Study on Efficient Planning for Advanced Logistics Network Model Based on Robust Genetic Algorithm

ATAKA, Shinichiro

February 2010

Waseda University Doctoral Dissertation

Study on Efficient Planning for Advanced Logistics Network Model Based on Robust Genetic Algorithm

ATAKA, Shinichiro

Graduate School of Information, Production and Systems

Waseda University

February 2010

Abstract

In recent years, the management concept of company is greatly reformed by rapid development of information technologies and spread of the Internet. By this transition, the role of Supply Chain Management (SCM) in production and logistic information systems is ever becoming more important, at the same time production and logistics model designs renovation requirements. The transportation is one of the most important roles for economic goals such as expansion of the market, balancing of physical distribution and production activity. Nowadays, diversification of demand has changed from the process of trading products after finished mass-production to using high-mix low-volume production. Currently, the problem that many companies need to deal with is to quickly and flexibly satisfy various needs of the customers. Therefore, the just-in-time delivery or deliveries with multi-product, small-lot, high frequency are important subjects with a physical distribution.

The transportation problem (TP) has been discussed in the field of Operations Research. The objective is to determine delivery amounts with minimizing the total delivery cost and satisfying customer demands. However, the general TP model cannot be applied to the real world. More concrete constraints and extension of the model are needed. Although various extended models and their solution methods have been proposed, the extended models could not deal with various elements of the real world such as inventory.

Genetic Algorithm (GA) is one of meta-strategies imitating evolution of creatures. It has been applied to various fields like TP since it was proposed. But GA has several problems, such as high calculation load, premature convergence to a local minimum and complexity of parameter setting.

This thesis propose realistic TP models which consider exclusionary side constraints, kind of products, inventory, direct shipping route and multiple periods. These elements reflected needs for the modern distribution. Although these proposal TP models have different complexity, in order to obtain stable solution in realistic time, we propose solution methods based on GA. We found the excellent technique by combine improved gene expression and selection technique. Moreover, through the analysis of evaluation result, we clarify the effectiveness and limit of each technique. This article has a great value that proposed effective and stability techniques for modern various TPs. This article greatly contributes to the future distribution optimization.

Contents:

List of Figures

List of Tables

Chapter 1. Introduction

1.1 Background and aim of this research

In recent years, the management concept of company is greatly reformed by rapid development of information technologies and spread of the Internet. By this transition, the role of Supply Chain Management (SCM) in production and logistic information systems is ever becoming more important, at the same time production and logistics model designs renovation requirements [1] [2]. Transportation, storage, packing, cargo work, distributed processing, and physical distribution information are included in logistics as the main functions. Among these six functions, the transportation is one of the most important roles for economic goals such as expansion of the market, balancing of physical distribution and production activity.

Formerly, in the manufacturing industries, the procurement of physically distributed raw materials necessary for production and shipping of products to physically distributed customers were the objectives of logistics systems. Moreover, in logistics or service industries, one of the objectives of a logistics system was the physical distribution activity that purchases a product from a maker and provides to a retail store, etc. Because of this, various efforts for efficiency have been performed.

However, unlike the production field currently introduced to management engineering, such as robots, TQC and the Kanban system, the rationalization of physical distribution fields with many labor-intensive elements, such as truck drivers, is behind. Nowadays, diversification of demand has changed from the process of trading products after finished mass-production to using high-mix low-volume production. Currently, the problem that many companies need to deal with is to quickly and flexibly satisfy various needs of the customers. Therefore, the just-in-time delivery or deliveries with multi-product, small-lot, high frequency [3] are important subjects with a physical distribution. In recent years, "Green Logistics" becomes a significant issue, because of the impact on the environmental problems.

The logistics include the portion of purchasing raw materials from suppliers. Then, raw materials will be sent to the plant. After product was manufactured, delivery to customers are the last step of these activities. Various conversions such as functional and physical conversion, position conversion and time conversion will be given in logistics bases. Although functional and physical conversion is mainly a production activity in a factory, the distributive processing in a delivery center is also included. Position conversion means keeping required quantity as inventory at required places. When order was coming, it enables to use goods immediately. Moreover, plants, Distribution Centers (DCs), retail stores, etc. are examples of logistics bases. Each company attaches various names and there are many bases with complex functions.

The transportation problem (TP) has been discussed in the field of Operations Research. The objective is to determine delivery amounts with minimizing the total delivery cost and satisfying customer demands. However, the general TP model cannot be applied to the real world. More concrete constraints and extension of the model are needed. Although various extended models and their solution methods have been proposed [4] [5] [6] [7] [8], the extended models could not deal with various elements of the real world such as inventory. In addition, for large-scale problems, approximate approaches based on the theoretical analysis of problem and heuristic algorithms have been extensively studied, because of calculating an exact solution is difficult. Although the former approaches have many advantages, such as being able to obtain a good solution in short time and the minimum of a solution. However, it cannot respond to the structural change of the problem.

In recent years, researches on optimization techniques based on meta-strategy such as Genetic Algorithm (GA), Simulated Annealing (SA), and Tabu Search (TS) are widely studied [9] [10]. These techniques have high robustness as compared with the techniques based on the conventional theories. They can be adapted to composite problems and addition constraints, so that the structure of problem can change over time. However, majority of researches on the conventional meta-strategy were based on the simple TP model. Genetic Algorithm (GA) is one of meta-strategies imitating evolution of creatures. It has been applied to various fields like TP since it was proposed. But GA has several problems, such as high calculation load, premature convergence to a local minimum and complexity of parameter setting.

The main purposes of this research are proposing a new extended TP model, and developing solution methods by GA. For this reason, the previous researches for TP and issues which should be considered are clarified. After the survey, we discuss and formalize more realistic TP models. Also, efficient optimization techniques for the proposed models are developed.

1.2 Composition of thesis

This paper consists of seven chapters.

Chapter 1 is an introduction. Chapter 2 describes the fundamental outline of the traditional TP model and GA. Then, we clarify the advantages of applying the GA to TP model.

In Chapter 3, the multi-product two-stage logistics network model with exclusionary side constraints is proposed. In this TP treats multi kinds of production and exclusionary side constraints. Furthermore, the model has the feature that the delivery cost of every product can be different. Until now, TP with exclusionary side constraints were difficult to be solved by GA. Then, in order to solve the proposal TP, we develop an improved priority-based representation. If this technique is used, the distinction routine of an infeasible solution is unnecessary. Therefore, we can reduce computation time. In the numerical simulation experiments, we compared following four GAs based on improved chromosome representation.

(1) Priority-based GA which applied WMX (Weight Mapping Crossover)

(2) Priority-based GA which applied PMX (Partially Matched Crossover)

(3) Hybrid Priority-based Genetic Algorithm (h-priGA) using WMX

(4) Hybrid Priority-based Genetic Algorithm (h-priGA) using PMX

Here, h-priGA has an auto tuning mechanism (FLC: Fuzzy Logic Controller) for the parameters [22] [35]. As a result, the proposed TP model can be solved by a GA which utilizes the improved priority-based representation.

In Chapter 4, the two-stage transportation problem with inventory and exclusionary side constraints (esc-2ITP) is proposed. This esc-2ITP considers the concept of inventory and designs a transportation plan over multiple periods. Many companies aim at improvement of logistics, as well as improvement of inventory control. Moreover, how to reduce inventory costs as much as possible is an important objective. However, if there is little volume of inventories, the service level may below to fall. Therefore, it is necessary to adjust the balance of suitable inventories and the service level. We formulate esc-2ITP with consideration of these problems.

In Chapter 5, a solution method for the esc-2ITP is proposed. The solution method for Chapter 3 was improved only the chromosome representation part. Thus, the weak points of GA were not improved. Therefore, we design a new GA approach called the Boltzmann random key-based GA (Brk-GA). Since this algorithm has a simple structure, we can reduce the computation time. The selection mechanism of Brk-GA uses the Boltzmann distribution [40]. It is known that Boltzmann selection can decrease a probability of premature convergence to local minimum. Mori, et al proposed Thermodynamic Genetic Algorithm (TDGA) which explicitly consider the diversity of population [41]. TDGA estimates the diversity of population clearly as entropy. In this literature, TDGA is applied to a knapsack problem in experiment. The effect for TP is not yet proved. Ambedkar, et al. [42] proposed a Cauchy annealing schedule for Boltzmann selection scheme. It based on a hypothesis that selection-strength should increase as evolutionary process goes on and distance between two selection strengths should decrease for the process to converge. They conducted comparative experiments of Boltzmann selection with Cauchy annealing schedule and traditional Boltzmann selection. Both techniques were applied to several variable functions. However, authors did not describe the computation time. It seems that computation time is needed since the proposal technique is complicated. We need flexible GA which can respond in an instant for the sequential change of road environment.

Including the above literatures, Boltzmann selection has mainly adapted classical GA which uses binary number coding for chromosome. Moreover, comparison of h-GA equipped with FLC and GA equipped with the using the Boltzmann selection technique is not performed. In numerical experiments, we apply four different GAs containing Brk-GA to esc-2ITP, and execute comparative experiments which solve each problem

30 times respectively. From the results, although Brk-GA is inferior to h-priGA in average computation time, Brk-GA shows the best performance in the best solution, the average of the best solution of 30 calculations, and the standard deviation. When compared with the st-GA that is oldest method in the four techniques, the best solution and computation time are respectively improved by 66.1% and 6.3%.

In Chapter 6, the Progressive Flexible Logistics Network (PFLN) model that considers today's delivery forms is proposed. The esc-2TIP model proposed in Chapter 4 includes the concepts of inventory and time. However, the basic network structure of esc-2ITP is two-stage TP. Therefore, in this chapter we propose the PFLN model as a more extended TP with new connection form. The main difference of PFLN and esc-2ITP proposed in Chapter 4 is addition of retailers and duplication of direct delivery from producing districts. These added new features allow us to model currently distribution styles. Moreover, we propose a network segment method which divides PFLN to three sub-networks based on difference of delivery form. This method can avoid inefficiency caused by using different chromosomes in accordance with slight changes of the network. Comparative experiments using five GAs show that the flexible logistics network model can be solved by Brk-GA. The evaluation results performed 20 times show that Brk-GA is superior to other five techniques in the best solution, the averages of solutions, the standard deviation, and the computation time. On the other hand, when Flc is included in Brk-GA, it is observed that the average and standard deviation of the solutions become worse. Although soft computing technique has feature of obtaining suboptimal solutions in an early stage, the robustness and reliability may be affected whenever we calculate the solutions that have large variation. In conjunction with the result of Chapter 5, it is shown that Brk-GA is superior and more effective than traditional GA developed for TP.

In Chapter 7, conclusions about the knowledge acquired from this research are discussed, and future subjects are described.

Chapter 2. Transportation Problem (TP) and Genetic Algorithm (GA)

2.1 Introduction

TP is one of the combinatorial optimization problems, and various models and their solution methods were proposed [4] [5] [6] [7]. In a large-scale problem, approximate approaches based on the theoretical analysis in problem or research on heuristic algorithms occupies most. Because of it is difficult to calculate a strict solution. Although the former has many advantage such as being able to obtain a good solution in short time or the minimum solution is guaranteed. It cannot respond fault to the structural change of problem. On the other hand, heuristic algorithms have many characters that disagree with this problem. This chapter describes traditional TP model and outline of GA.

2.2 The outline of Transportation Problem (TP)

***Basic TP model*:**

TP is basic network problem proposed by Hitchcock (1941). The objective is minimizing the total cost, and all constraints are satisfied. At the same time, TP makes transportation plan of products that send from several plants to several destinations. In a field of OR, this model is known widely as one of linear programming. In a field of logistics network optimization, the main issue is proposal of improvement models based on this model.

The basic TP with I plants and J DCs is formulated as follows:

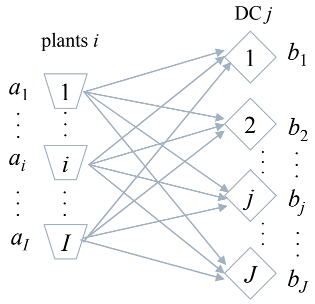


Fig. 2.1: Basic TP model

$$\min \quad z(x) = \sum_{i=1}^{I} \sum_{j=1}^{J} c_{ij} x_{ij} \tag{2.1}$$

$$\text{s.t.} \quad \sum_{j=1}^{J} x_{ij} \le a_i, \ i = 1,2,\ldots,m \tag{2.2}$$

$$\sum_{i=1}^{I} x_{ij} \ge b_j, \ j = 1,2,\ldots,n \tag{2.3}$$

$$x_{ij} \ge 0, \qquad \forall i, j \tag{2.4}$$

Indices:

i : index of plants (i = 1, 2,…, I)

j : index of DCs (j = 1, 2,…, J)

Parameters:

a_i : capacity of Plant i

b_j : demand of DC j

c_{ij} : delivery cost from plant i to DC j

Decision variables:

x_{ij} : delivery amounts from plant i to DC j

***Fixed-charge TP model*:**

The fixed-charge TP (fcTP) is one of the famous models as extension of traditional TP [11] [12] [13] [14]. This model has fixed costs for every route and transportation costs that is proportional to shipment amount. In a logistics network, a fixed cost may be incurred each shipment between plant and DC, or it may result in a fixed amount on investment. The fcTP is difficult to solve due to the presence of fixed costs, which cause discontinuities in the objective function. The objective is to decide shipment amounts of each routes so that total cost is minimized.

$$\min \quad z = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(c_{ij}x_{ij} + d_{ij}y_{ij}\right) \tag{2.5}$$

$$\text{s.t.} \quad \sum_{j=1}^{J} x_{ij} \le a_i\,,\ \forall i, j \tag{2.6}$$

$$\sum_{i=1}^{I} x_{ij} \ge b_j\,,\ \forall i, j \tag{2.7}$$

$$x_{ij} \ge 0,\ \forall i, j \tag{2.8}$$

$$\text{where } y_{ij} = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x = 0 \end{cases}$$

Indices:

i : index of plants (i = 1, 2,…, I)

j : index of DCs (j = 1, 2,…, J)

Parameters:

a_i : capacity of Plant i

b_j : demand of DC j

c_{ij} : delivery cost from plant i to DC j

d_{ij} : fixed cost for using route i - j

Decision variables:

x_{ij} : delivery amount from plant i to DC j

y_{ij} : 0-1 decision variable.

***TP with exclusionary side constraints*:**

Sun (1998) proposed nonlinear TP model that called TP with exclusionary side constraints (escTP) [15] [16] [17]. This escTP has additional constraints in which simultaneous shipment from specific plant pair to specific customer is prohibited. For example, both of frozen and nonfreezing foods are not delivered, or foods and poisonous products cannot be stored together, although they may be transported via the same distribution system. Moreover, manufacturer may ship the same product to different delivery addresses. In this thesis, we propose advanced TP that mainly used this feature. If exclusionary side constraints can be efficiently satisfied in a calculation algorithm, the limitation of a course according to kind of product etc. will attain.

$$\min \quad z(x) = \sum_{i=1}^{I} \sum_{j=1}^{J} c_{ij} x_{ij} \tag{2.9}$$

$$\text{s.t.} \quad \sum_{j=1}^{J} x_{ij} \le a_i, \ \forall i \tag{2.10}$$

$$\sum_{j=1}^{J} x_{ij} \ge b_j, \ \forall j \tag{2.11}$$

$$x_{ij} x_{lj} = 0, \quad (i,\ l) \in D_j, \ \forall j \tag{2.12}$$

$$x_{ij} \ge 0, \quad \forall i, j \tag{2.13}$$

Fig. 2.2: escTP model

Indices:

a_i : number of units available at plant *i*.

b_j : number of units demanded at DC *j*.

