

## ORIGINAL ARTICLE

Toshiro Tochigi · Chiaki Tadokoro · Jun Kobayashi  
Izumi Sugawara · Sachie Takahashi

## Improving production systems of timber-processing plants IV: Application of genetic algorithms for appropriate positioning of processing machines

Received: November 14, 1997 / Accepted: April 22, 1998

**Abstract** The development of a layout plan for a new plant with the aid of genetic algorithms was studied to place the machines so that the plant floor was effectively utilized and the operation would not be impeded. Genetic algorithms are search algorithms based on the mechanics of natural evolution and natural genetics to solve problems in engineering fields. Simulation with the aid of genetic algorithms was undertaken step by step. The first seven hundred strings (chromosomes) were generated at random to organize an initial population. Each string consisted of 40 bits (genes), which represented characteristics of machines ( $x$ - and  $y$ -coordinates and inlet and outlet formations of materials on machines) in the binary coding. Then the simulation was undertaken by repeating selection, crossover, reproduction, and mutation of strings until all strings were saturated with the highest evaluation (fitness of chromosomes to environments in the case of creatures). Under some limitations, an acceptable layout plan of the modeled plant involving four wood-processing machines was obtained according to evaluation indices.

**Key words** Genetic algorithms · Plant design · Production control

### Introduction

As when constructing a new plant on a limited site or reestablishing the processing system of an existing plant so

it can produce new products, if the structure and floor area of the plant are already specified we must determine the most appropriate place to locate the processing machines. Another consideration is the most suitable transportation route between these machines. In past studies, after discussing the PERT method, the queuing theory, the OR technique, and the network theory, we introduced a computer simulation based on a calculus-based method (random search algorithms).<sup>1,2</sup> Unfortunately, it required a huge amount of time and labor. In this study we introduce genetic algorithms to solve the optimum layout problem mentioned above.

### Evolution of creatures and genetic algorithms

Evolution of creatures and exchange of genes

From the formation of the earth to the present, environmental conditions have been changing dramatically. During the development of the earth millions of plant and animal species have appeared, and large numbers have disappeared. Only those that adapted to their environment have survived and continued their evolution. Biological evolution entails repeated genetic changes, and the procedure can be defined simply as follows: (1) crossover; (2) reproduction; (3) selection; and (4) Mutation.

*Crossover* is the partial exchange of genetic materials (mainly genes) between homologous chromosomes (parents) by their mating. It yields two new chromosomes (children) partially with their parents' chromosomes. *Reproduction* is the process by which individual chromosomes survive during the propagation of a population to the next generation depending on the results of selection and crossover. *Selection* determines the degree of fitness of chromosomes in the environment. High-fitness chromosomes have a high probability of producing one or more offspring that survive to the next generation. Low-fitness chromosomes vanish from the living world before the next generation. *Mutation* is the occasional alteration of gene(s)

T. Tochigi (✉) · C. Tadokoro  
Institute of Agricultural and Forest Engineering, University of  
Tsukuba, Tsukuba, 305-8572, Japan  
Tel. + 81-298-53-4894; Fax +81-298-53-4894  
e-mail: toshiro@sakura.cc.tsukuba.ac.jp

J. Kobayashi · I. Sugawara · S. Takahashi  
Faculty of Regional Environment Science, Tokyo University of  
Agriculture, Tokyo 156-8502, Japan

Part III of this series appeared in *Mokuzai Gakkaishi* 37: 702–710, 1991. Part of this report was presented at the 47<sup>th</sup> annual meeting of the Japan Wood Research Society, Kochi, April 1997

in a chromosome with low survival probability. Mutation rates are extremely low in nature.

Genetic algorithms

Genetic algorithms (GAs) are search algorithms based on the mechanics of natural evolution and natural genetics.<sup>3,4</sup> In every generation, a new set of artificial creatures (chromosomes, called “strings” in GAs) is created using “bits” (genes) and pieces of the fittest of the earlier generation. Therefore, GAs are defined as computer simulation techniques that can solve problems in engineering fields by artificially imitating the mechanics of natural evolution.

