it_{n_6}^k = it_{n_6}^k + S \quad (70)$$

$$t_{-P_{n_6+1}^k} = t_{-f_{n_6}^k} + it_{n_6}^k \quad (71)$$

4.4 Discussion and Some Special Situations

After the first conveyor stoppage ends, the equations which correspond to the relevant case will be used. If a second conveyor stoppage occurs, then the cases and the equations above are repeated, and so on.

This paper explains six different situations for when conveyor stoppage happens. Although other cases exist theoretically, they rarely actually take place. For example, a worker has just finished his operation, the conveyor is stopped, and when he walks upstream to the left boundary of his station, the stoppage time ends; another special case is when a worker is in walking state, and the stoppage ends during the remaining walking time.

Another special situation is when more than one worker causes a conveyor stoppage at the same time. This case seldom happens and will not be discussed in this paper.

4.5 About the Program

The program of this paper also includes two parts; the first is the calculation part, which calculates the six cases of the conveyor. On the basis of the results of the first part, the second part uses the genetic algorithm to obtain the optimal selection. These two parts were carried out in Visual Basic 6.0 on a Pentium 200 MHz computer. The diagram of this two-part program is shown in Fig. 5:

4.5.1 Calculation part

The diagram in Fig. 6 shows the processes of the calculation.

The processes of the calculation are as follows:

- a). Input the data of K, t_c, v_c, v_k, L^k, M and t_n^k .
- b). Input the data of $\pi(n)$.
- c). Initiation.
- d). Calculate f_{labs}^k .
- e). Calculate d_k .
- f). Begin to calculate the stoppage time of every station.
- g). Calculate every station when the conveyor stoppage happens.

- h). If there isn't conveyor stoppage, then move on to the process m.
- i). Find the number of station, number and time duration of the recent conveyor stoppage.
- j). Calculate the station which causes the conveyor stoppage, that is the station which $k=l$.
- k). Calculate f_{sabs}^n .
- l). Calculate the stations which don't cause the conveyor stoppage, stations which $k \neq l$.
- m). Calculate $ST(\pi)$ and $IT(\pi)$.