Parameters:

c_{ij} : shipping cost 1 unit from plant *i* to destination *j*.

$D_j = \{(i, l) |$ good from source *i* and *l* cannot be simultaneously shipped to destination $j\}$.

Decision variables:

x_{ij} : delivery amounts from plant i to destination j.

***Two-stage TP*:**

TP models described above were classified into one-stage models. This means that there are only two kinds of elements such as suppliers and demand-places on a network. The two-stage TP (ts-TP) consider two kinds of elements, and consists of two segments between Plants and DCs, and between DCs and customers [18] [19]. Compared with basic TP, ts-TP has more realistic delivery form. However, this model treats total delivery cost for one kind of product.

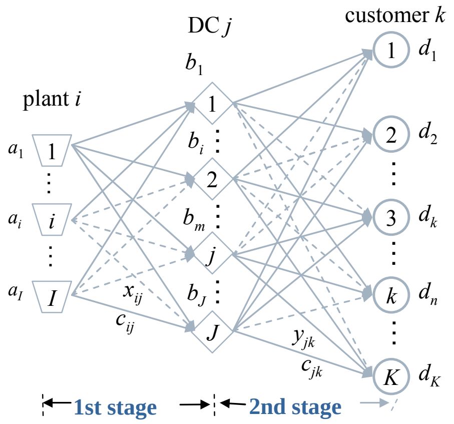


Fig. 2.3: The image of ts-TP

$$\min \quad z = \sum_{i=1}^{I}\sum_{j=1}^{J} c_{ij}x_{ij} + \sum_{j=1}^{J}\sum_{k=1}^{K} c_{jk}y_{jk} \qquad (2.14)$$

$$\text{s. t.} \quad \sum_{j=1}^{J} x_{ij} \le a_j, \quad \forall i \qquad (2.15)$$

$$\sum_{k=1}^{K} y_{jk} \le b_j, \quad \forall j \qquad (2.16)$$

$$\sum_{j=1}^{J} y_{jk} \ge d_k, \quad \forall k \qquad (2.17)$$

$$x_{ij},\ y_{jk} \ge 0, \quad \forall i, j, k \qquad (2.18)$$

Indices:

i : index of plants (i = 1, 2,…, I)

j : index of DCs (j = 1, 2,…, J)

k : index of customers (k = 1, 2,…, K)

Parameters:

a_i : capacity of plant i

b_j : capacity of DC j

c_{ij} : delivery cost from Plant i to DC j

c_{jk} : delivery cost from DC j to Customer k

Decision variables:

x_{ij} : delivery amounts from plant i to DC j

y_{jk} : delivery amounts from DC j to Customer k

TP models introduced in this section are extremely common models. Although various derived models based on these TPs have been proposed [20] [21] [22] [23] [24] [25] [26]. However, it is difficult to apply these models to the real world. If it applies to the real world, we must consider the additional concept such as inventory or product life cycle etc. We propose more realistic TP model after Chapter 3.

2.3 Outline of Genetic Algorithm (GA)

In early the 1970s, John Henry Holland and his group invented GA to solve complex optimization problem easily and powerfully with simple mechanism by using the feature of nature evolution.

In recent years, evolutionary approaches have been successfully applied to TP. Michalewicz and Viagnaux (1991) are first researchers who discussed about application of GA for solving linear and nonlinear transportation problems [27] [28]. GA is one of search technique that obtained the hint from the theory of Darwin. In addition, we do not need to consider limit of application field. In detection of optimal solution, in order to search solution space systematically and efficiently, GA simulates the evolution process of a creature. Usually, GA treats coded solution as chromosome, at the same time set of chromosomes form a population. Each chromosome expresses points in solution space. In GA calculation, iteration process to obtain a desirable solution is called generation. In addition, the chromosomes with high fitness value in current generation can generate a new solution (offspring) by crossover operator. Moreover, in order to add diversity, mutation operator generates offspring that has new feature into new solution space. A population of next-generation is generated from a set of offspring and current population. In GA, individual with high fitness can survive for a long time. At the same time, this individual can pass down the advantageous feature to next generation based on the theory of natural selection. After several generations, the algorithms choose one chromosome as optimum or suboptimal solution.

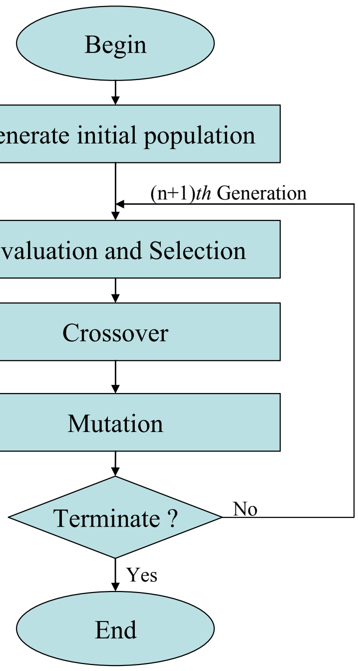


Fig. 2.4: Main flow of GA

***Composition and flow of GA*:**

- *Generating the initial population*:

In initial stage of calculation, GA generates chromosomes at random until it fills the population size. We need to decide the population size beforehand. Generally, the population size keeps constant size during GA processing. In basic GA, a bit sequence of binary number expresses the chromosome. However, in recent years, the mainstream approach is to use the order-phenotype for chromosome design. In order-phenotype GA, chromosomes express tree structures or delivery patterns. This research also adapts order-phenotype coding.

- *Evaluation and Selection*:

In GA, in order to keep the population size constant, it is necessary to screen several individuals from the group that consists of current population and offspring. Therefore, evaluation process determines which individual should survive. This process calculates fitness value of each individual using a fitness function, and individuals with low fitness value are screened (deleted).

After that, GA performs selection process for the crossover. GA chose parent-chromosomes stochastically by roulette wheel strategy. This strategy selects the parent-chromosomes using proportional rate based on fitness value. Moreover, these parents obtain an opportunity of crossover. In order to keep superior chromosome information to next generation, chromosomes with high fitness have high selection probability. On the other hand, chromosomes with low fitness have low selection probability.

- *Crossover*:

  Crossover is principal element of genetic operator. A typical method is one-point crossover. To generate new offspring, a part of two parent-chromosomes is exchanged.

- *Mutation*:

  Mutation changes gene in chromosomes. This operator selects one gene at random, and changes the gene into other value. After this operation, population can get new feature that parent's generation did not have. As a result, diversity appears in current generation.

- *End conditions*:

  When generation reaches the maximum generation number, iteration of evolution is finished. We can also set up the end condition according to problems.

2.4 Summary

This chapter described conventional TP model and fundamental structure of GA. The traditional TP model has simple structure and only calculates the cost according to shipment amount. When we consider the total optimization of logistics network, we have to consider various phenomena, such as the network structure, operation cost, inventory cost of each facility, and a transportation unit etc. To solve these problems, we propose improving models from Chapter 3.

On the other hand, recently, GA can apply to various combinatorial optimization problems. However, GA falls into a local minimum in some cases, and it needs much computation time to obtain the solution. Consideration and improvement of weak point of GA are also the aims of this research. Furthermore, according to minor difference of problem, development of new GA is inefficient. Therefore, we try to design a robust GA that can respond to structural change of TP flexibly.

Chapter 3. Multi-product Two-stage Logistics Network Model with Exclusionary Side Constraints

3.1 Introduction

A logistics networks design is one of the important elements of supply chain management (SCM). It aims optimization of whole supply chain in long-term efficiency. With the globalization of market and spread of the Internet, logistics networks have become more diverse. Usually, TP is used to design logistics network model. To minimize the total shipment cost, TP finds optimal routes from suppliers to demand places. However, when we adapt to real-world problems, it needs to extend or modify traditional TP model according to specific constraints.

Genetic algorithms (GAs) are solution search methods inspired by biological evolution [27], [28]. Recently GA has attracted attention as comparatively fast and simple methods to solve nonlinear problems. GA has been applied to a variety of fields, including TP. However, GA has some weak points that are high computational load, premature convergence to local minimum, and complex setting of parameters (crossover rate and mutation rate)

In this chapter, we propose a multi-product two-stage logistics network model with exclusionary side constraints and an efficient GA-based solution method. The proposed model is different from the traditional TP. For example, the network model is multi-staged, and it treats multi-item products. In addition, proposed TP has exclusionary side constraints. Such an inevitably realistic model becomes nonlinear. Furthermore, effective GA for exclusionary side constraints unreported previously.

To solve this problem, we apply a hybrid priority-based genetic algorithm (h-priGA) using priority-based encoding method. This h-priGA can overcome the drawbacks of GAs such as a complicated parameter setting and premastered conferencing to local solutions. In addition, the Fuzzy Logic Controller (FLC) incorporated in the h-priGA. FLC provides auto-tuning of parameters during the evolution, and maintains the diversity of solutions [29]. In this chapter, we also describe how to satisfy the constraints on the proposal TP model by utilizing GA features.

3.2 Traditional TP model and solution method

The objective of TP is to determine the delivery routes with lowest delivery cost. However, when we apply the basic TP model to real world, following problems occur.

(1) Traditional TP models consider only one-stage (from DCs to customers).

(2) Traditional TP assume all products are same kind.

(3) Traditional TP only considers the delivery cost in every delivery stage.

Other issues such as weather or traffic condition, and drivers' physical condition are not considered.

The real logistics networks can be interpreted as a multi-stage TP model [18] [19] [20]. When we consider three elements (plants, DCs, customers), these elements constitute two-stage TP model. In this case, the first stage corresponds to the delivery zone from plants to DCs and the second stage is the delivery zone from DCs to customer. As an example of additional constraints, Sun (1998) proposed transportation problem with exclusionary side constraints (escTP). This escTP model should satisfy the additional constraints, which prohibits simultaneous deliveries from the specific plant pair. Sun solved this problem by using Tabu Search and a branch-and-bound method [15], [16].

In this chapter, we propose a multi-product two-stage transportation model with exclusionary side constraints. We extend traditional escTP model for real-world problems. The escTP model described previously is a nonlinear model. Cao (1992) showed solution by CPLEX in a small-scale problem [31], however, it point out that much computation time is required in large-scale problem. On the other hand, although Cao, Uebe and others (1995) proposed solution method by Taboo Search [32] and Sun (2002) proposed solution by branch and bound method [16].

However, effective GA for escTP did not proposed previously. It is difficult to solve escTP, because GA method expresses the solutions as chromosomes. Sharif and Gen (2003) applied a spanning tree-based GA (st-GA) to this TP. The st-GA was developed for basic TP, and it uses the Prüfer code for chromosome representation (Prüfer number-based representation) [17]. Prüfer code can express a tree with n vertices by n - 2 sequences, and it can save the memory for computation. However, when we consider the following reasons, Prüfer code is inefficient technique [33]. The authors [17] developed feasibility criteria for Prüfer code, but Prüfer coding criterion is complex. In addition, Prüfer-number based representation easily generates infeasible chromosomes after crossover. However, they did not introduce any procedure for handling infeasible chromosomes. The authors did not mention how to satisfy the exclusionary side constraints.

Meanwhile, in literature [30], the authors propose the improved st-GA. However, the repair mechanisms for infeasible solution were only added. In numerical experiments, treated network models were only one-stage model. If the difference between the number of source nodes and the number of demand nodes is very big, it is difficult to generate the feasible chromosome. It only works when the number of source nodes and demand nodes are almost the same. In the case of multi-stage network like our proposal model, it is more difficult to adapt Prüfer number-based representation. Even if it actually calculates, we need much computation time to obtain solutions.

Therefore, in this chapter, in order to satisfy the exclusionary side constraints easily, we propose improved priority-based representation besides a network model design.

3.3 Proposal TP model

The proposed logistics network model is extended two-stage transportation problem (see Fig. 3.1). In this model, we assumed that plants produce several kinds of products, and their delivery routes are different correspond to product kind.

In the real world, plants do not produce only one kind of product. In response, delivery routes are different by product kind and usage. In proposed model, we consider the constraints on delivery routes not only plant-DC part but also DC-customer part. This means that customers are assigned to DC according to a product kind.

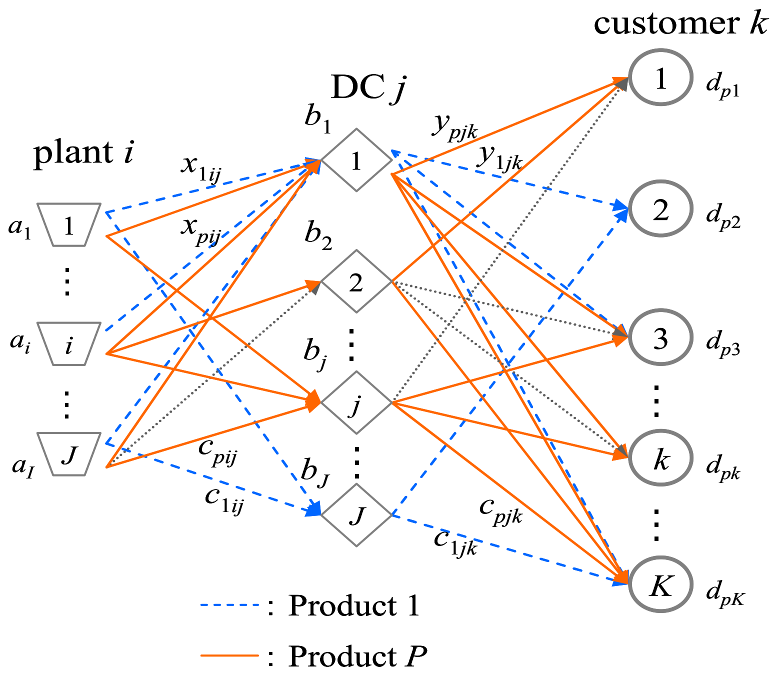


Fig. 3.1: Proposal network model

***Indices*:**

a_i : capacity of i-th plant

b_j : capacity *of* j-th DC

d_j : customer demand for product p

p : product kind

***Parameters*:**

c_{pij} : delivery cost of product p from i-th plant to j-th DC

c_{pjk} : delivery cost of product p from j-th DC to k-th customer

g_j : operation cost *of* j-th DC

t_{pij} : 1 if this plant-DC route is used; 0 otherwise

t_{pij} : 1 if this DC-customer route is used; 0 otherwise

***Decision variables*:**

x_{pij} : delivery amounts of product p from i-th plant to j-th DC

y_{pjk} : delivery amounts of product p from j-th DC to k-th customer

z_j : 1 if j-th DC is used, 0 otherwise

***Mathematical model*:**

$$\min \quad z(x) = \sum_{p=1}^{P}\left(\sum_{i=1}^{I}\sum_{j=1}^{J} t_{pij} c_{pij} x_{pij} + \sum_{j=1}^{J}\sum_{k=1}^{K} t_{pjk} c_{plk} y_{pjk}\right) + \sum_{j=1}^{J} g_j z_j \tag{3.1}$$

$$\text{s. t.} \quad \sum_{J=1}^{J} t_{pij} x_{pij} \le a_i, \quad \forall i, p \tag{3.2}$$

$$\sum_{p=1}^{P}\sum_{k=1}^{K} t_{pjk} y_{pjk} \le b_j z_j, \quad \forall j \tag{3.3}$$

$$\sum_{j=1}^{J} t_{pij} y_{pjk} \ge d_{pk}, \quad \forall k, p \tag{3.4}$$

$$\sum_{p=1}^{P}\sum_{i=1}^{I} t_{pij} x_{pij} = \sum_{p=1}^{P}\sum_{k=1}^{K} t_{pjk} y_{pjk}, \quad \forall j \tag{3.5}$$

$$x_{pij} \ge 0, \quad \forall p, i, j \tag{3.6}$$

$$y_{pjk} \ge 0, \quad \forall p, j, k \tag{3.7}$$

$$z_j = \{0,1\} \quad \forall j \tag{3.8}$$

Equation (3.1) expresses the minimization of total cost (delivery cost and operation cost). In this model, we include the operation cost of DCs. Equation (3.2) and (3.3) express the constraints on plant capacity and DC capacity, respectively. Equation (3.4) represents satisfaction of customer demand, and (3.5) is the constraint on demand-supply balance.