The mechanics of a simple GA are simple, involving nothing more complex than a rearrangement of strings in order (selection of chromosomes), copying strings (reproduction of chromosomes), exchanging partial strings (crossover of chromosomes), and occasional and random alteration of a bit (mutation of a gene).

**Model plant**

The floor area of a model plant is proposed hypothetically. It is limited and divided into  $12 \times 7$  grids, as shown in Fig. 1. We assume that four processing machines must be placed on the floor, and their arrangements are defined as shown in Fig. 2. Inlet and outlet formations of materials on machines are defined as shown in Fig. 3. The nodes of the grids are places where processing machines could be placed.

Limitations for designing the model plant

To design a model plant, the following limitations are tentatively established.

1. *Positional limitation.* An entrance gate of raw materials (SP, Fig. 1) and an output gate of products (EN, Fig. 1)

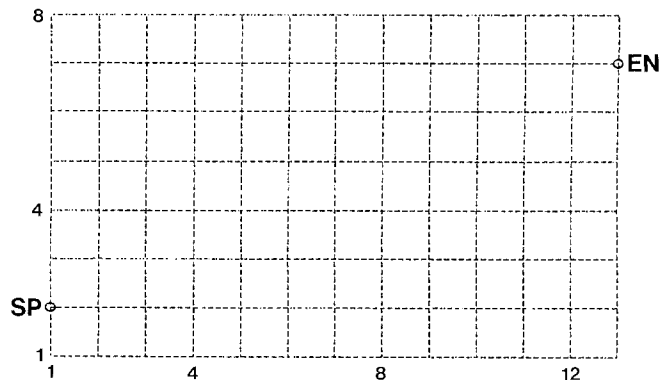


Fig. 1. Floor area of a model plant divided into  $12 \times 7$  grids. SP, entrance gate of raw materials; EN, output gate of products

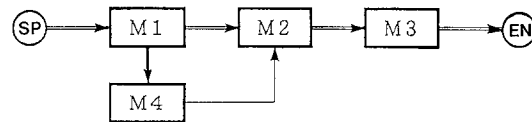


Fig. 2. Proposed arrangements of processing machines in the model plant. M1–M4, processing machines;  $\Rightarrow$ , main production line;  $\rightarrow$ , secondary production line from M<sub>1</sub> to M<sub>4</sub>;  $\rightarrow$ , secondary production line from M<sub>4</sub> to M<sub>2</sub>. Arrows represent paths of material flow