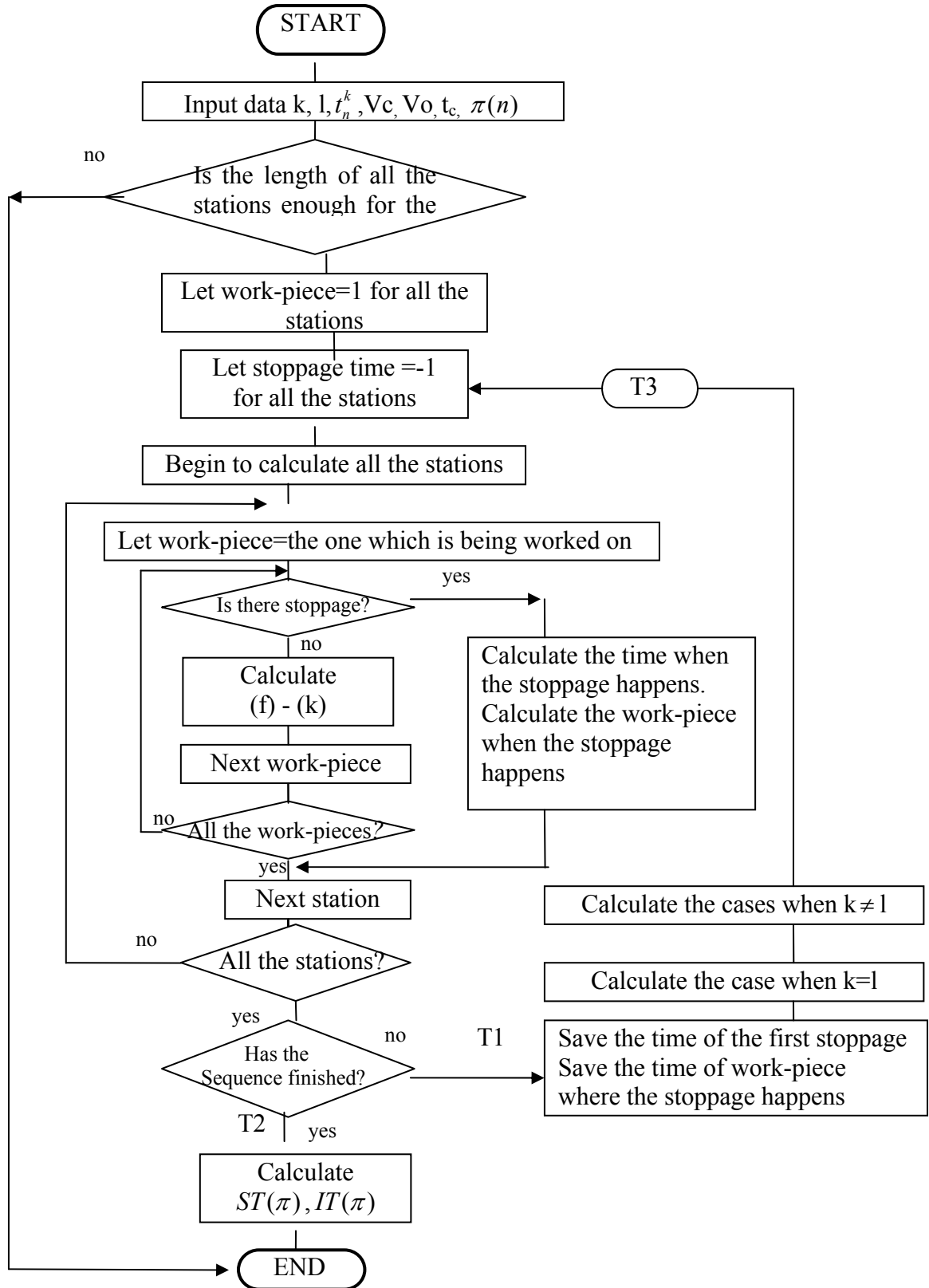