3.4 Priority-based representation and its improvement

In this chapter, we improve priority-based encoding method. This method proposed by Gen and Cheng (2000) [28]. This technique searches the delivery routes with low cost preferentially. However, there are the following problems in traditional GA.

- ***High calculation load*:** To evaluate individuals, GA repeats calculation for each fitness value to the number of population size.
- ***Premature convergence to local solutions*:** When some individuals have very high fitness values, such genes rapidly spread over the population. At the same time, the population loses diversity, and algorithm may converge to local minimum.
- ***Complicated parameter setting*:** The GA involves various parameters such as population size, crossover rate and mutation rate. Although we decide these parameters empirically, they give strong affects to searching ability. Furthermore, optimal parameter settings change with problem characteristic. Therefore, to decide optimal parameters, we need many preliminary experiments.

To deal with the problems, we use the hybrid priority-based genetic algorithm (h-priGA). In order to improve efficiency, Fuzzy Logic Controller (FLC) is included. FLC is auto-tuning mechanism for crossover rate and mutation rate.

3.4.1 Chromosome representation

In this section, we describe the priority-based encoding method. Additionally, we describe improved method for escTP. Fig. 3.2 shows an example of chromosome and Fig. 3.3 shows the decoding procedure. The gene values express the priority order of plants and DCs. Also, the length of the chromosome is equal to the total number of plants (|*I*|) and DCs (|*J*|). The gene numbering corresponds to the node ID; the vertical and horizontal axes of the cost matrix respectively correspond to plant ID and DC ID. When we generate chromosomes, the natural number from 1 to total number of nodes are inserted into each gene in ascending order, and then they are shuffled randomly.

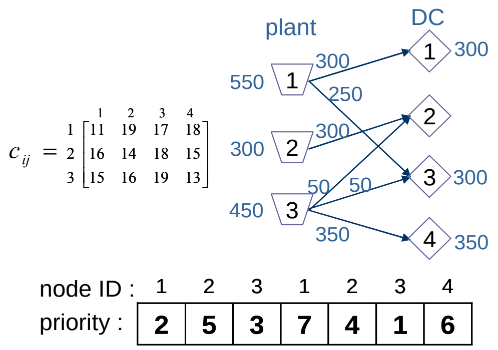


Fig. 3.2: Sample of chromosome

In decoding process, gene with highest value has the highest priority, and decoding mechanism select the lowest-cost route with reference to cost matrix. For example, DC1 has the highest priority value 7, and cost matrix indicates that route Plant1 - DC1 has the lowest cost 11. Therefore, the arc connects plant1 and DC1 and the respective capacities are updated. After that, decoding mechanism select DC4 has next priority 6, the arc connects DC4 and Plant3, and the capacities are updated. The decoding mechanism repeats same process until all demands are satisfied.

The DCs capacity must be sufficient to meet the customers' demands. Therefore, decoding is started from part which correspond to second stage. In addition, after deciding spanning tree of second stage, decoding mechanism obtain DCs operated for first stages. Fig. 3.4 shows an example of network model for one product, and the corresponding chromosome. The dashes in the cost matrix indicate prohibited delivery routes (i.e. exclusionary side constraints).

procedure 1 : Decoding of the chromosome for transportation tree
Input : I : set of plant, J : set of DCs,
b_j : demand on DC j, $\forall j \in J$,
a_i : capacity of plant i, $\forall k \in I$,
c_{ij} : transportation cost of one unit of product from plant i to DC j, $\forall i \in I, \forall j \in J$,
$v(i+j)$: chromosome, $\forall i \in I$, $\forall j \in J$,
output : g_{ij} : the amount of product shipped from plant i to DC j
step 1. $g_{ij} \leftarrow 0$, $\forall i \in I$, $\forall j \in J$,
step 2. $z \leftarrow \text{argmax}\{v(t),\ t \in |I|+|J|\}$; select a node
step 3. **if** $z \in I$, **then** $i^* \leftarrow z$; select a plant
$j^* \leftarrow \text{argmin}\{c_{ij} \mid v(j) \neq 0, j \in J\}$; select a DC with the lowest cost
else $j^* \leftarrow z$; select a DC
$z^* \leftarrow \text{argmin}\{c_{ij} \mid v(j) \neq 0, i \in I\}$;select a plant with the lowest cost
step 4. $g_{i^*j^*} \leftarrow \min\{a_{i^*}, b_{j^*}\}$; assign available amount of units
Update availabilities on plant (i^*) and DC (j^*)
$a_{i^*} = a_{i^*} - g_{i^*j^*}$ $\quad$ $b_{j^*} = b_{j^*} - g_{i^*j^*}$
step 5. **if** $a_{i^*} = 0$ **then** $v(i^*) = 0$
if $b_{j^*} = 0$ **then** $v(j^*) = 0$
step 6. **if** $v(|I|+j) = 0$, $\forall j \in J$, *then* calculate transportation cost and return,
else goto step 1.

Fig. 3.3: Decoding procedure

The proposed TP includes exclusionary side constraints. Therefore, we propose the effective method to satisfy exclusionary side constraints. This technique does not need special check routines for infeasible solutions, so it can reduce computation time. First, we consider whole network as ordinary two-stage transportation model. Next, we assign immeasurable costs to prohibited routes. These prohibitive costs correspond to the dashes in cost matrix. This is equivalent to reproduction of network model with restricted delivery routes.

As explained above, decoding process of priority-based representation chooses low cost route preferentially. Therefore, the decoding process is not chooses very high cost routes. On the other hand, if decoding process selects route with immeasurable costs, selection process treats it as an obviously poor solution.

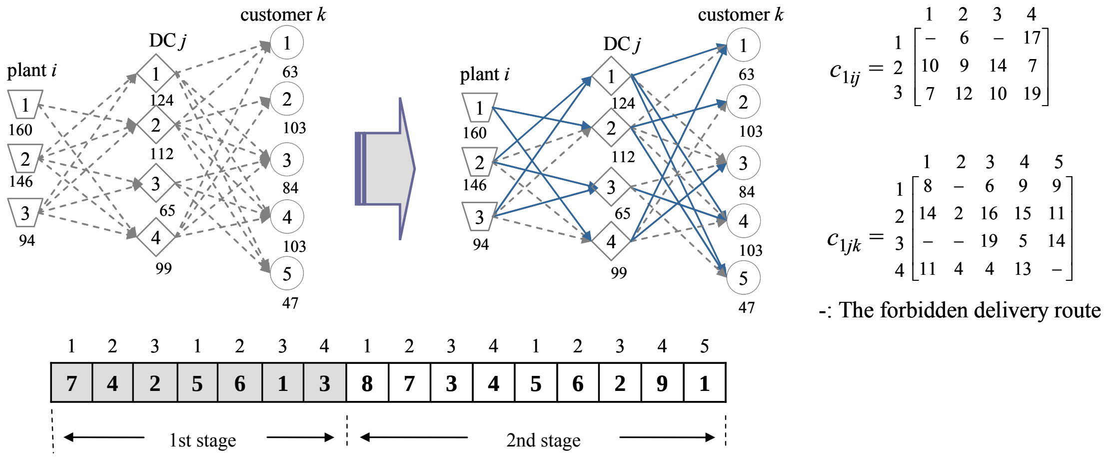


Fig. 3.4: A sample of transportation tree and chromosome

3.4.2 Crossover

Crossover is performed to search new solution space. In this research, we adopt Weight Mapping Crossover (WMX) [18]. This method is extension of one-point crossover. The parents inherit priority order of crossover portion to offspring. In one offspring, there are no same priority orders in right side and left side of cut point.

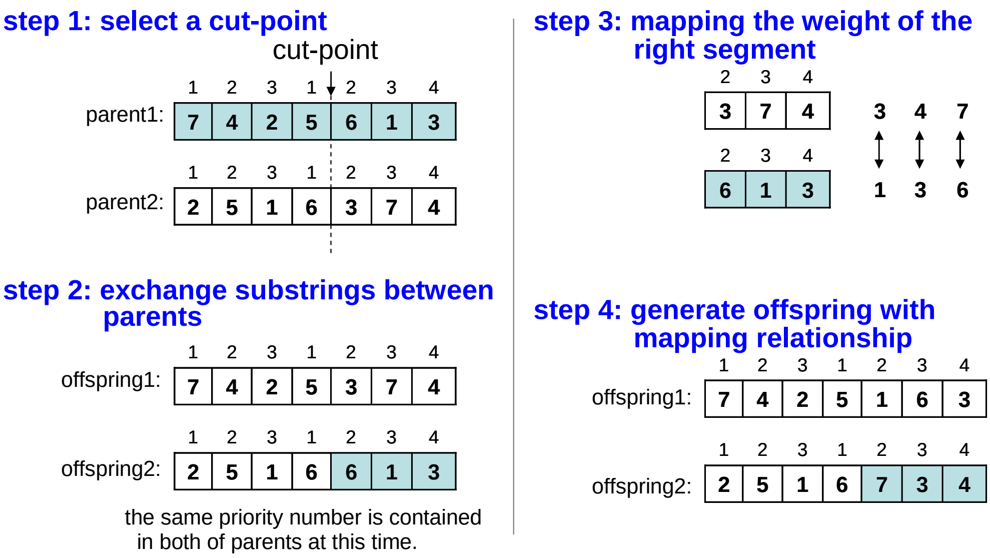


Fig. 3.5: Illustration of WMX

3.4.3 Mutation

Mutation is a genetic operation to maintain the diversity of population by changing some genes. In this chapter, we adopt a swap mutation. This operator selects a pair of genes randomly and exchanges them. We apply this mutation at first and second stages for diversity.

3.4.4 Evaluation and selection

The evaluation function is inverse of objective function. Moreover, roulette strategy is adapted for selection method.

3.5 Hybrid priority-based genetic algorithm (h-priGA)

In this chapter, we also prepare priGA with Fuzzy Logic Controller (FLC) [27] [28] [29]. This algorithm is called hybrid priority-based GA (h-priGA). FLC is the auto-tuning method for the crossover rate and mutation rate. We also apply improved priority-based encoding to h-priGA. In numerical experiments, we check the effect of FLC. We show the process flow of h-priGA in Fig. 3.6. Moreover, the composition of processing is described below.

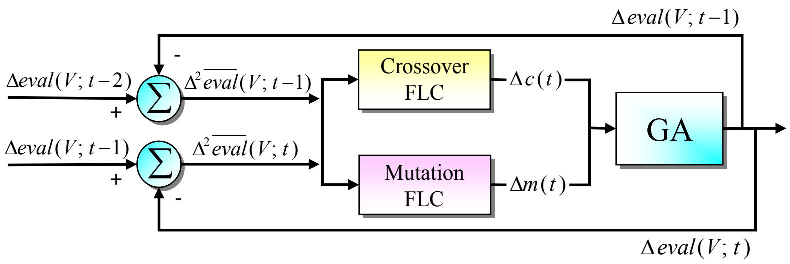


Fig. 3.6: Illustration of FLC

Fuzzy Logic Controller procedure

Step 1**:** Using the following expression, the average evaluation for the previous population ($\overline{eval}(v;\ t\text{-}1)$) and the current population ($\overline{eval}(v;\ t)$) are calculated:

$$\overline{eval}(\upsilon;t)=\frac{\sum_{k=1}^{popSize} eval(\upsilon_k;t)}{popSize}-\frac{\sum_{k=popSize+1}^{popSize+offSize} eval(\upsilon_k;t)}{offSize}$$

Step 2**:** Using the fuzzy decision table (Table 3.1), the control actions for variations of the crossover probability $\Delta c(t)$ and mutation probability $\Delta m(t)$ for the above values ($\overline{eval}(v;\ t\text{-}1)$), ($\overline{eval}(v;\ t)$) are determined.

- If $0<\overline{eval}^2(v;t)\le\varepsilon$, then the crossover probability and mutation probability are increased in the next generation. Here ε is a small positive real number.
- If $-\varepsilon\le\overline{eval}^2(v;t)<0$, then the crossover probability and mutation probability are decreased in the next generation.
- If the variation of the average evaluation is small, then the crossover probability and the mutation probability are increased sharply in the next generation.

Table 3.1: Fuzzy decision table

$\Delta c(t)$		$\Delta\overline{eval}(t-1)$								
		NR	NL	NM	NS	ZE	PS	PM	PL	PR
$\Delta\overline{eval}(t)$	NR	NR	NL	NL	NM	NM	NS	NS	ZE	ZE
	NL	NL	NL	NM	NM	NS	NS	ZE	ZE	PS
	NM	NL	NM	NM	NS	NS	ZE	ZE	PS	PS
	NS	NM	NM	NS	NS	ZE	ZE	PS	PS	PM
	ZE	NM	NS	NS	ZE	PM	PS	PS	PM	PM
	PS	NS	NS	ZE	ZE	PS	PS	PM	PM	PL
	PM	NS	ZE	ZE	PS	PS	PM	PM	PL	PL
	PL	ZE	ZE	PS	PS	PM	PM	PL	PL	PR
	PR	ZE	PS	PS	PM	PM	PL	PL	PR	PR

NR (Negative Larger); PS (Positive Small);
NL (Negative Large); PM (Positive Medium);
NM (Negative Medium); PL (Positive Large);
NS (Negative Small); PR (Positive Larger);
ZE (Zero)

***Step 3*:** Normalized values are decided in advance as Table 3.2 (look-up table), and the variations of the crossover probability and mutation probability are calculated as follows:

$$\Delta c(t) = \gamma_1 \cdot z(i, j) \quad , \quad \Delta m(t) = \gamma_2 \cdot z(i, j)$$

Here $z(i, j)$ are elements in Table 3.2 that express the normalized values of the control action for the crossover probability and mutation probability, and γ_1 and γ_2 are accommodation coefficients.

Table 3.2: Look-up table

$z(i,j)$		i								
		-4	-3	-2	-1	0	1	2	3	4
j	-4	-4	-3	-3	-2	-2	-1	-1	0	0
	-3	-3	-3	-2	-2	-1	-1	0	0	1
	-2	-3	-2	-2	-1	-1	0	0	1	1
	-1	-2	-2	-1	-1	0	0	1	1	2
	0	-2	-1	-1	0	2	1	1	2	2
	1	-1	-1	0	0	1	1	2	2	3
	2	-1	0	0	1	1	2	2	3	3
	3	0	0	1	1	2	2	3	3	4
	4	0	1	1	2	2	3	3	4	4

Step 4**:** The crossover probability and mutation probability are modified as follows:

$$p_C(t+1) = p_C(t) + \Delta c(t),\ p_M(t+1) = p_M(t) + \Delta m(t)$$

Here $p_c(t)$ and $p_m(t)$ are, respectively, the crossover probability and mutation probability for the next generation t.