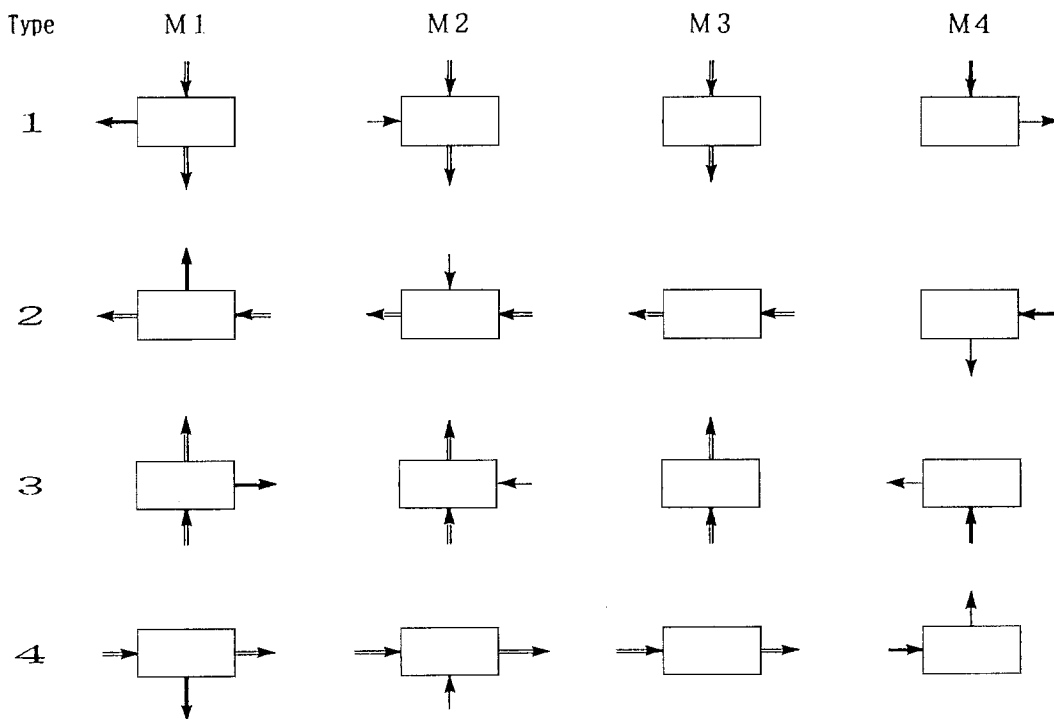


Fig. 3. Inlet and outlet formation of materials on machines processing machines

were fixed, and their coordinates were given by (1, 2) and (13, 7), respectively. Because of conditions of a plant's location and function, there are certain areas where machines and transportation routes cannot be installed.

2. *Distance from machine to machine.* The Distance from the entrance gate of the raw materials (SP) to the first machine ( $M_1$ ), from  $M_1$  to the second machine ( $M_2$ ), from  $M_2$  to the third machine ( $M_3$ ) and from  $M_1$  to the fourth machine ( $M_4$ ) were defined as  $\geq 4, 6, 4,$  and  $5$  units, respectively, even if materials were transported horizontally or vertically (or both). What is meant here by "a unit" is the distance between the node of the grid and the adjoining node (width of a grid).

3. *Relation between machines.* The  $x$ -coordinate of  $M_4$  should be more than that of  $M_1$  and less than that of  $M_2$  simultaneously. The  $y$ -coordinate of  $M_4$  should be less than the  $y$ -coordinates of  $M_1$  and  $M_2$  simultaneously.

4. *Transportation rate of materials.* It was possible to change the direction of the route to the  $x$ - and  $y$ -axes only at the nodes of the grids. As the routes were installed to transport materials from one machine to others, the route could not be extended backward to prevent a counterflow of materials once the route had been extended. Overpassing the existing routes was also prohibited. The transportation rate of materials for running horizontally was defined and required 2s/unit. Vertically it took 1s/unit. Turning the running direction, from horizontal to vertical or vice versa, consumed 0.5s once.

## Simulation of the layout plan by GA

### Procedures of the GA simulation

A simulation procedure of genetic algorithms to obtain the fittest layout of wood-processing plants was undertaken step by step as shown in Fig. 4: Seven hundreds of strings were generated at random, and they organized the initial population as shown in Fig. 5. Each string consisted of 40 bits. Therefore, this population was chosen through 28000 ( $700 \times 40$ ) successive randomization. In a string, each 10 bits were provided for representing characteristics of a machine in a binary coding. The  $x$ - and  $y$ -coordinates of a machine were represented by four bits each, two bits were used for the inlet and outlet formations on the machine, respectively.

Selection, the next step of the GA simulation, is a process wherein individual strings are rearranged according to their fitness (described later). Strings with higher fitness values were ranked in the upper portion of the population and had a higher probability of contributing one or more offspring to the next generation according to priorities of fitness determined by the weighted roulette method.<sup>4</sup> This operation was an artificial version of natural selection, a darwinian survival of the fittest among chromosome creatures.