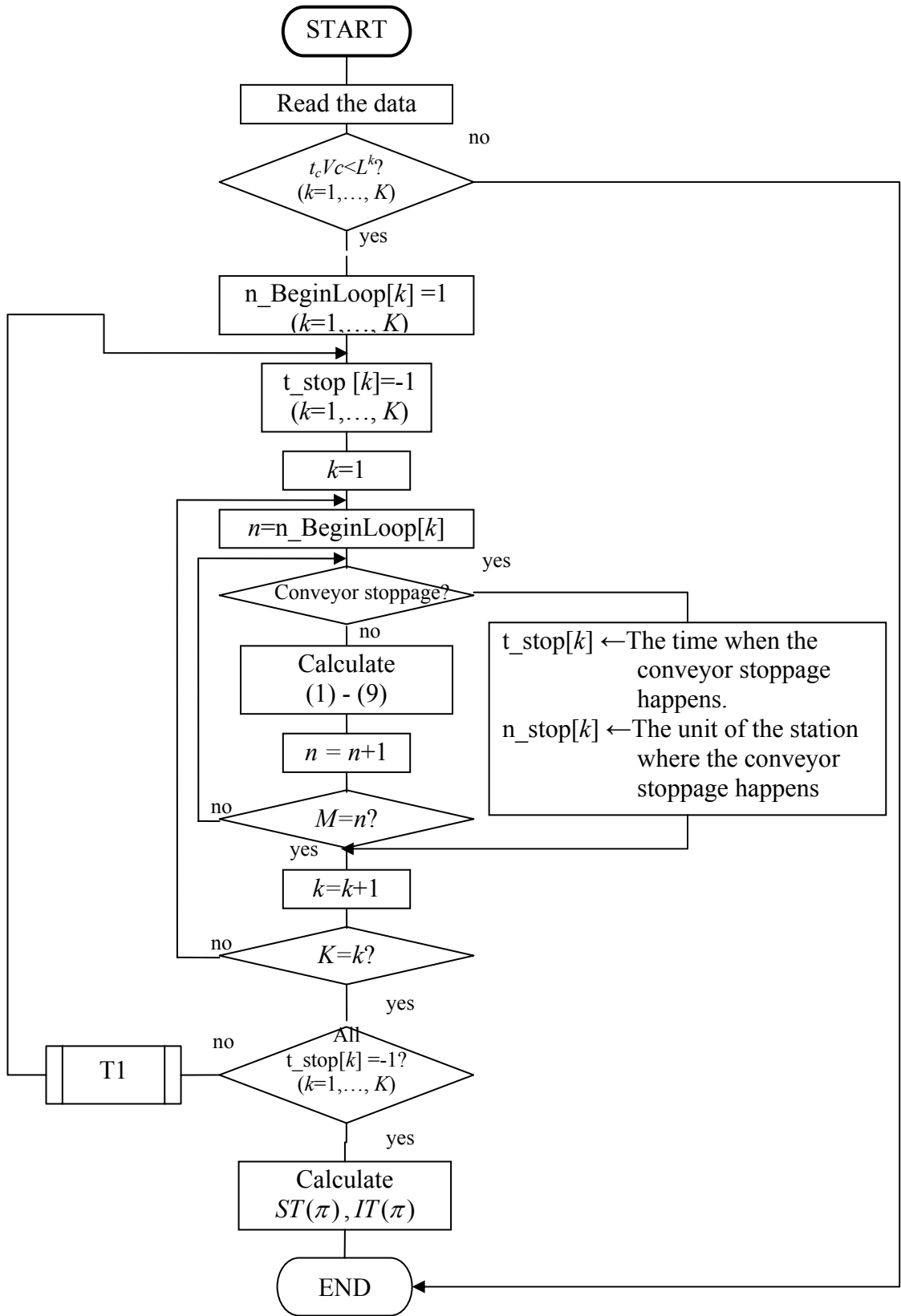