Step 5**:** The procedure returns to the GA loop.

3.6 Numerical experiments

In this section, we perform comparison experiment using proposed TP. Following four GAs are used for comparison.

- priGA with PMX (priGA-PMX)
- priGA with WMX (priGA-WMX)
- h-priGA with PMX (h-priGA-PMX)
- h-priGA with WMX (h-priGA-WMX)

These four GAs were prepared in order to investigate the effect of FLC and WMX for improved priority-based encoding. We show the test data in Table 3.3. We prepared four problems with number of plants from 4 to 15, and number of DCs and customers from 5 to 25 and 15 to 50, respectively. We generate delivery cost and operation cost of DCs randomly. The plants treat two types of products. Moreover, the prohibited routes in each stage were randomly generated 15% or less of total number of nodes.

Table 3.3: Test problems size

problem no.	No. of plants (i)	No. of DC (j)	No. of customers(k)	No. of constraints	No. of variables
1	4	5	15	48	195
2	8	10	20	76	570
3	10	20	40	140	2020
4	15	25	50	175	3275

The initial population size was set to 100. For non-hybrid algorithm, we prepared three kinds of crossover rate (p_c) and mutation rate (p_m). The h-priGA is not dependent on initial value of both parameters. Each problem was calculated 30 times respectively.

We show the calculation results in Table 3.4. Four GAs that applied improved priority-based encoding were able to calculate solutions. The table presents p_c, p_m, the best solutions in 30 runs (Best), the average solutions in 30 runs (AVG), standard deviation (SD) and average computing time (ACT). As a result, h-priGA-WMX was best method for all test problems. In particular, h-priGA-WMX outperformed two methods with initial parameters (priGA-WMX, priGA-PMX). These results proved the validity of FLC. In addition, WMX excelled PMX in solution quality and processing time. The reason is the repair mechanism of chromosomes (infeasible solutions) after

PMX. On the other hand, FLC was very effective to reduce the computation time. Furthermore, when we compare h-priGA-WMX and priGA-PMX(0.5, 0.3) in problem 1, h-priGA-WMX reduced Best and ACT(s) about 7.2%, 76% respectively. Fig. 3.7 shows the evolutionary process of all methods. We can see that h-priGA evolve rapidly to find the best solution.

Table 3.4: Comparison result

Problem No.		priGA -PMX			priGA -WMX			h-priGA -PMX	h-priGA -WMX
		(p_C, p_M)			(p_C, p_M)			-	-
		(03, 0.1)	(0.5, 0.3)	(0.7, 0.5)	(03, 0.1)	(0.5, 0.3)	(0.7, 0.5)	-	-
1	Best	8674.01	8674.02	8656.00	8270.00	8270.00	8062.03	8070.00	8046.27
	AVG	8784.40	8769.00	8778.47	8272.53	8271.73	8070.67	8077.00	8065.93
	SD	40.86	38.37	35.90	12.62	12.27	13.98	15.25	10.35
	ACT(s)	7.31	8.06	9.30	2.91	4.82	6.81	1.95	1.93
2	Best	11668.30	11668.00	11668.00	11644.00	11644.00	11644.00	11668.00	11624.10
	AVG	11745.13	11728.87	11711.07	11689.67	11679.00	11673.07	11733.60	11664.27
	SD	33.07	28.68	30.61	22.93	24.61	25.07	23.26	18.80
	ACT(s)	8.72	9.45	10.38	4.64	8.07	10.05	2.98	2.80
3	Best	10950.00	10968.03	11028.23	10850.00	10856.20	10806.21	10898.22	10806.21
	AVG	11971.33	11143.40	11080.47	10967.20	10911.33	10903.53	11064.07	10902.01
	SD	46.91	48.27	53.14	44.28	47.37	43.32	50.39	42.02
	ACT(s)	9.31	17.21	25.10	8.95	17.17	23.77	5.24	5.11
4	Best	12652.71	12697.00	12660.03	12612.00	12642.40	12654.80	12643.50	12630.50
	AVG	12753.67	12720.93	13708.80	12729.60	12715.67	12695.93	12768.94	12681.13
	SD	38.91	37.15	35.46	37.71	31.35	31.26	27.20	21.92
	ACT(s)	14.98	26.38	37.80	13.35	25.31	35.88	8.21	7.12

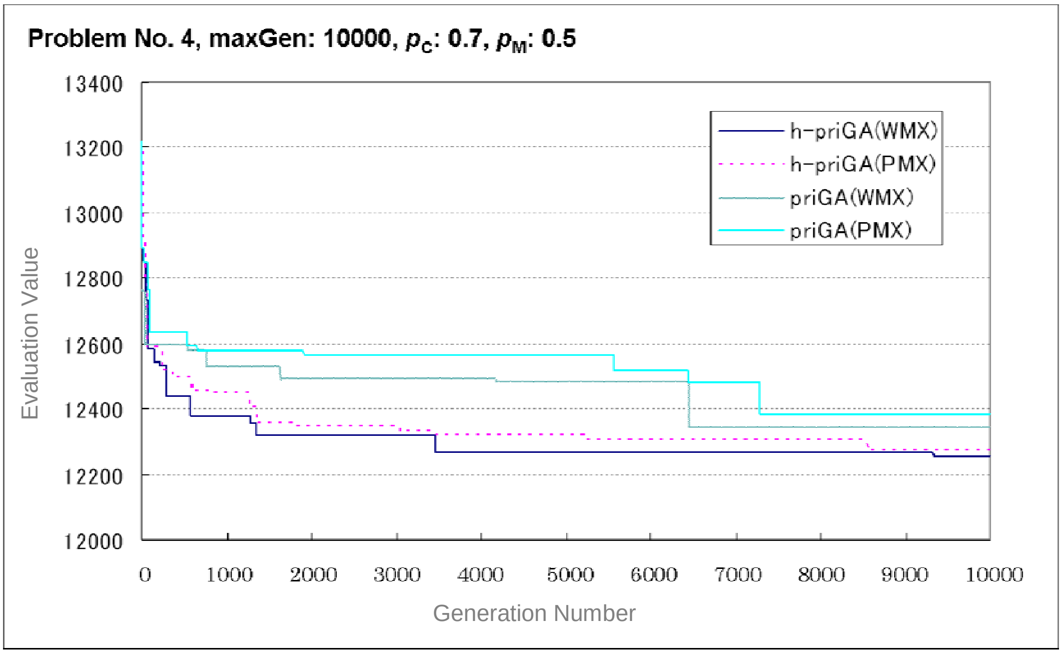


Fig. 3.7: The evolution process in both methods

3.7 Summary

In this chapter, we proposed multi-product two-stage transportation model with exclusionary side constraints. This model considers the drawbacks of traditional TP and real-world conditions. In proposed TP, plants produce several kinds of items, and the delivery routes and transportation costs vary with the products. Since GA expresses the solutions as chromosomes, it was difficult to solve this kind of model. Therefore, in order to calculate by GA and to satisfy exclusionary side constraints easily, we also present improved priority-based representation. This technique has no necessary check routine of infeasible solution. In addition, it can reduce the computation time simultaneously.

In numerical experiments, we compared four GAs. As a result, it became clear that proposal TP model could solve by four GA with improved priority-based representation. Moreover, h-GA reduced computation time sharply and achieved good results.

However, proposed TP model did not cover inventory concept. Furthermore, in recent years, real delivery routes are becoming more complicate. These problems are future research, so we propose new TP model in following chapter.

Chapter 4. Two-stage Transportation Problem with inventory and exclusionary side constraints (esc-2ITP)

4.1 Introduction [34]-[39]

In Chapter 3, we proposed multi-product two-stage transportation model with exclusionary side constraints. In addition, in order to satisfy exclusionary side constraints easily, we also proposed improved traditional priority-based representation. However, proposed TP model in Chapter 3 did not consider several important subjects. For example, these are concept of inventory and time. At the same time, we only showed that problem solving at four types of priGAs with proposed representation method. We did not overcome the weak point of GA.

In this chapter, we propose a two-stage transportation problem with inventory and exclusionary side constraints (esc-2ITP). The esc-2ITP is extended TP proposed in Chapter 3. This model designs delivery plan for multiple periods and demands of DCs are satisfied for each term. For numerical experiments, each plant decides next production quantity according to ordering quantity of DC. This ordering quantity calculated from a simple demand-forecasting equation.

4.2 Formulation of esc-2ITP model

When we try to sell certain goods, it is necessary to keep a certain quantity until they sold. If there are no goods to satisfy consumer demands, the opportunity of business is lost. However, if storage space for inventory is required, inventory cost occurs. Therefore, inventory costs and we must keep inventory amounts as small as possible without inventory shortage. Moreover, a concept of time factor is needed to calculate a carrying cost in certain time. Further, when the company does not have enough inventories, service level may decrease. It is necessary to keep proper inventory and service level. Although the company has introduced the various demand-forecasting techniques, there is no positive technique. The main reasons are short product life cycle (PLC) and diversification of demand.

The esc-2ITP designs a distribution network to satisfy customer demands at minimum cost according to capacity of plants and the DCs. Fig. 4.1 shows the sample model of esc-2ITP. The esc-ITP has similar form with TP proposed in Chapter 3. However, in esc-2ITP, inventory and a carrying cost are generated in each plants and DCs. Moreover, if products are manufactured, management cost occurs in plants.

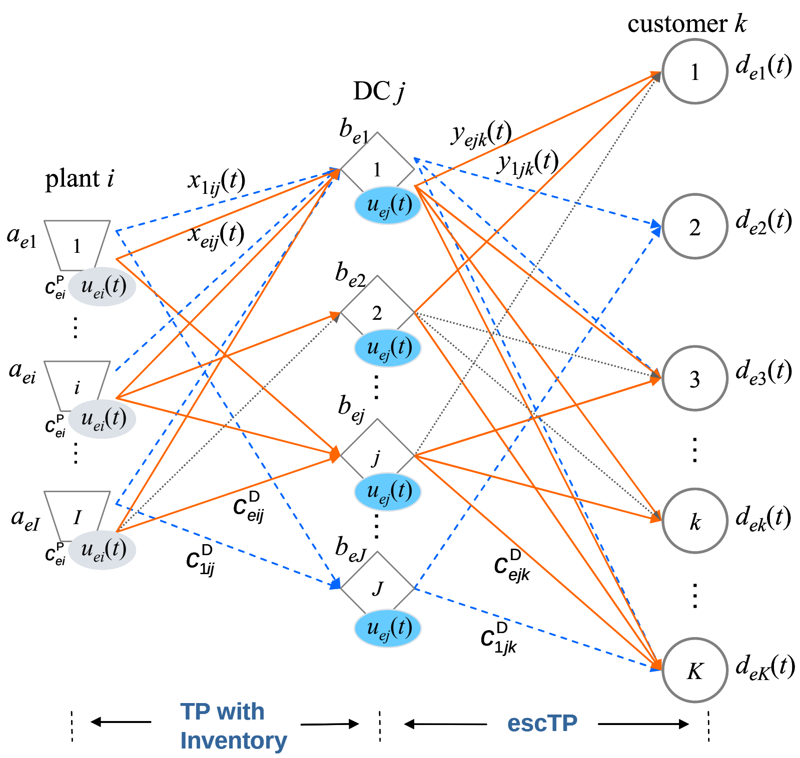


Fig. 4.1: Sample of esc-2ITP

***Indices*:**

i: plant, i=1, 2,…, I

j: DC, j=1, 2,…,l ,…,J

k: customer, k=1, 2,…, K

t: time period, t=1, 2,…, T

e: product ID, e=1, 2,…, E

I: total number of plants

J: total number of DCs

K: total number of customers

E: total kind of products

T: total number of periods

***Parameters*:**

a_{ei} : capacity of plant i for product e

b_{ej} : capacity of DC j for product e

$d_{ek}(t)$: demands of customer k for product e in period t

$c^{\mathrm{D}}_{eij}(t)$: delivery cost for product e from plant i to DC j in period t

$c^{\mathrm{D}}_{ejk}(t)$: delivery cost for product e from DC j to customer k in period t

c^{P}_{ei} : manufacture cost of product e

c_{ei} : inventory cost of plant i

c_{ej} : inventory cost of DC j

***Decision variables*:**

$x_{eij}(t)$: shipment amount of product e from plant i to DC j in period t

$y_{ejk}(t)$: shipment amount of product e from DC j to customer k in period t

$p_{ei}(t)$: production quantity of plant i in period t

$u_{ei}(t)$: amount of inventories of the plant i in period t

$u_{ej}(t)$: amount of inventories of the DC j in period t

The first stage is formulated as TP with inventory. The second stage is formulated by escTP model. When we formulize, we assume the following assumption this research.

A1. Plant and DC have storage spaces for each product.

A2. Demand of customer can be filled with the capacity of the above-mentioned space.

A3. Customer demand is obtained exponential smoothing.

A4. Arrangement place of each facility is known.

A5. Delivery cost is known at each stage.

A6. Kind of product manufactured at each factory is known.

A7. Production is performed in an instant.

A8. Delivering from the plant to the customer is carried out immediately.

A9. This model draws up a delivery plan for multiple periods.

In this research, we did not consider the amount of maximum accumulation of plants and DCs. Also changing of the commodity value by progress of time does not consider. We formulate esc-2ITP as follows.

$$\min \quad z=\sum_{t=1}^{T}\sum_{e=1}^{E}\left(\sum_{i=1}^{I}\sum_{j=1}^{J}c_{eij}^{\mathrm{D}}x_{eij}(t)+\sum_{j=1}^{J}\sum_{k=1}^{K}c_{ejk}^{\mathrm{D}}y_{ejk}(t)+\sum_{i=1}^{I}c_{ei}u_{ei}(t)+\sum_{j=1}^{J}c_{ej}u_{ej}(t)+\sum_{i=1}^{I}c_{ei}^{\mathrm{P}}p_{ei}(t)\right) \tag{4.1}$$

$$\text{s. t.} \quad u_{ei}(t)=u_{ei}(t-1)+p_{ei}(t)-\sum_{j=1}^{J}x_{eij}(t),\ \forall t,e,i \tag{4.2}$$

$$u_{ej}(t)=u_{ei}(t-1)+\sum_{i=1}^{I}x_{eij}(t)-\sum_{k=1}^{K}x_{ejk}(t),\ \forall t,e,j \tag{4.3}$$

$$u_{ei}(t-1)+p_{ei}(t)<a_{ei},\ \forall t,e,i \tag{4.4}$$

$$u_{ej}(t-1)+\sum_{i=1}^{I}x_{eij}(t)\le b_{ej},\ \ \forall t,e,j \tag{4.5}$$

$$\sum_{j=1}^{J}y_{ejk}(t)\ge d_{ek}(t),\ \ \forall t,e,k \tag{4.6}$$

$$y_{ejk}y_{glk}=0 \quad \text{for } (e,g)\in P_k,\ (j,l)\in D_k,\ \forall k \tag{4.7}$$

$$x_{eij}(t),\ y_{ejk}(t)\ge 0,\quad \forall e,i,j,k,t \tag{4.8}$$

Equation (4.1) is objective function for minimizing total cost. Constraints (4.2) and (4.3) show the inventory in the plant and that in the DC respectively. Constraints (4.4) and (4.5) represent the capacity constraints of plants and DCs. Constraint (4.6) ensures that all demands in period t are satisfied. Constraint (4.7) represents the extended exclusionary side constraint. This shows that two different DCs, j and l, are not allowed to serve customer k simultaneously. P_k= {(e, g)| product e and g cannot be simultaneously shipped to customer k}. In the same way, D_k = {(j, l)| products from DC j and l cannot be simultaneously shipped to customer k}.