Reproduction is an inevitable process that creates new strings to the next generation. In this study it was done in

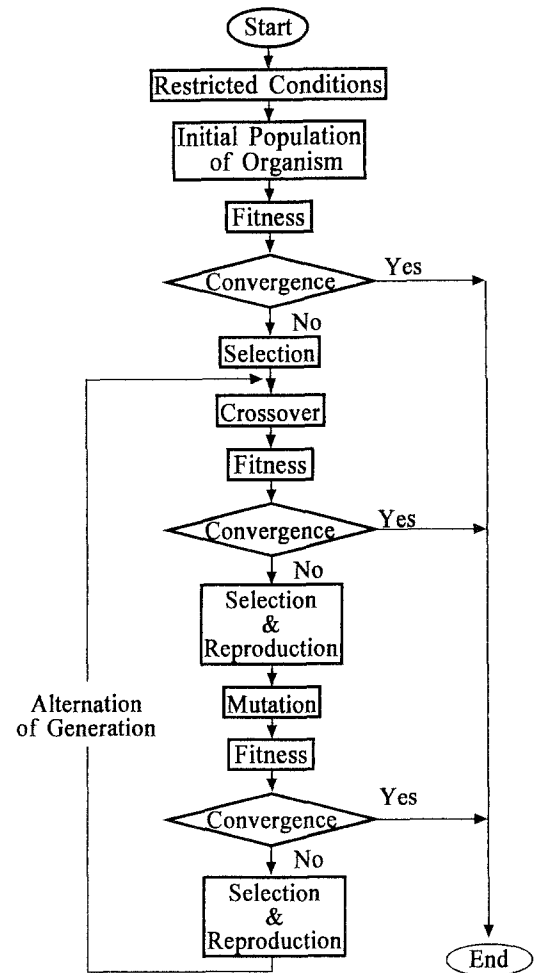


Fig. 4. Flow of genetic algorithm (GA) simulation in this study

three ways, as shown in Fig. 6: The first step was an exact copying of the strings with high fitness values after the selection of strings in the old population, the second was transformation of strings in order of fitness, and the third was random transformation of strings from the old population to the new.

After reproduction of the strings to the new population, simple (one-point) crossovers of strings proceeded.<sup>5</sup> First, two strings were mated at random. After deciding on the position of bits ( $k$ ) at random between 1 and string length less than 1 ( $L - 1$ ), the paired strings underwent crossing-over by exchanging all bits between  $k + 1$  and  $L$ , inclusively, as shown in Fig. 7; then two new strings were created from their parents. Because strings with higher fitness values were given higher priorities for crossing over, they crossed over many times and exchanged their strings with different mates at each crossover. In this study, strings for crossing over were decided at random according to their fitness values.

Mutation handling was the occasional alteration of the value of a bit, as shown in Fig. 8; it did not happen in every generation. The frequency of mutation was low with small probability. It was fixed at 6%, in this study, to the number of alterations of generations and the number of strings in

```

101010101100100100010101101000000001000
0000111110010010010111000010110101001010
1101010100010110010001110101000100111010
1111011000011000010011110101001101011110
0110001001001001110110010001101101000000
000101110111111000110110001100000001111
011101101101111010101001011111010001000
1010101101111000111000100001110000101010
1101011101101101011010011100111010011110
000111100100100111001111010100001110001
0000000010010010011001100000100110000100
0011100111111000100100011000010111101010
0111011110100010010001100101011011100100
0111110011000111001110110000111110100110
11000001111011010010110011011101010001
111101101110000001011100011011000110110
100000111001000110001010111101100010101
0110110100000110111101000100000110101011
1011001111000101100011101110100000001011
1111010100011101000001001001001011100101
101010101111011001011001110100110010011
101111100101110100111000110100111100001
0011010001101110110011101110110100011001
111101001100001111101000100010111100111
10111101101100110011001001001101110111
1011110111000101111001010101000111011110
0010110100100001101110011100101100011001
0100111111000111000101111001111000111011
0000000110110110101000010110101100010101
1001110100111011010011010111010001001010
000101011110100000011000100100111001011
0010011010001111011011000111101110110000
111110001001011000100000110011011011110
1111001100000000110010110111110001100010
0000000110001001010110011000101101011001
010010111011000000100110101111110110110
01010010101100000010110100111110110110
0101001010110000001011010011100100010011
    
```