Fig.5 Diagram of the Process of the Calculation and GA



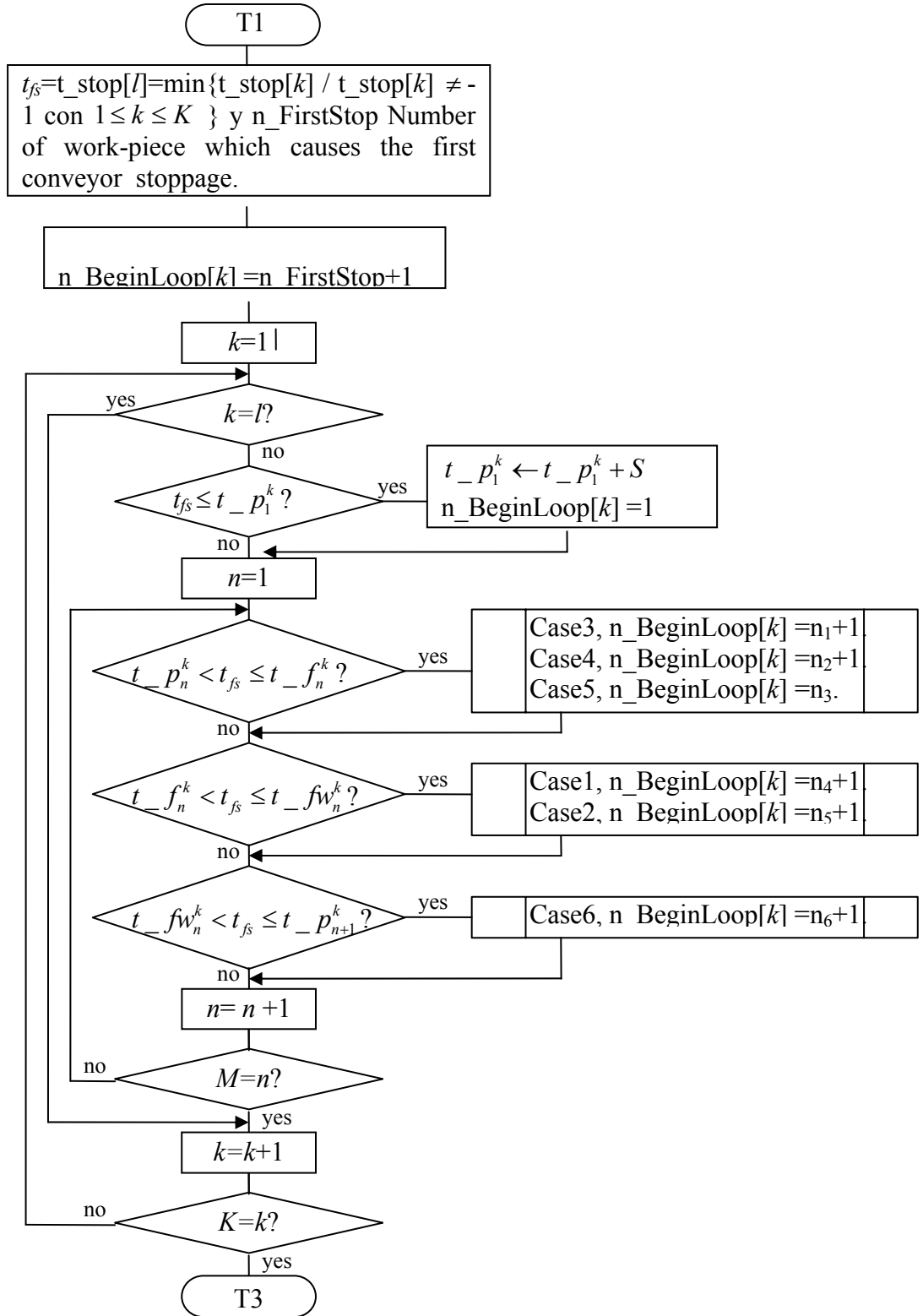


Fig.6 Diagram of the Process of L of the Calculation Part

4.5.2 Genetic Algorithm Part

The method which used in this paper for the selection process is the roulette wheel method; crossover utilizes order crossover method and mutation uses the traditional method.

5. Design experiments

The objective of this paper is to find the best sequence for the work-pieces with the least amount of idle time and stoppage time of the conveyor. Based on the different parameters of the assembly line—number of units, number of models, number of stations, the minimal part set, and the different parameters of the genetic algorithm are the population size, maximum generation, crossover ratio and mutation ratio; they will be changed randomly from small, medium to large quantity. Table 1 show the parameters utilized in the design experiment

Name	Description	Range	Value
Unit	Total number of the work-piece needed to fabricate	1)Medium 2)Large	1) 20 2) 50
Model	Total kinds of the work-piece needed to fabricate	1)Medium 2)Large	1) 6 2) 12
Station	Total number of the station in the conveyor	1)Medium 2)Large	1) 5 2) 15
Minimal part set	A smallest possible set of product type quantities, to be called the multiplicities, in which the numbers of assembled products of the various types are in the desired ratios.	1)A/B/C.../N A=B=C...N 2)A/B/C.../N A=50% B=C...N	For example: 20 work-pieces 5 stations: 1)3/3/3/3/5 2)10/2/2/2/2
Population size	Number of individuals in each generation of the GA	1)Small 2)Medium 3)Large	1) 25 2) 50 3) 70
Maximum generation	Maximum number of generation	1)Small 2)Medium 3)Large	1) 30 2) 75 3) 100
Crossover ratio	Fraction of selected pairs undergo crossover	1)Small 2)Medium 3)Large	1) 20 2) 45 3) 80
Mutation ratio	Percentage of genes in the population which are replaced with random values each generation.	1)Small 2)Medium 3)Large	1) 10 2) 40 3) 60

Table 1 Parameters Utilized in the Design Experiment

From the table above it can shown that there are 20 parameters in this experiment, 6 of the assembly line, 12 of the genetic algorithm and 2 cases of the minimal part set. While one of them is fixed, other parameters change, so the numbers of experiments that will be done are $2^4 \cdot 3^4 = 16 \cdot 81 = 1296$. The combinations of the parameters are: Table 2.