Moreover, we decide production quantity in the current period from equation (4.9) (exponential smoothing). For numerical simulation, we use this equation to determine the next order quantity of DCs. Exponential smoothing is the representative time-series analysis technique. Usually, this technique uses time-series data to forecast a value in the future. In weighting past data, high value is placed on newer data. In one of the weighted average methods, a low weight is assigned to past data (exponential decrease), and the moving average is calculated. To decide the degree of influence of the past forecast value, coefficient α (the smoothing factor) is set in a range of 0< α <1. This is simulated by accumulating the past data. Usually α is set so that the forecast value and error of result are minimized.

Predicted value = α × *Last track record value*＋(1 －α) × *The last predicted value*

$$y^{\mathrm{F}}_{ejk}(t) = \alpha y_{ejk}(t-1) + (1-\alpha) y^{\mathrm{F}}_{ejk}(t-1) \qquad (4.9)$$

In this case, $y^{\mathrm{F}}_{ejk}(t)$ is the demand forecast value in period t. Then subtract current inventory amount from $y^{\mathrm{F}}_{ejk}(t)$, we can get next order quantity. We assumed the total volume of production in this period.

4.3 Summary

In this chapter, we proposed ecs-2ITP model as extended TP proposed in Chapter 3. This model includes the concepts of proper inventory and various costs generated on logistics network. Simultaneously, ecs-2ITP designs the delivery plan for multiple periods. In Chapter 4, we describe the algorithm that improves weak points of GA. The development of solution algorithm for ecs-2ITP is next subject.

Chapter 5. Boltzmann random key-based GA (Brk-GA)

5.1 Introduction

In this chapter, we develop a solution algorithm for esc-2ITP. The solution method for Chapter 3 was improved only chromosome representation. Thus, the weak points of GA were not improved. Therefore, we design a new GA approach which consider the search mechanism of GA synthetically. This approach is a Boltzmann random key-based GA (Brk-GA). Since this algorithm has a simple structure, the computation time can be reduced. In addition, we incorporate Boltzmann selection mechanism [40]. In classic GA, it is known that Boltzmann selection can decrease a probability of premature convergence to local minimum.

The following literatures are examples of improved Boltzmann selection and example that applied Boltzmann selection. Mori, et al proposed Thermodynamic Genetic Algorithm (TDGA) which explicitly consider the diversity of population [41]. TDGA estimates diversity of population clearly as entropy. In addition, this GA forms population to minimize free energy that harmonized adaptive value and entropy using temperature parameter. This technique is effective for combination optimization problem. Further, since TDGA maintains the diversity, it has a possibility of applicable to multi-objective optimization problem. In the literature, TDGA applies to a knapsack problem in evaluation experiment. The effect for TP is not yet proved. Ambedkar, et al [42] proposed a Cauchy annealing schedule for Boltzmann selection scheme. This selection schema is based on a hypothesis that selection-strength should increase as evolutionary process goes on. At the same time, distance between two selection strengths should decrease for the process to converge. In addition, they developed formalism for selection mechanisms using fitness distributions and give an appropriate measure for selection-strength. In numerical experiments, they compare Boltzmann selection with Cauchy annealing schedule and traditional Boltzmann selection. Both techniques were applied to several variable functions. However, authors did not describe the computation time. It seems that computation time is needed, because proposed technique is complicated. We need flexible GA which can respond in an instant for the sequential change of road environment.

Meanwhile, classical GA uses a bit vector for chromosome expression. On the other side, st-GA and priGA are order-phenotype GA. In order-phenotype GA, these chromosomes express tree structures or delivery patterns. Many literatures that treat Boltzmann selection use classical GA [43]. This chapter verifies the effectiveness of the Boltzmann selection to order-phenotype GA. Moreover, it is necessary to compare with FLC examined in Chapter 3.

In this chapter, we show the performance assessment of Brk-GA using esc-2ITP proposed in Chapter 4.

5.2 Boltzmann random key-based GA (Brk-GA)

As discussed in Chapter 3, traditional GA has several problems. These are a high calculation load, precocious convergence to local minimum, and complexity of parameter setting. Premature convergence to a local minimum is related with the width of search space. If it falls into local minimum in early stages of search, wide range search does not perform. Although GA scatters many samples called an initial population to search space, it does not escape from local minimum.

In addition, the features of order-phenotype GA are compact expression of solutions and small calculation cost of coding. However, in order-phenotype GA, crossover operator changes the feature of a chromosome (solution) a lot. Therefore, if high crossover rate is set up, it does not only bar convergence of search, but it will increase computation time. Furthermore, the roulette strategy tends to fall into a local minimum. Simultaneously, we consider that an elite strategy is a risky method, which it is easy to run into a local minimum.

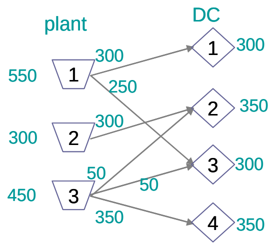
Therefore, in order to cope with these problems, we design Brk-GA. This approach has several features such as a random key-based encoding method, simple structure and Boltzmann roulette selection. Brk-GA allows robust search that consider the balance of exploration and exportation. This technique also aims at shortening computing time and reducing precocious convergence by reducing the unevenness of solutions. Finally, numerical experiments with various scales of esc-2ITP show the effectiveness and the efficiency of Brk-GA.

5.2.1 Random key-based representation

In Brk-GA, we adopt random key-based encoding. This technique is advanced encoding method proposed in Chapter 3. This technique can shorten the processing time of intersection. The length of chromosome equals to total number of plants and DCs. the gene IDs are represent node IDs in the network. The value of genes represents their priority. These priority values are used in decoding to construct transportation tree. For esc-2ITP, we apply this encoding to each stage. While the first part represents the transportation tree between plants and DCs, the second part represents the transportation tree between DCs and customers.

Fig. 5.1 represents a transportation tree with three plants and four DCs, its cost matrix and chromosome. In this chapter, we generated a random number in the range of 0 to 1 to design the chromosomes.

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array}\begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 11 & 19 & 17 & 18 \\ 16 & 14 & 18 & 15 \\ 15 & 16 & 19 & 13 \end{bmatrix} \end{array}$$



node ID l :	1	2	3	1	2	3	4
priority $v(l)$:	0.216	0.464	0.331	0.714	0.381	0.185	0.538

Fig. 5.1: Sample of random key-based GA

Table 5.1: Trace table of decoding procedure

step	$v(i+j)$	a	b
0	[0.216 0.464 0.331 | 0.714 0.381 0.185 0.538]	(550, 300, 450)	(300, 350, 300, 350)
1	[0.216 0.464 0.331 | 0.000 0.381 0.185 0.538]	(250, 300, 450)	(0, 350, 300, 350)
2	[0.216 0.464 0.331 | 0.000 0.381 0.185 0.000]	(250, 300, 100)	(0, 350, 300, 0)
3	[0.216 0.000 0.331 | 0.000 0.381 0.185 0.000]	(250, 0, 100)	(0, 50, 300, 0)
4	[0.216 0.000 0.331 | 0.000 0.000 0.185 0.000]	(250, 0, 50)	(0, 0, 300, 0)
5	[0.216 0.000 0.000 | 0.000 0.000 0.185 0.000]	(250, 0, 0)	(0, 0, 250, 0)
6	[0.000 0.000 0.000 | 0.000 0.000 0.000 0.000]	(0, 0, 0)	(0, 0, 0, 0)

procedure 1 : Decoding of the chromosome for transportation tree

Input : I : set of plant, J : set of DCs,

b_j : demand on DC j, $\forall j \in J$,

a_i : capacity of plant i, $\forall k \in I$,

c_{ij} : transportation cost of one unit of product from plant i to DC j, $\forall i \in I, \forall j \in J$,

$v(i+j)$: chromosome, $\forall i \in I$, $\forall j \in J$,

output : g_{ij} : the amount of product shipped from plant i to DC j

step 1. $g_{ij} \leftarrow 0$, $\forall i \in I$, $\forall j \in J$,

step 2. $z \leftarrow \text{argmax}\{v(t),\ t \in |I|+|J|\}$; select a node

step 3. **if** $z \in I$, **then** $i^* \leftarrow z$; select a plant

$j^* \leftarrow \text{argmin}\{c_{ij} \mid v(j) \neq 0, j \in J\}$; select a DC with the lowest cost

else $j^* \leftarrow z$; select a DC

$z^* \leftarrow \text{argmin}\{c_{ij} \mid v(j) \neq 0, i \in I\}$;select a plant with the lowest cost

step 4. $g_{i^*j^*} \leftarrow \min\{a_{i^*}, b_{j^*}\}$; assign available amount of units

Update availabilities on plant (i^*) and DC (j^*)

$a_{i^*} = a_{i^*} - g_{i^*j^*}$ $\quad b_{j^*} = b_{j^*} - g_{i^*j^*}$

step 5. **if** $a_{i^*} = 0$ **then** $v(i^*) = 0$

if $b_{j^*} = 0$ **then** $v(j^*) = 0$

step 6. **if** $v(|I|+j) = 0$, $\forall j \in J$, *then* calculate transportation cost and return,

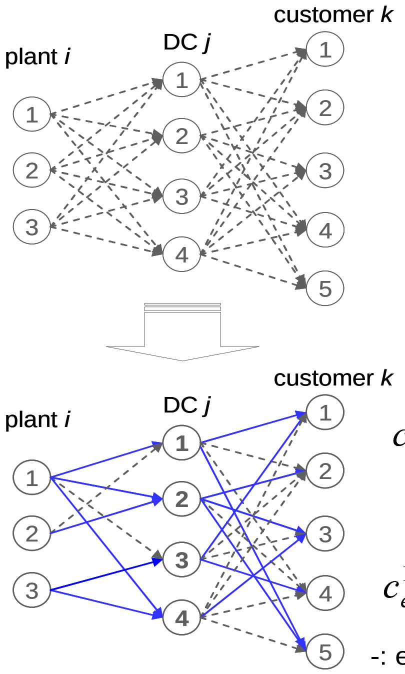
else goto step 1.

Fig. 5.2: Decoding procedure

This technique has a decoding mechanism like priority-based encoding developed by Gen & Cheng [27] [28]. Fig. 5.2 shows the decoding procedure, and its trace table is given in Table 5.1. As the trace table shows, in the first step, since DC 1 has the highest priority in the chromosome and the lowest cost is between Plant 1 and DC 1. Then, an arc between DC 1 and plant 1 is added to transportation tree. After determining the amount of shipment that is g_{11} = min{550, 300} = 300, the capacity of the plant and the demand of the DC are updated as a_1 = 550-300 = 250, b_1 = 300 – 300 = 0, respectively. Since b_1 = 0, the priority of DC 1 is set to 0, and DC 4 with the next-highest priority is selected. After adding an arc between DC 4 and plant 3, the amount of the shipment between them is determined and their capacity and demand are updated as explained above. This process repeats until the demands of all DCs are satisfied.

This representation uses a random number, so it has advantage for crossover compare with priority-based encoding. To keep priority order, priority-based encoding needed special crossover (WMX). We need chromosome representation which do not need special repair mechanism.

Fig. 5.3 is the sample network model and chromosome. The first part of the chromosome consists of seven digits; the second part consists of nine digits. The hyphen in a cost matrix is the route included exclusionary side constraints. Since decoding mechanism is same with improved priority-based representation, we can satisfy exclusionary side constraints using same way in Chapter 3. While random key-based representation has an advantage of improved priority-based representation, it can use simple crossover.



$$c^{D}_{eij} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 16 & 18 & 15 & 19 \\ 2 & 12 & 7 & 16 & 14 \\ 3 & 15 & 20 & 3 & 12 \end{array}$$

$$c^{D}_{ejk} = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 5 & 19 & - & 14 & 9 \\ 2 & - & 9 & 18 & 10 & 14 \\ 3 & 7 & 17 & 7 & 3 & - \\ 4 & 3 & 18 & 2 & 5 & 2 \end{array}$$

-: exclusionary side constraints are meant.

node ID l :	1	2	3	1	2	3	4	1	2	3	4	1	2	3	4	5
priority $v(l)$:	0.647	0.138	0.119	0.592	0.846	0.716	0.265	0.337	0.247	0.794	0.575	0.503	0.354	0.635	0.752	0.152
	1st stage							2nd stage								

Fig. 5.3: Sample of esc-2ITP model and chromosome

5.2.2 Genetic operators

The main characteristics of Brk-GA are as follows:

1. The Boltzmann roulette selection method is used in the selection mechanism.

2. The simple chromosome representation searches the lowest course preferentially.

3. The simple crossover reduces the computing time.

In this time, we do not improve crossover and mutation method. We consider diversity of population and width of search space mainly.

Therefore, the most desirable technique can cover wide solution space in short time. Because we adopt random key-based method, we can use traditional one-cut crossover.

In the case of order-phenotype GA, the offspring generated in the crossover often show sudden variations. We need attention to set crossover rate and mutation rate in order-phenotype GA. In order-phenotype GA, if we set large crossover rate, it gives substantially change to population (solutions). This substantially change worsens computation time. So in this chapter, we carefully examine initial parameter value.

Additionally, we apply swap mutation in each stage. Two digits are randomly selected and their positions are exchanged. The fitness function is the inverse of objective function. In selection, we apply Boltzmann roulette selection. In traditional roulette wheel method, the same chromosome may be chosen. In addition, there are more problems such as good individual that is not chosen or becomes extinct immediately even if very good individual appears. Furthermore, the individuals may fall into a local minimum. Boltzmann roulette selection is effective to these problems.

$$eval(v_k) = \frac{e^{\frac{F_k}{b_T}}}{\frac{1}{popSize}\sum_{n=1}^{popSize} e^{\frac{F_A}{b_T}}} \qquad (5.9)$$

Equation 5.9 is a general Boltzmann distribution. b_T expresses the temperature and F_A is the total fitness value in the current generation. At the initial stage of the evolution, this equation gives a lower evaluation standard to maintain the diversity of the population. Later, the evaluation standard becomes severe as evolution advances. This method does not let the population perform a sudden evolution at the same time it evades precocious convergence to the local solution. In this chapter, temperature b_T was decided from the result of the preliminary experiment. In this technique, the evaluation value is updated every generation even if it is the same individual. The roulette wheel selection operates based on the evaluation value obtained from equation 5.9.