Fig. 5. Example of the initial population of creatures (strings) in the GA simulation

Reproduction

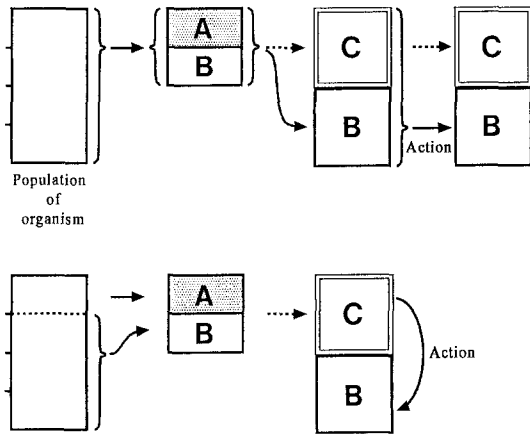


Fig. 6. Reproduction of strings introduced in this study. A, strings selected, in order of fitness; B, strings selected in random; C, copied strings

the population. The times of mutation were decided at random according to the frequency of mutations. When mutation occurred, a string and a bit were also decided at random, and the value of the bit was altered automatically.

Simulation was undertaken until all of the strings in the population were saturated with the fitness value<sup>6</sup>. When the simulation was completed, an optimum route for a line from

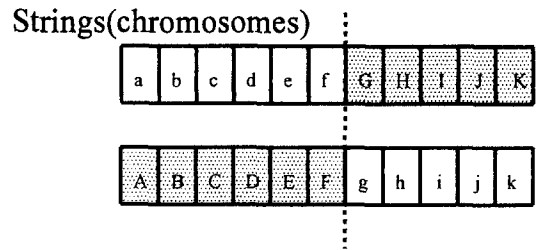
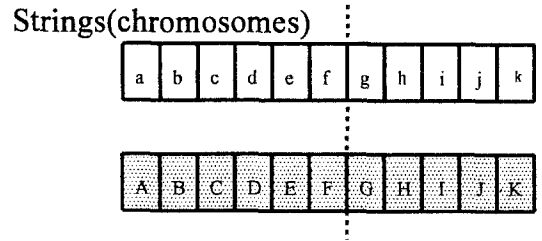


Fig. 7. Simple (one-point) crossover of strings

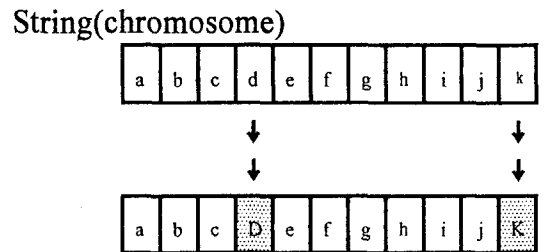


Fig. 8. Mutation of bits (genes) in a string

one machine to another was determined by the Dijkstra method,<sup>7</sup> which could determine the shortest route between two machines, avoiding crossing the existing routes.

Evaluation of the simulation (fitness)

During the course of manufacturing a product, there are various positions for the machines and routes connecting them to be considered. The best layout plan among such possibilities should be chosen with evaluation indices, called the “fitness value” in this study.

The speed of the processing machine was not considered in this study. Of course, a single condition or a combination of the following four conditions could be selected to determine the most appropriate layout plan. In a natural population, fitness can be determined by the creatures’ ability to survive predatory enemies and environments. In this study, simulated with respect to layout plans of wood-processing plants, to obtain the fittest conclusion fitness was evaluated by the following three indices and a supplemental one.