Unit-model-station	Number of every model	Number of every model
20-6-5	A) 3/3/3/3/5	B) 10/2/2/2/2
20-6-15	A) 3/3/3/3/5	B) 10/2/2/2/2
20-12-5	A) 2/2/2/2/2/2/2/1/1/1/1	B) 9/1/1/1/1/1/1/1/1/1/1
20-12-15	A) 2/2/2/2/2/2/2/1/1/1/1	B) 9/1/1/1/1/1/1/1/1/1/1
50-6-5	A) 8/8/8/8/9/9	B) 25/5/5/5/5/5
50-6-15	A) 8/8/8/8/9/9	B) 25/5/5/5/5/5
50-12-5	A) 4/4/4/4/4/4/4/4/5/5	B) 25/3/3/3/2/2/2/2/2/2/2
50-12-15	A) 4/4/4/4/4/4/4/4/5/5	B) 25/3/3/3/2/2/2/2/2/2/2

Table 2 Combination of the Parameters in the Design Experiment

One of the results of the design experiments is shown in Table 3; the other results will be shown in Appendix. (From Table A1—A14).

- Table 3 20-6-5 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P		Small				
	*	Medium	V.G		V.G	V.G
		Large		V.G		
G		Small		V.G		G
		Medium				
	*	Large	V.G		V.G	G
C	*	Small	G		G	G
		Medium		V.G		G
		Large	G		G	
M	*	Small	G	G	V.G	G
		Medium				G
		Large	G	G		

From the table above, when the parameters of the assembly line are all in medium size and the minimal part set is case A, the optimal parameters of the genetic algorithm are medium size in population size, large size in maximum generation, small size in crossover ratio and in medium size mutation ratio respectively.

- Table 2 20-6-5 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P		Small				
		Medium				
	*	Large	V.G	V.G	V.G	V.G
G	*	Small	V.G	G	V.G	V.G
		Medium				
		Large		G		
C		Small			G	G
	*	Medium	G	V.G		G
		Large	G		G	G
M		Small	G	G	G	
	*	Medium	G		G	V.G
		Large		G	G	

From the table above, when the parameters of the assembly line are all in medium size and the minimal part set is case B, the optimal parameters of the genetic algorithm are

large size population size, small size maximum generation, medium size crossover ratio, and medium size mutation ratio respectively.

From the table above, when the parameters of the assembly line are all in large size, and the minimal part set is case B, the optimal parameters of the genetic algorithm are all in medium size.

6. Conclusions

In this paper, we have looked at *the Japan (Toyota) Pattern*, which is if the task cannot be finished within the work stations of the assembly line; the conveyor is stopped and doesn't move again until the unfinished task is completed. Here the objective is to minimize stoppage and idle time.

The program is also based on the genetic algorithm, which works well and the calculation time for getting results is less than two minutes.

As in paper 5, with different combinations of the parameters of the assembly line and genetic algorithm, more than 5,000 experiments have been done, and we have seen the results above. That is that with the *Japan (Toyota) Pattern*, with different scales of assembly line, and with the objective of minimizing the stoppage time and idle time, the optimal parameters of the genetic algorithm are summarized as follows (where unit means number of the work-piece, see Table 5-4):

For the parameters of the assembly line:

- if the unit is medium range (in this paper there are 20 units), the optimal parameters of the genetic algorithm are: population size, small; maximum generation, small; crossover ratio, small and mutation ratio, medium, respectively.
- if the unit is large range (in this paper there are 50 units), the optimal parameters of the genetic algorithm are: population size, small; maximum generation, small; crossover ratio, small and mutation ratio, medium, respectively.
- if the model is medium range (in this paper there are 6 models), the optimal parameters of the genetic algorithm are in small sizes.
- if the model is large range (in this paper there are 12 models), the optimal parameters of the genetic algorithm are: population size, small; maximum generation, small; crossover ratio, small and mutation ratio, medium, respectively.
- if the station is medium range (in this paper there are 5 stations), the optimal parameters of the genetic algorithm are in small sizes.
- if the station is large range (in this paper there are 15 units), the optimal parameters of the genetic algorithm are: population size, small; maximum generation, small; crossover ratio, small and mutation ratio, medium, respectively.

- if the minimal part sets is in case A (in this paper means that all the models of the work-pieces have the same number), the optimal parameters of the genetic algorithm are all in small sizes.
- if the minimal part sets is in case B (in this paper means that one kind of the model has 50% of all the number of the work-pieces, and the other models have the same number), the optimal parameters of the genetic algorithm are: population size, small; maximum generation, small; crossover ratio, small and mutation ratio, medium, respectively.