The procedure of the Boltzmann roulette selection method is shown in Fig. 5.4.

```
procedure: Boltzmann roulette selection
input: population, P(t -1), C(t -1), popSize
output: population, P(t), C(t)
begin
   step1: Check the current Boltzmann temperature
          if (b_T < b_Tm ) then
             b_T ← b_Tm
   step2: Calculate the average fitness F_AVG for the population using e
          for (k=1 to popSize)
             F_SUM += exp(eval(v_k) / b_T);
             F_AVG ← F_SUM / popSize
   step3: calculate the new fitness
          for (k=1 to popSize)
             eval(v_k) ← exp(eval(v_k) / b_T) / F_AVG;
   step4: Calculate the total fitness F for the population
```

$$F_T = \sum_{k=1}^{popSize} eval(v_k);$$

```
   step5: Calculate selection probability
          p_k=eval(v_k) / F ,  for each v_k

   // The following step 6 & step7 are general roulette wheel selection.
   step6: Calculate cumulative probability
   step7: Generate a random number r from the range[0, 1] to select
          the individuals.
   step8: Update the current Boltzmann temperature
          b_T ← b_T - b_Tc
end
```

Fig. 5.4: The procedure of Boltzmann roulette selection

We also present the total procedure of the Brk-GA in Fig. 5.5. Here, $P(t)$ and $C(t)$ are the parents and offspring in current generation t. b_{Tc} is the amount of change of temperature and b_{Tm} is the minimum temperature, and 1 is inputted to this value in the present study. F_{SUM} and F_{AVG} are defined as the total fitness value and the average fitness value, respectively.

```
procedure: Brk-GA
input: problem data, GA parameters (popSize, maxGen, P_C, P_M)
output: the best solution
begin
    t ← 0;
    initialize P(t) by random key-based encoding routine;
    evaluate P(t) by random key-based decoding routine;
    while (not terminating condition) do
        create C(t) from P(t) by one-cut crossover;
        create C(t) from P(t) by insertion mutation routine;
        climb C(t) by local search routine;
        evaluate P(t) by random key-based decoding routine;
        select P(t+1) from P(t) and C(t) by Boltzmann roulette selection
        t ← t+1;
    end
    output: the best solution
end
```

Fig. 5.5: The overall procedure of Brk-GA

5.3 Numerical experiment

To confirm the performance of Brk-GA, we perform a comparison experiment of four techniques:

- GA 1. spanning tree-based GA (st-GA)
- GA 2. priority-based GA (priGA)
- GA 3. hybrid priority-based GA (h-priGA)
- GA 4. Boltzmann random key-based GA (Brk-GA))

Table 5.2: Constitution of each technique

GA	Encoding method	Crossover	Mutation	FLC	Selection
st-GA	Prüfer number-based	one-cut point crossover	swap	-	roulette wheel
priGA	priority-based	WMX	swap	-	roulette wheel
h-priGA	priority-based	WMX	swap	○	roulette wheel
Brk-GA	random key-based	one-cut point crossover	swap	-	Boltzmann roulette

We show the operator of each GA in Table 5.2. The WMX is crossover method for priority-based encoding method. The FLC is the auto-tuning method of the crossover rate and the mutation rate. Refer to Chapter 3 for the flow of detailed processing. In GA, parameters are fixed and the evolution process is followed based on the value. In this case, crossover and mutation always occur by same probability, and the evolution process must not contain any diversity. If suitable probability is applicable to the generation, we can obtain better solution quality, and the computation time will be shorter. In order to cope with these problems FLC was proposed. In Chapter 3, FLC surely improves evolution speed. However when we verify a graph of evolution process, we can confirm the possibility that fell into a local minimum at an initial stage. In GA, the calculation time is one of the weak points. But, FLC decreases many merits of the multi-thread search instead of improving the calculation time.

Table 5.3 shows the test data for experiment, and we generate these data randomly. The delivery cost was generated from a uniform distribution between 20 and 35. The prohibited delivery routes were generated at 5% of total number of nodes. In addition, the number of products was set to 2. The unit inventory holding costs of each product were set in order to 2 and 3. GA parameters were decided based on a preliminary experiment. However, hybrid GA is not affected by the initial parameter. Additionally, esd-2ITP draws up delivery plans for four periods.

Table 5.3: Test problem size

Problem No.	No. of plants (i)	No. of DCs (j)	No. of customers (k)	Population size	Max generation	Crossover rate (p_C)	Mutation rate (p_M)
1	4	5	15	100	1000	0.3	0.1
2	6	10	35	100	1000	0.3	0.1
3	7	14	45	100	1000	0.5	0.2
4	10	20	60	100	1000	0.5	0.2

As the preceding chapter having described, we decide the amount of production in the current period from equation 4.9 (exponential smoothing).

$$y_{ejk}^{\mathrm{F}}(t) = \alpha y_{ejk}(t-1) + (1-\alpha) y_{ejk}^{\mathrm{F}}(t-1) \qquad (4.9)$$

In this case, $y_{ejk}^{\mathrm{F}}(t)$ is the amount of demand calculated in period t. The esc-2ITP determines next order quantity of DC using this formula (as Chapter 3 described). Also at the start of experiment (*i.e.* t=0), we predict before-period demand by random number from 100 to 150. Moreover, the initial inventory quantities of the DC are set with random numbers 10-20.

Table 5.4 gives the best solution, average of solutions, standard deviation (SD), and average computation time (ACT), of 30 runs. The lower stand of the best value expresses the delivery cost of each period. As a result, Brk-GA gave a superior experimental best value in all problems. The computing time of h-priGA was shorter than Brk-GA. However, Brk-GA was superior in SD value; and, when comparing h-priGA with priGA in SD value, we can confirm large variations of solution increases by incorporating FLC. In the improvement rate that we compared with st-GA, the best value was about 5.7% and the computational time was about 65%. As for the calculation time of Brk-GA provided from this experiment, it seems that it was within the permitted range. The effectiveness of GA with diversity was proven. Finally, we show the evolutional graph of problem 4 in Fig. 5.6. Also an example of the designed network (problem 1) is shown in Fig. 5.7.

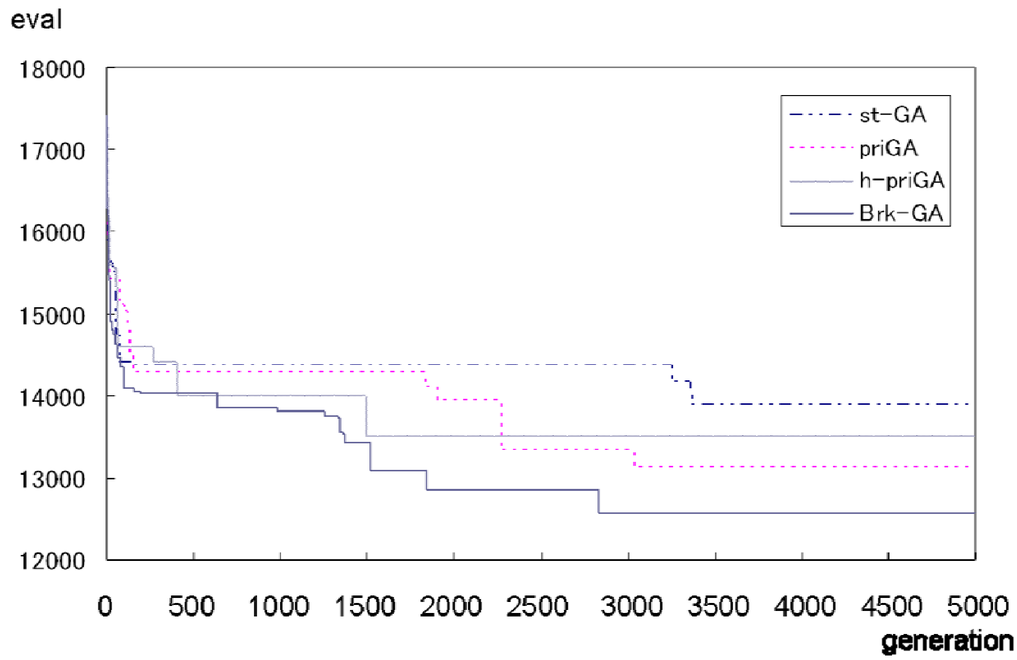


Fig. 5.6: The evolutional graph

Table 5.4: Result of comparison experiments

	st-GA				**priGA**			
Problem No.	**Best**	**AVG**	**SD**	**ACT(s)**	**Best**	**AVG**	**SD**	**ACT(s)**
1	13546 *t*=1: 3704 *t*=2: 3421 *t*=3: 3696 *t*=4: 2725	13884.41	129.77	1.20	13863 *t*=1: 3797 *t*=2: 3277 *t*=3: 3499 *t*=4: 3290	14116.08	46.29	1.03
2	22069 *t*=1: 5728 *t*=2: 5323 *t*=3: 5603 *t*=4: 5415	23714.47	1078.18	2.82	22437 *t*=1: 5778 *t*=2: 5500 *t*=3: 5776 *t*=4: 5383	23650.04	1224.46	2.47
3	38021 *t*=1: 9124 *t*=2: 9708 *t*=3: 9803 *t*=4: 9386	40364.02	887.15	5.60	37770 *t*=1: 9684 *t*=2: 9261 *t*=3: 9207 *t*=4: 9548	39374.20	768.27	4.75
4	57774 *t*=1: 14590 *t*=2: 14464 *t*=3: 14099 *t*=4: 14621	57338.00	1159.67	13.03	56214 *t*=1: 14317 *t*=2: 14514 *t*=3: 13706 *t*=4: 13677	56195.22	1024.03	7.29

	h-priGA				**Brk-GA**			
Problem No.	**Best**	**AVG**	**SD**	**ACT(s)**	**Best**	**AVG**	**SD**	**ACT(s)**
1	13967 *t*=1: 3785 *t*=2: 3255 *t*=3: 3492 *t*=4: 3435	14282.00	72.61	0.55	13501 *t*=1: 3454 *t*=2: 3392 *t*=3: 3271 *t*=4: 3484	13534.91	20.15	1.03
2	22400 *t*=1: 5808 *t*=2: 5504 *t*=3: 5733 *t*=4: 5355	30436.50	3294.66	0.96	21977 *t*=1: 5701 *t*=2: 5350 *t*=3: 5603 *t*=4: 5323	23539.30	891.85	2.11
3	36218 *t*=1: 9522 *t*=2: 8806 *t*=3: 9097 *t*=4: 8793	40922.86	1569.68	1.78	35586 *t*=1: 8834 *t*=2: 8915 *t*=3: 8929 *t*=4: 8908	38956.47	503.52	2.71
4	55192 *t*=1: 13480 *t*=2: 14310 *t*=3: 13265 *t*=4: 14137	56251.93	1362.59	2.53	54470 *t*=1: 13248 *t*=2: 13999 *t*=3: 13758 *t*=4: 13465	54578.66	663.20	4.61

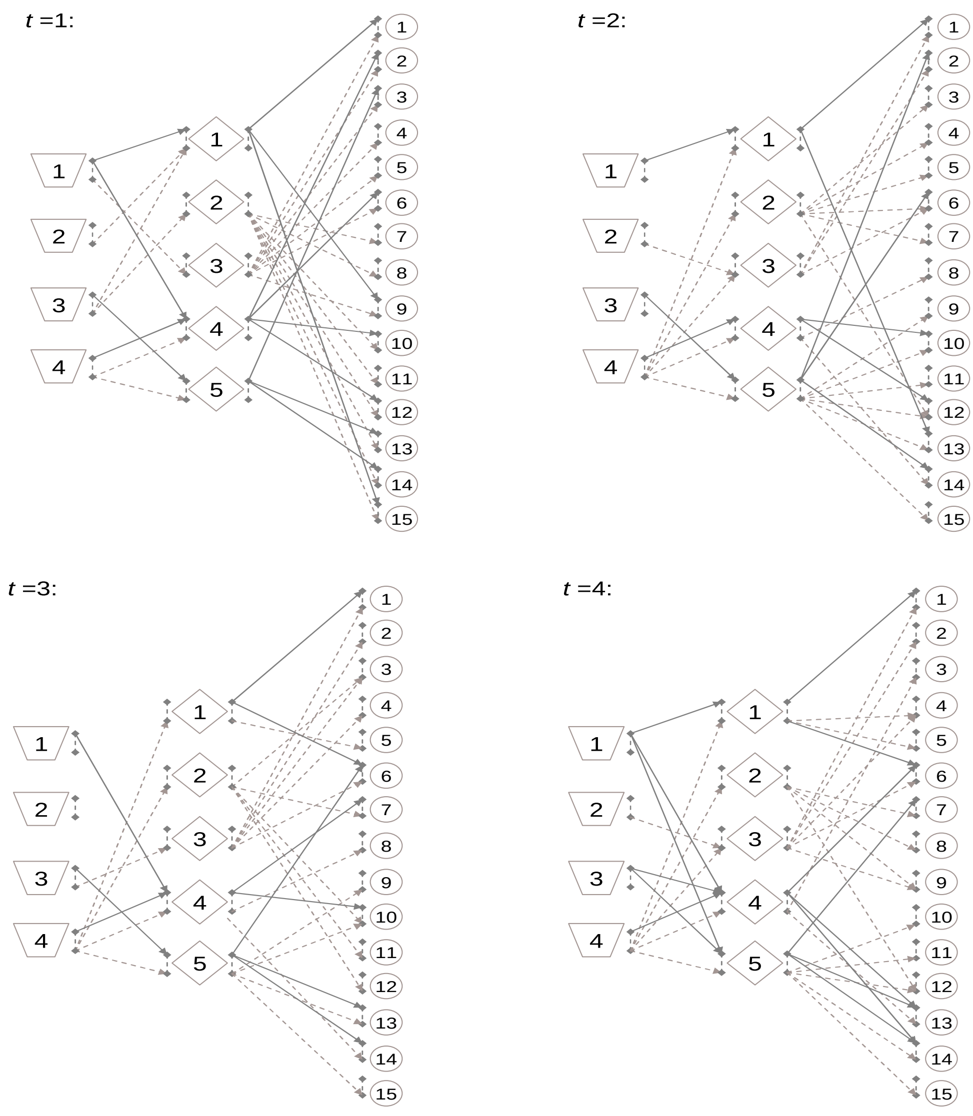


Fig. 5.7: Delivery situation of each term

5.4 Summary

In this research, we designed Brk-GA. Since this algorithm has a simple structure, the computation time can be reduced. In numerical experiments, four different GAs including Brk-GA were applied to esc-2ITP. In this comparative experiment, each GA calculates each problem 30 times.

From the results, although Brk-GA was inferior to h-priGA in average computation time, Brk-GA show the best performance in the best solution, average of the best solution of 30 calculations, and the standard deviation. When we compared with the st-GA that is the oldest method in four GAs, the best solution and computation time were improved about 66.1% and 6.3%. However, today's distribution routes take on a flexible form. It means that products may not be delivered based on conventional order (plants - DCs – customers). We are planning to consider more concrete inventory-control techniques and a shorter product life cycle.

Chapter 6. Progressive Flexible Logistics Network Model

6.1 Introduction

In Chapter 4, we proposed esc-2ITP. This model includes concepts of inventory and time. However, the basic network structure is two-stage TP. Therefore, we propose a more extended TP with new connection form. This TP model is Progressive Flexible Logistics Network (PFLN) model. The main differences between the PFLN and esc-2ITP proposed in Chapter 4 are the additional of a retailer and the duplicating of direct delivery from. If it assumes that there are four elements (plants, DCs, retailers, and customers), traditional TP considers only basic flow that passes each element in order. However, PFLN models the deliver directly routes such as plants to customers and deliver directly from DC to customers (products do not pass retailers) etc. Therefore, we propose the solution technique for flexible logistics network model called as the network segment method. This network segment method resolves a complicated network to simple TP models. If this technique is applied, we do not need to design a special chromosome representation (i.e. a lot of GA for TP can be applied).