1. *Minimum distance.* This step is to minimize the distance between the entrance gate of raw materials and the output gate of products, which would in fact minimize the number of times the direction of a route must be changed.

This does not mean that the time for transporting the materials would be minimized as well. The length of an entire transportation route was obtained by summing the distances between processing machines.

2. *Minimum time consumption.* The total time it took for a material to pass over the entire route was determined. A comparison of routes was made by measuring the time needed for a material to enter the entrance gate of raw materials and reach the output gate of products, taking into consideration the distance between machines, and the time spent for transportation.

3. *Minimum number of times for changing direction of routes.* To change the direction of a route too often would result in complications for the layout plan and an instant impediment to the flow of materials, thereby taking more time to transport materials. The total distance necessary for processing would mean greater construction costs. If the processing area exceeds the available space, however, it is necessary to change the direction of the route. In such a case, it would be best to install the fastest possible route and extend the parts of it that flow straight. On the other hand, changing the direction of the route should shorten the slow portions of the route.

4. *Comparison of area balance.* This is a supplemental index to evaluate the simulation. It considers the conditions pertaining to the balance within the space separated by transportation routes between the entrance gate of raw materials and the output gate of products.

Fitness values were obtained by summing the values of these three indices after each genetic action. Strings with smaller values were evaluated as having higher fitness. Evaluation with a supplemental index was also considered when the most advantageous string was determined.

## Results and discussion

The ratios of fitness values to the final fitness (saturated) value for the whole population were calculated after each genetic action (Fig. 9). Simulation was started from a hybrid population at first. As genetic actions were repeated and progressive generations were established, the fitness of the population changed to facilitate better survival. After about 50 generations, the population became homogeneous, consisting of close relatives. Afterward, if the renewal of generations was continued, the population returned to a hybrid one, and its ability to survive was becoming recessive.

After the whole-population saturated fitness value was determined, the final conclusion were drawn with respect to the recommended positions of machines, and their orientation was obtained. The final output of the GA simulation is shown in Fig. 10. The recommended layout in the modeled wood-processing plant, the final result of the simulation, is shown in Fig. 11.

Recommended layouts, when the entrance gate of raw materials and the output gate of products were changed, are illustrated in Figs.12–15.

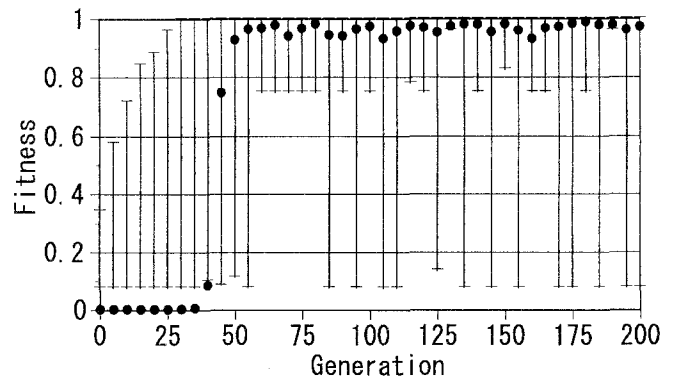


Fig. 9. Ratio of fitness values in the GA simulation. Vertical lines, range of fitness values in the population (maximum to minimum); filled circles, ratio of saturated strings in the population