From the resume above, it can be seen that in *the Japan (Toyota) pattern*, for the different parameters of the assembly line, most of the optimal parameters of the genetic algorithm are in small sizes while cases of large sizes are very few.

References

1. Aigbedo, H. and Monden, Y. (1996) A Simulation Analysis for Two-Level Sequencing for Just-In-Time (JIT) Mixed-Model Assembly Lines. *International Journal of Production Research*, 34, 3107—3124.
2. Bautista, J. and Jordi Pereira (2003) Ant Algorithms for Assembly Line Balancing. *Lecture Notes in Computer Science*. ISSN: 0302-9743, p65.
3. Baybar, I. (1986) A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem. *Management Science*, 32, 909--932.
4. Dar-EL, E. M. and Cothor, R. F. (1975) Assembly Line Sequencing for Model Mix. *International Journal of Production Research* 13, 463-477.
5. Dar-EL, E. M. and Cucuy, S. (1977) Optimal Mixed-Model Sequencing for Balanced Assembly Lines. *OMEGA* 5, 333-342.
6. Goldberg, D. E. (1989) Genetic Algorithm in Search, Optimization and Machine Learning.
7. Inman, R. R and Bulfin, R. L. (1991) Sequencing JIT Mixed-Model Assembly Lines. *Management Science* 37, 901-904.
8. Inman, R. R and Bulfin, R. L. (1992) Quick and Dirty Sequencing for Mixed-Model Multi-Level JIT Systems. *International Journal of Production Research* 30, 2011-2018.
9. Niu, H. and Coves, A. M. M. (2004) Scheduling and Sequencing Problem of Mix-Model-Assembly-Line". *IE&EM'2003*, Shanghai, R. P. China.
10. Okamura, K and Yamashina, H. (1979) A Heuristic Algorithm for the Assembly Line Model-Mix Sequencing Problem to Minimize the Risk of Stopping the Conveyor. *International Journal of Production Research* 17, 233-247.

11. Pleschberger, T. E. and Hitomi, K. (1993) Flexible Final-Assembly Sequencing Method for a JIT Manufacturing Environment. *International Journal of Production Research* 31, 1189-1199.
12. Scholl, A. and S. Voß, S. (1996): Simple Assembly Line Balancing - Heuristic Approaches. *Journal of Heuristics* 2, 217-244.
13. Scholl, A. (1999) Balancing and Sequencing of Assembly Lines. *Second Edition. Darmstadt, Germany*
14. Scholl, A. and Klein, R. (1999) Balancing Assembly Lines Effectively – A Computational Comparison. *European Journal of Operational Research* 114, 50-58.
- Steiner, G and Yeomans, S. (1996) Optimal Level Schedules In Mixed-Model, Multi-Level JIT Assembly Systems with Pegging. *Europe Journal of Operational Research* 95, 38-52.
15. Tsai, L. (1995) Mixed Model Sequencing to Minimize Utility Work and the Risk of Conveyor Stoppage. *Management Science* 41, 485—495.
16. Yano, C. A. and Bolat (1989) Survey: Development, and Application of Algorithm for Sequencing Paced Assembly Lines. *Journal of Manufacturing and Operations Management* 2, 172-198.
17. Zhao, X.B. and Ohno, K. (1994) A Sequencing Problem for a Mixed-Model Assembly Line in a JIT Production System . *Computers Industrial Engineering. Vol. 27, 71—74.*
18. Zhao, X. B. and Ohno, K. (1996) Algorithms for Sequencing Mixed Models on An Assembly Line in a JIT Production System. *Computers Industrial Engineering. Vol. 31, No.1 47—56.*
19. Zhao, X. B. and Zhao, Z. (1999) Algorithm for Toyota ´s Goal of sequencing mixed Models on An Assembly Line with Multiple Workstations. *Journal of the Operation research society* 50, 704—710.
20. Zhao, X. B. and Ohno, K. (2000) Properties of A Sequencing Problem for A Mixed Model Assembly Line with Conveyor Stoppages. *Europe Journal of Operation Research* 124, 560—570.

Appendix

Here, the results of the design experiment will be shown form Table A1—A14.