The outline of this chapter is as follows. Section 6.2 describes the mathematical formulation of the model. Section 6.3 includes contains of Brk-GA. Section 6.4 show the numerical experiments. Finally, section 6.5 concludes with conclusion.

6.2 Model explanation

This TP can treat all currently delivery patterns (Fig. 6.1). For example, companies manage a shop in Internet without having a real shop. We did not consider this pattern in

Chapter 4. Moreover, the difference of PFLN model and esc-2ITP is not only delivery routes. We consider that proposing PFLN will be one of the foundations of delivery planning optimizer development. Some precedent studies described with minor differences of proposal model. However, PFLN can correspond to today's distribution form and inventory. In addition, PFLN can draw the optimal delivery routes according to kind of product. Precedent studies also proposed various solution methods. According to minor difference of problem, development of new chromosome is inefficient. We also strive to solve this problem.

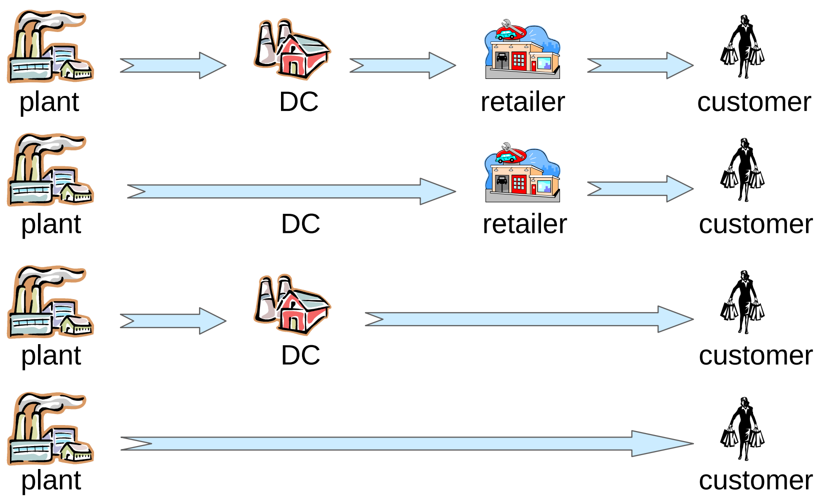


Fig. 6.1: Various Delivery Pattern

In Fig. 6.2, we show the image of PFLN model. To define this model, following assumptions are used:

A1. Maximum capacity of each facility (i, j, k) is known.

A2. Demand of customer l is also already known.

A3. Arrangement place of each facility (i, j, k) is known.

A4. Delivery cost $(c^1_{pij}, c^2_{pjk}, \ldots, c^6_{pjl})$ is known in each stage.

A5. Kind of product p manufactured at each factory is known.

A6. Production is instantly performed.

A7. Delivery from plant i to customer l is carried out in an instant.

A8. Change of the value by progress of time is not taken into consideration.

A9. This model draws up the delivery plan for multi periods.

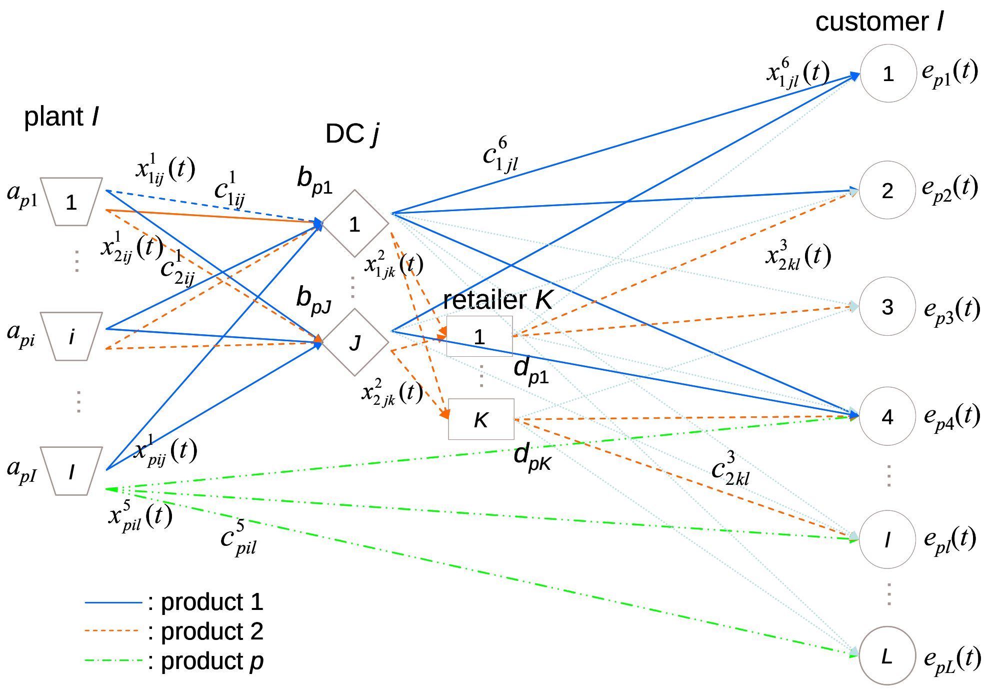


Fig. 6.2: Illustration of proposal model

***Indices*:**

i: index of plants (I = 1, 2,…,I)

j : index of DCs (j = 1, 2,…,J)

k : index of retailers (k= 1, 2,…,K)

l : index of customers (l= 1, 2,…,L)

p : kind of products.

***Parameters*:**

a_{pi} : capacity of plant i for product p.

b_{pj} : capacity of DC j for product p.

d_{pj} : capacity of Retailer k for product p.

$e_l(t)$: demand of customer l for product p in time period t.

c^1_{pij}: unit cost of delivery from plant i in product p to DC j.

c^2_{pjk}: unit cost of delivery from DC j in product p to retailer k.

c^3_{pkl}: unit cost of delivery from retailer k in product p to customer l.

c^4_{pik}: unit cost of delivery from plant i in product p to retailer k.

c^5_{pil}: unit cost of delivery from plant i in product p to customer l.

c^6_{pjl}: unit cost of delivery from DC j in product p to customer l.

c^0_{pi} : fixed manufacture cost of product p.

c_{pj} : carrying cost in DC j for product p.

$q_{pi}(t)$: quantity of production of plant i in period t.

***Decision variables*:**

$x^1_{pij}(t)$: shipment amount of product p from plant i to DC j in time period t.

$x^2_{pjk}(t)$: shipment amount of product p from DC j to retailer k in time period t.

$x^3_{pkl}(t)$: shipment amount of product p from retailer k to customer l in time period t.

$x^4_{pik}(t)$: shipment amount of product p from plant i to retailer k in time period t.

$x^{5}_{pil}(t)$: shipment amount of product *p* from plant *i* to customer *l* in time period *t*.

$x^{6}_{pjl}(t)$: shipment amount of product *p* from DC *j* to customer *l* in time period *t*.

u_{pi}(t): amount of inventories of the plant *i* in period *t*.

u_{pj}(t): amount of inventories of the DC *j* in period *t*.

z_i: 0-1 variable that takes on the value 1 if plant *i* is opened.

We formulate the mathematical model of the problem as follows.

$$\min \quad z=\sum_{t=1}^{T}\sum_{p=1}^{P}\left(\sum_{i=1}^{I}\sum_{j=1}^{J}c^{1}_{pij}x^{1}_{pij}(t)+\sum_{j=1}^{J}\sum_{k=1}^{K}c^{2}_{pjk}x^{2}_{pjk}(t)+\sum_{k=1}^{K}\sum_{l=1}^{L}c^{3}_{pkl}x^{3}_{pkl}(t)+\sum_{i=1}^{I}\sum_{k=1}^{K}c^{4}_{pik}x^{4}_{pik}(t)\right.$$
$$\left.+\sum_{i=1}^{I}\sum_{l=1}^{L}c^{5}_{pil}x^{5}_{pil}(t)+\sum_{j=1}^{J}\sum_{l=1}^{L}c^{6}_{pjl}x^{6}_{pjl}+\sum_{j=1}^{J}c_{pj}u_{pj}(t)+\sum_{i=1}^{I}c^{0}_{pi}z_i(t)\right) \tag{6.1}$$

s. t.

$$u_{pi}(t)=u_{pi}(t-1)+q_{pi}(t)-\sum_{j=1}^{J}x^{1}_{pij}(t)-\sum_{k=1}^{K}x^{4}_{pik}(t)-\sum_{l=1}^{L}x^{5}_{pil}(t),\quad \forall t,i,p \tag{6.2}$$

$$u_{pj}(t)=u_{pj}(t-1)+\sum_{i=1}^{I}x^{1}_{pij}(t)-\sum_{k=1}^{K}x^{2}_{pjk}(t)-\sum_{l=1}^{L}x^{6}_{pjl}(t),\quad \forall t,\ p,\ j \tag{6.3}$$

$$u_{pi}(t-1)+q_{pi}(t)\le a_{pi}z_i(t),\ \forall t,\ p,i \tag{6.4}$$

$$u_{pj}(t-1)+\sum_{i=1}^{I}x^{1}_{pij}(t)\le b_{pj},\quad \forall t,p,j \tag{6.5}$$

$$\sum_{i=1}^{I}x^{4}_{pik}(t)+\sum_{j=1}^{J}x^{2}_{pjk}(t)\le\ d_{pk},\quad \forall t,k,p \tag{6.6}$$

$$\sum_{i=1}^{I}x^{5}_{pil}(t)+\sum_{j=1}^{J}x^{6}_{pjl}(t)+\sum_{k=1}^{K}x^{3}_{pkl}(t)\ge\ e_{pl}(t),\quad \forall t,l,p \tag{6.7}$$

$$x^{1}_{pij},x^{2}_{pjk},x^{3}_{pkl},x^{4}_{pjk},x^{5}_{pil},x^{6}_{pjl}\ge 0,\ \forall i,j,k,l,p \tag{6.8}$$

$$u_{pi}(t)\ge 0,\ \forall\ p,i,t \tag{6.9}$$

$$u_{pj}(t)\ge 0,\ \forall\ p,\ j,t \tag{6.10}$$

$$z_i(t)=\{0,1\},\ \forall i \tag{6.11}$$

While constraints (6.2) represent the inventory of plant *i* in period *t*, and (6.3) meens inventory of DC *j* in period *t*, respectively. The constraint (6.4) shows that remainder of before period and quantity of production does not exceed the capacity. The constraint (6.5) shows that remainder of before period and order quantity of current term do not exceed a capacity. The constraints (6.6) and (6.7) respectively represent the demand of retailer and customer.

6.3 Solution method by GA

In article [45], [46], the authors proposed new hybrid evolutionary algorithm (hEA) technique to solve a problem. At the same time, they also proposed unique chromosome representation. However, the chromosome representation is specialized for flexible logistics network model. Furthermore, this technique is difficult to reproduce a flexible delivery form completely. Even if it uses such a technique, the width of the search space will become narrow. In addition, this technique needs much computation time.

In the same way, PFLN model has complexity delivery route, so chromosome representation becomes particularly important. According to minor difference of problem, development of new chromosome is inefficient. Therefore, proposed a network segment method which divides PFLN to three sub-networks based on difference of delivery form. If this method applied, it becomes possible to avoid inefficiency which uses different chromosome due to the slightly change of network model.

6.3.1 Network segment method

Designing the chromosome for PFLN is difficult for correctly expressing of delivery routes. Even though GA generates the initial solutions randomly, we cannot take all kinds of solution patterns. The initial populations may have only similar solutions. Therefore, a search space is narrow even if we perform calculation. In addition, the coding method that makes combination immense not only extending the computation time but also reducing the probability of obtaining high quality semi-optimal solution.

Then, we propose a solution technique for the flexible logistics network model. We call this technique as the network segment method. This method resolves complicated network to simple TP models. It is not necessary to design a special chromosome representation. In this chapter, we use a random key-based encoding. This encoding method encodes a solution with random numbers. These values are sort keys to decode the solution. Refer to Chapter 5 for detailed explanation of this method. We show the example that applied random key-based encoding model in Fig. 6.3. When we apply network segment method, at first we consider the three-stage TP model that all nodes are connected. Next, we divide a network model into three steps. It is suitable method for chromosome expression. By using this method, PFLN can solve like three-stage TP. Fig. 6.4 show an example of a chromosome.

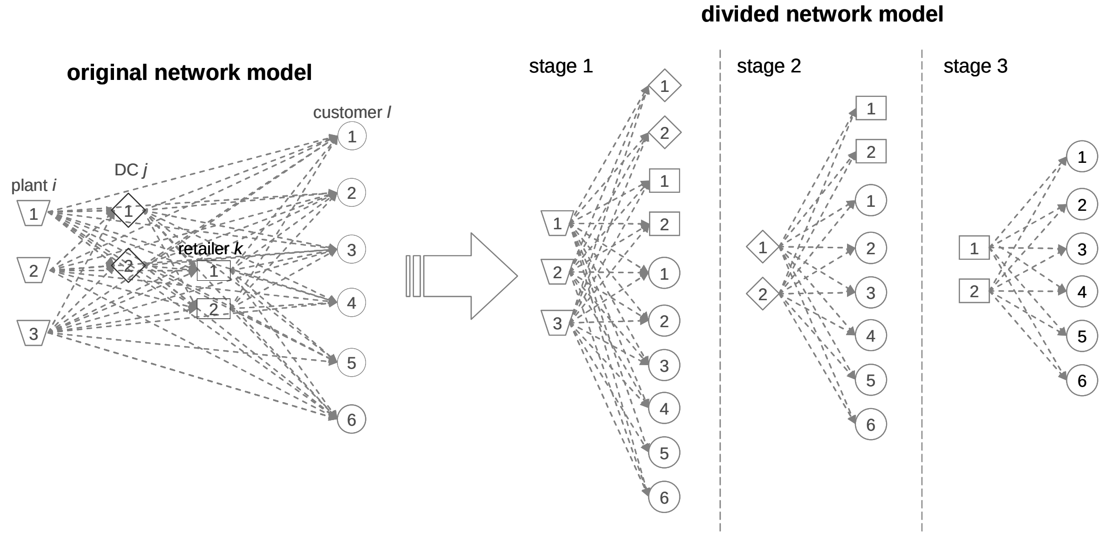


Fig. 6.3: Solution process of this problem

node ID l :	1	2	3	1	2	1	2	1	2	3	4	5	6
priority v (l) :	0.912	0.361	0.233	0.757	0.133	0.416	0.109	0.344	0.805	0.276	0.562	0.183	0.331

1	2	1	2	1	2	3	4	5	6	1	2	1	2	3	4	5	6
0.643	0.932	0.119	0.592	0.246	0.716	0.168	0.737	0.503	0.194	0.372	0.863	0.241	0.955	0.152	0.663	0.394	0.625

Fig. 6.4: A sample of chromosome

6.3.2 Genetic operators

Genetic operators other than chromosome representation need to refer to Chapter 5.