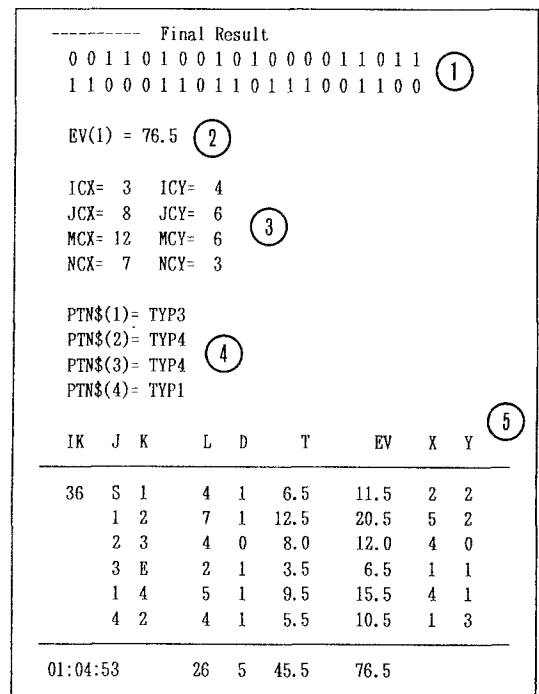


Fig. 10. Final result of the GA simulation. Encircled Numbers: 1, saturated string represented its bits in binary coding; 2, fitness value of the saturated string; 3, x- and y-coordinates of machines; 4, inlet and outlet formation of materials on machines; 5, data for the optimum layout plan representing generation (IK), total units from a machine or the input gate to a machine or the output gate (L), number of times to change the direction of a path (D), times required (T), fitness value (EV) and horizontal and vertical units (X and Y) from a machine or the input gate (J) to a machine or the output gate (K)

## Conclusion

Developing a layout plan for a model plant with the aid of genetic algorithms to place machines so that the plant floor was effectively utilized and the operation would not run into impediments was studied. Under some limitations (e.g., allowable length between machines, the direction of inlet and outlet formation of materials on machines, and the time required for transporting materials) we obtained an accept-

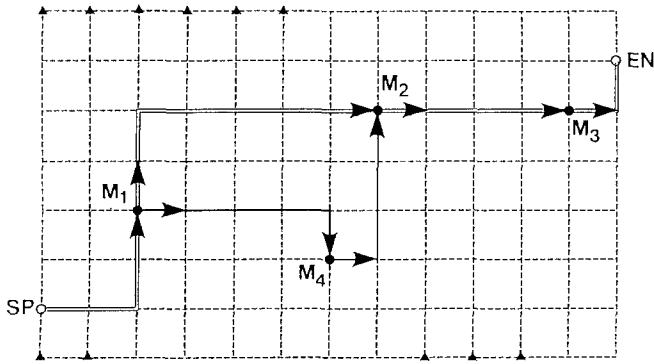


Fig. 11. Recommended layout plan in the modeled plant by the GA simulation. *Triangles*, prohibited places to install processing machines and transportation routes

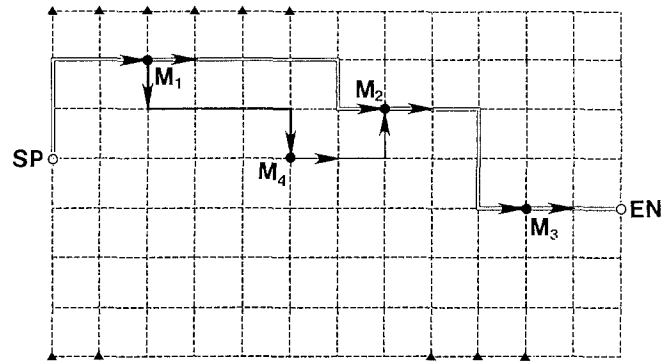


Fig. 14. Recommended layout plan when SP and EN were fixed on (1,5) and (13,4), respectively. See Fig. 11. for further explanations