- Table A1 20-6-15 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	V.G	V.G		G
		Medium				G
		Large			V.G	
G	*	Small		V.G	V.G	
		Medium	G			G

		Large	G			G
C		Small				
		Medium				V.G
	*	Large	V.G	V.G	V.G	
M	*	Small	V.G			V.G
		Medium		V.G	V.G	
		Large				

From the table above, when the parameters of the assembly line are medium size units, medium size models and large size stations respectively, and the minimal part set is case A, the optimal parameters of the genetic algorithm are small size population size, small size maximum generation, large size crossover ratio, and small size mutation ratio respectively.

- Table A2 20-6-15 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	G	G
		Medium	G	G	G	
		Large				G
G		Small				G
		Medium				
	*	Large	V.G	V.G	V.G	G
C		Small	G			
		Medium		G	G	G
	*	Large	G	G	G	G
M		Small	V.G			
	*	Medium		G	V.G	V.G
		Large		G		

From the table above, when the parameters of the assembly line are medium size in units and model, large size in station, the minimal part set is case B, the optimal parameters of the genetic algorithm are small size population size, large size in maximum generation and crossover ratio respectively, medium size mutation ratio.

- Table A3 20-12-5 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	G	G
		Medium				
		Large	G	G	G	G
G	*	Small	V.G	G	G	V.G
		Medium				
		Large		G	G	
C	*	Small	V.G	G	V.G	V.G
		Medium				
		Large		G		
M	*	Small	G	G	G	G
		Medium	G	G	G	
		Large				G

From the table above, when the parameters of the assembly line are medium size units, large size models and medium size stations respectively, and the minimal part set is case A, the optimal parameters of the genetic algorithm are small size in population size, small size in maximum generation, small size in crossover ratio, and small size in mutation ratio respectively.

- Table A4 20-12-5 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G		G
		Medium			G	
		Large	G	G	G	G
G		Small	G		G	G
	*	Medium	G	V.G		G
		Large			G	
C	*	Small	G			V.G
		Medium		G	G	
		Large	G	G	G	
M		Small			G	G
		Medium			G	G
	*	Large	V.G	V.G		G

From the table above, when the parameters of the assembly line are medium size units, large size models and medium size stations respectively, and the minimal part set is case B, the optimal parameters of the genetic algorithm are small size in population size, medium size in maximum generation, small size in crossover ratio, and large size in mutation ratio respectively.

- Table A5 20-12-15 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	V.G	V.G
		Medium				
		Large	G	G		
G		Small	G	G	G	
	*	Medium	G		G	V.G
		Large	G	G	G	
C	*	Small	V.G	G	V.G	V.G
		Medium				
		Large		G		
M		Small	V.G	G		
	*	Medium		G	V.G	V.G
		Large		G		

From the table above, when the parameters of the assembly line are medium size units, large size models and stations respectively, and the minimal part set is case A, the optimal parameters of the genetic algorithm are small size in population size, medium size in maximum generation, small size in crossover ratio, and medium size in mutation ratio respectively.

- Table A6 20-12-15 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	V.G	V.G	V.G	V.G
		Medium				
		Large				
G	*	Small	V.G	V.G	V.G	V.G
		Medium				
		Large				
C	*	Small	V.G	V.G	V.G	V.G
		Medium				
		Large				
M		Small		V.G		
	*	Medium	V.G		V.G	V.G
		Large				

From the table above, when the parameters of the assembly line are medium size units, large size models and stations respectively, and the minimal part set is case B, the optimal parameters of the genetic algorithm are small size in population size, maximum generation and crossover ratio respectively, medium size in mutation ratio.

- Table A7 50-6-5 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P		Small		G		G
	*	Medium	V.G	G	G	G
		Large		G	G	
G		Small	G	V.G		
	*	Medium	G		V.G	G
		Large	G			G
C	*	Small	G	G		V.G
		Medium	G			
		Large	G	G	V.G	
M	*	Small	V.G	G		
		Medium		G	G	G
		Large		G	G	G

From the table above, when the parameters of the assembly line are large size in units, medium size in model and station, and the minimal part set is case A, the optimal parameters of the genetic algorithm are medium size in population size and maximum generation, small size in crossover ratio and mutation ratio respectively.