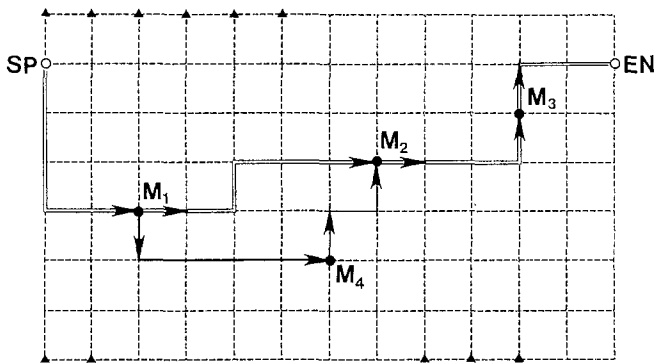


Fig. 12. Recommended layout plan when SP and EN were fixed on (1,7) and (13,7), respectively. See Fig. 11. for further explanations

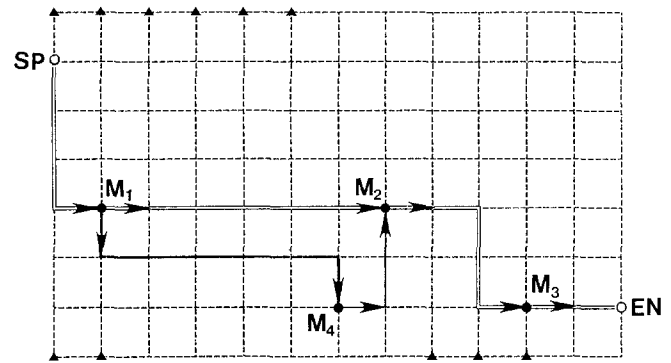


Fig. 15. Recommended layout plan when SP and EN were fixed on (1,7) and (13,2), respectively. See Fig. 11. for further explanations

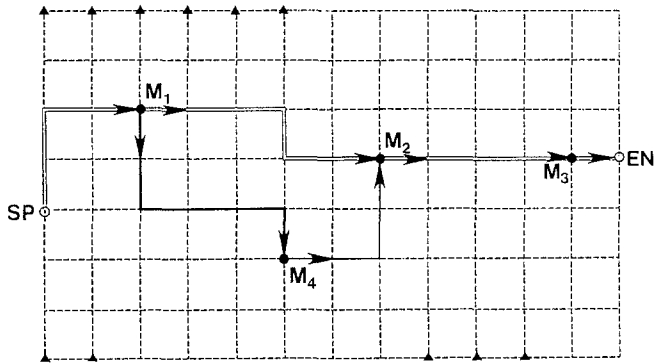


Fig. 13. Recommended layout plant when SP and EN were fixed on (1,4) and (13,5), respectively. See Fig. 11. for further explanations

able layout plan of the model plant, involving four wood-processing machines. In a future study, the time analysis of material flows in the wood-processing plant simulated in this study will be considered by introducing the queuing theory.

We believe that genetic algorithms are strong tools for solving problems in engineering fields and obtaining an optimum solution if candidates for the solution (strings in an

initial population) and the final optimum solution can be represented in binary coding.

**Acknowledgment** The authors thank Dr. Koichi Ichihara, Associate Professor, Institute of Agricultural and Forest Engineering, University of Tsukuba, for his valuable suggestions.

**References**

1. Tochigi T, Amemiya R, Tadokoro C, Hashimoto H (1988) Improving production systems of timber-processing plants. Part 1. Development of the computer-simulation procedure to design the production line, *Mokuzai Gakkaishi* 34:314-319
2. Tochigi T, Amemiya R, Tadokoro C, Hashimoto H (1988) Improving production systems of timber-processing plants. Part 2. Appropriate positioning of processing machines in sawmills. *Mokuzai Gakkaishi* 34:320-325
3. Holland JH (1992) Genetic algorithms. *Sci Am* 9:44-50
4. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, New York, pp 10-18
5. Hirano H (1992) How to solve building-blocks problems by genetic algorithms (in Japanese). *Interface* 2:108-112
6. Kitano H (1992) Genetic algorithms (in Japanese). Sangyo-Tosho, Tokyo, pp 13-14
7. Iri M (1986) Modern applied mathematics (in Japanese). Nihon Hoso Shuppan Kyokai, Tokyo, pp 114-118