- Table A8 50-6-5 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	G	G
		Medium	G	G	G	G
		Large				
G	*	Small	G	G	G	G
		Medium			G	G
		Large	G	G		
C	*	Small	G	V.G	V.G	G
		Medium	G			G
		Large				
M	*	Small	G	V.G	V.G	G
		Medium	G			G
		Large				

From the table above, when the parameters of the assembly line are large size units; medium size models and stations respectively, and the minimal part set is case B, the optimal parameters of the genetic algorithm are small sizes.

- Table A9 50-6-15 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	G	G
		Medium				
		Large	G	G	G	G
G	*	Small	V.G	V.G	V.G	G
		Medium				
		Large				G
C	*	Small	G	G	G	G

M		Medium				
		Large	G	G	G	G
		Small			G	G
	*	Medium	V.G	V.G	G	G
		Large			G	

From the table above, when the parameters of the assembly line are large size units, medium size models and large size stations respectively, and the minimal part set is case A, the optimal parameters of the genetic algorithm are small size in population size, small size in maximum generation, small size in crossover ratio, and medium size in mutation ratio respectively.

- Table A10 50-6-15 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	G	G
		Medium	G			G
		Large	G	G	G	
G	*	Small	G	G	G	G
		Medium		G	G	
		Large	G	G	G	G
C		Small	G	G		G
	*	Medium	G	G	V.G	G
		Large				
M	*	Small	V.G	V.G	V.G	V.G
		Medium				
		Large				

From the table above, when the parameters of the assembly line are medium size units, large size models and medium size stations respectively, and the minimal part set is case B, the optimal parameters of the genetic algorithm are small size in population size, small size in maximum generation, medium size in crossover ratio, and small size in mutation ratio respectively

- Table A11 50-12-5 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	G	G	G	
		Medium				G
		Large	G	G	G	G
G		Small				
	*	Medium	V.G	V.G	V.G	V.G
		Large				
C		Small	G			
		Medium				G
	*	Large	G	V.G	V.G	G
M		Small				
		Medium				G
	*	Large	V.G	V.G	V.G	G

From the table above, when the parameters of the assembly line are large sizes units and models and medium size stations respectively, and the minimal part set is case A, the optimal parameters of the genetic algorithm are small size population size, medium size maximum generation, large size crossover ratio, and large size mutation ratio respectively.

- Table A12 50-12-5 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P		Small				
		Medium				
	*	Large	V.G	V.G	V.G	V.G
G	*	Small	G	V.G	V.G	V.G
		Medium				
		Large	G			
C	*	Small	G		G	G
		Medium		V.G		
		Large	G		G	G
M		Small				G
	*	Medium	G	G	G	G
		Large	G	G	G	

From the table above, when the parameters of the assembly line are large size in units and model, medium size in station, and the minimal part set is case B, the optimal parameters of the genetic algorithm are large size in population size, small size in maximum generation and crossover ratio, medium size mutation ratio.

- Table A13 50-12-15 in case A

Item	Best	Range	First running	Second running	Third running	Fourth running
P	*	Small	V.G	V.G	V.G	V.G
		Medium				
		Large				
G	*	Small	V.G	V.G	V.G	V.G
		Medium				
		Large				
C	*	Small	V.G		G	G
		Medium		V.G		
		Large	G		G	G
M		Small				
	*	Medium	V.G	G	G	V.G
		Large		G	G	

From the table above, when the parameters of the assembly line are all in large size, and the minimal part set is case A, the optimal parameters of the genetic algorithm are in small size in population size, maximum generation and crossover ratio respectively, in medium size in mutation ratio.

- Table A14 50-12-15 in case B

Item	Best	Range	First running	Second running	Third running	Fourth running
P		Small	G			V.G
	*	Medium	G	V.G	V.G	
		Large	G			
G		Small	G			V.G
	*	Medium	G	V.G	V.G	
		Large				
C		Small	V.G			G
	*	Medium		V.G	V.G	G
		Large				G
M		Small				
	*	Medium	V.G	V.G	V.G	V.G
		Large				