6.4 Comparative experiments

We apply network segment method to five GAs. Then we perform a comparison experiment. The five GAs shown below:

1. spanning tree-based GA (st-GA), [17] [30]
2. priority-based GA (priGA)
3. hybrid priority-based GA (h-priGA)
4. Boltzmann random key-based GA (Brk-GA))

5. hybrid Boltzmann random key-based GA (h-BrkGA)

Table 6.1: Constitution of each technique

GA	encoding method	crossover	mutation	FLC	selection
st-GA	prüfer number-based	one-point crossover	swap	-	roulette wheel
priGA	priority-based	WMX	swap	-	roulette wheel
h-priGA	priority-based	WMX	swap	○	roulette wheel
Brk-GA	random key-based	one-point crossover	swap	-	Boltzmann roulette
h-BrkGA	random key-based	one-point crossover	swap	○	Boltzmann roulette

In table 6.1, we show the operators of each GA. In Chapter 3, we described in details of FLC. Although this technique shortens computational time sharply we checked possibility of falling into a local minimum in Chapter 4. So, we compare with Boltzmann roulette selection. Moreover, we also compare normal Brk-GA with hybrid Boltzmann random key-based Genetic Algorithm (h-BrkGA) which incorporated FLC to Brk-GA.

Table 6.2 is the test data. We generated these data randomly. The delivery costs generated from a uniform distribution between 20 and 45. The forbidden delivery routes were consisted of 10% on the total number of nodes. In addition, the number of products set to 2. The unit inventory holding cost assumes 2 and 3 respectively. Then GA parameters were decided based on preliminarily experiment. In this experiment, we design the delivery plan over 4 terms. In this proposal model, we also decide the amount of production in current period from exponential smoothing. We described this technique in Chapter 4.

$$\textit{Predicted value} = \alpha \times \textit{Last track record value} + (1 - \alpha) \times \textit{The last predicted value}$$

$$x^{\mathrm{F}}_{pj}(t) = \alpha\left\{x^{2}_{pjk}(t-1) + x^{6}_{pjl}(t-1)\right\} + (1-\alpha)x^{\mathrm{F}}_{pj}(t-1) \qquad (6.13)$$

α is the smoothing factor, $x_{pj}^{\mathrm{F}}(t)$ is the amount of demand forecasting in period t. The value that deducted the present volume of inventories from $x_{pj}^{\mathrm{F}}(t)$ serves as each order quantity of DC.

Table 6.3 gives the best solution, average of solutions (AVG), standard deviation (SD) and average computation time (ACT) of 20 runs. As a result, the each best solution was obtained from Brk-GA. In addition, Brk-GA was the best in AVG and ACT. In problem 4, st-GA and priGA show low SD. However, when we compare AVG value with Brk-GA, we can understand that searching converged without providing a good solution. In addition, FLC can largely reduce the calculation time, but when we compared priGA with h-priGA or Brk-GA with h-BrkGA, we are able to confirm that SD turned worse.

Table 6.2: Experimental data

Problem No.	No. of plants (i)	No. of DCs (j)	No. of retailers (k)	No. of customers (l)	Population size	Crossover rate	Mutation rate
1	2	5	8	20	100	0.2	0.1
2	3	7	12	30	100	0.2	0.1
3	5	3	12	45	100	0.3	0.1
4	6	8	20	80	100	0.3	0.1

In fact, sudden evolution urges to converge to local minimum. From the quality of solutions, Brk-GA effectively reduces calculation time. When we compared Brk-GA with st-GA that suggested earliest in four techniques about problem 4, the best solution was improved 6.3%. Particularly calculation time was improved 66.1%.

We present the evolution graph of problem 4 which calculate 5000 times and period is set to 1 (*i.e.* t=1) in Fig. 6.5. Searching solution in all GA finished at an early stage. For the reasons mentioned above, we achieved at main aim of Brk-GA such as shortening the computing time, precocious convergence to the local minimum and reduction the unevenness of the solution.

Finally, Fig. 6.6 shows the delivery route that obtained from Brk-GA. This is a delivery route in problem 1. From this figure, we can see that the flexible delivery route is designed and delivery plan for four periods is designed. The proposal solution process easily solved complicated network models. However, improvement is necessary when we consider delivery time of each product or each route.

Table 6.3: Results of comparison experiments

	st-GA				**priGA**			
Problem No.	Best	AVG	SD	ACT (s)	Best	AVG	SD	ACT (s)
1	55318.12	56763.05	778.97	14.45	55188.03	55888.55	576.54	15.25
2	59601.00	60743.65	674.84	24.79	59280.22	60621.00	762.91	24.47
3	82874.00	85566.35	1108.78	34.66	82331.00	83917.25	1315.4	31.01
4	87038.01	88138.30	765.76	34.84	86915.00	88369.35	755.95	32.61
	h-priGA				**h-BrkGA**			
Problem No.	Best	AVG	SD	ACT (s)	Best	AVG	SD	ACT (s)
1	55045.40	56191.05	573.02	10.80	54929.00	56349.15	756.64	3.83
2	59065.03	60600.90	582.70	16.04	58887.80	60229.00	696.65	5.00
3	81726.06	84165.50	991.04	19.89	79875.00	81827.60	1044.15	6.17
4	85921.00	87878.05	1000.40	20.42	82391.20	85845.55	1476.72	7.17
	Brk-GA							
Problem No.	Best	AVG	SD	ACT (s)				
1	54909.22	55003.15	294.20	3.73				
2	58263.17	59978.90	570.62	4.95				
3	79761.87	80949.85	669.86	6.13				
4	81544.05	83480.15	948.74	6.91				

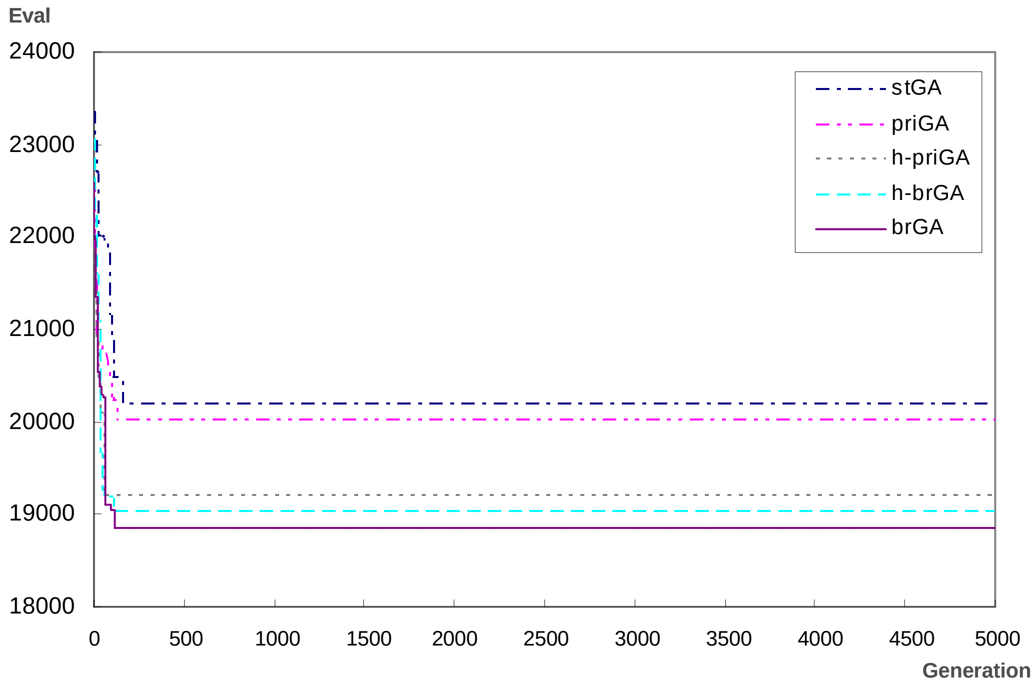


Fig. 6.5: Graph of the evolution process

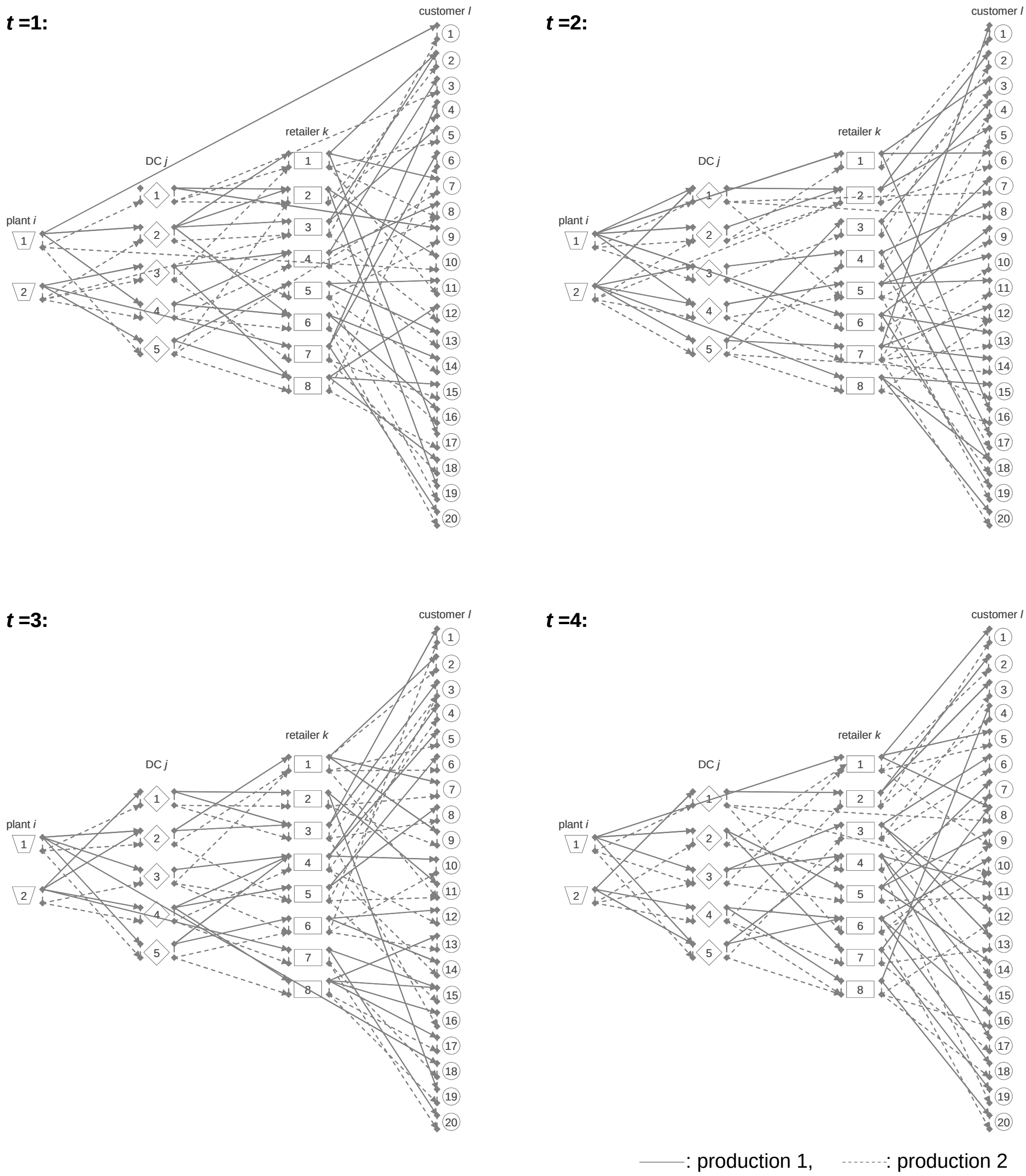


Fig. 6.6: Delivery routes obtained from Brk-GA

6.5 Summary

In this chapter, we proposed PFLN model as a more realistic TP model. The main difference of PFLN and esc-2ITP is the addition of retailer and direct delivery routes. In addition, in Flexible Logistics Network, designing chromosomes was difficult. Because, correctly expressing of delivery routes is difficult. Therefore, we suggested network segment method which deconstruct complexity network model to simple TP. In numerical experiments, we apply the proposed solution technique to five GAs. As a calculation result, the five GAs with network segment method can solve PFLN model. Brk-GA was superior to other five techniques in the best solution, the averages of solutions, the standard deviation, and the computation time. On the other hand, when we combine FLC to Brk-GA, the solution quality became worse. Especially standard deviation gets worse. In conjunction with the result of Chapter 5, we show robustness and reliability of Brk-GA. Brk-GA is superior and more effective than traditional GA developed for TP.

Proposal of chromosome representation that can investigate large solution space quickly and the experiment that used more large-scale data sets or real data sets are feature research.

Chapter 7. Conclusion

In this research, we addressed the proposal of new TP model and development of the solution technique applicable to TP general-purpose. The proposed TP model includes difficult constraints for calculating a solution. In order to solve these models, we improved GA which is one of the meta-strategies. GA can apply to various combinatorial optimization problems with devising the chromosome design. In addition, it can apply even when the characteristic of the objective function is not clear enough. However, GA has several problems such as solutions falls into local minimum and it takes so much computation time. Therefore, we attempt to overcome the weak point of GA, specifically, improvement of chromosome representation and development of new selection mechanism for GA.

In Chapter 2, we described traditional TP model and fundamental GA. Simultaneously we describe the advantage using GA.

In Chapter 3, we proposed a multi-product two-stage transportation model with exclusionary side constraints. In this model, plants produce multiple items, and the delivery routes and transportation costs vary with the products. Since GA expresses a solution as a chromosome, it was difficult to solve a model including the exclusionary side constraints. Therefore, in order to calculate by GA and to satisfy exclusionary side constraints easily, we also propose improved priority-based representation. This technique has no necessary check routine of infeasible solution. Additionally, it can reduce the computation time simultaneously. In numerical experiments, we compared four GAs. As a result, it became clear that proposal TP model could solve by four GA with improved priority-based representation. Moreover, h-GA reduced computation time sharply and achieved good results.

In Chapter 4, we proposed ecs-2ITP model as extended TP proposed in Chapter 3. This model includes the concepts of proper inventory and various costs generated on logistics network. Simultaneously, ecs-2ITP designs the delivery plan for multiple periods. The development of solution algorithm for ecs-2ITP is next subject. In Chapter 4, we describe the algorithm that improves weak points of GA.

In Chapter 5, in order to solve esc-2ITP, we designed Brk-GA. This GA has following features:

1. The Boltzmann roulette selection method uses in the selection mechanism.
2. The simple chromosome representation searches the lowest course preferentially.
3. The simple crossover reduces the computing time.

Traditional GA has several problems. These are a high calculation load, precocious convergence to local minimum, and complexity of parameter setting. In this chapter, we consider diversity of population and width of search space mainly. In a numerical experiment, we compared four GA techniques. The Brk-GA was superior to the previous methods in the quality of the solution. In the improvement rate that we compared with st-GA developed for TP, the best value was about 5.7% and the computational time was about 65%.

In Chapter 6, we proposed the PFLN model. The main difference of PFLN and esc-2ITP proposed in chapter 4 is addition of retailers and direct delivery routes. Moreover, this TP can models all currently delivery patterns. Moreover, we propose a network segment method which divides PFLN to three sub-networks based on difference of delivery form. This method can avoid inefficiency caused by using different chromosomes in accordance with slight changes of the network. In numerical experiments, five GAs combined network segment method can solve the PFLN model.

In addition, we obtained the each best solution from Brk-GA. When we compared Brk-GA with st-GA, the best solution was improved 6.3%. Particularly calculation time was improved 66.1%. The simple crossover method and random key-based representation are effective methods to reduce the calculation time. In conjunction with the result of Chapter 5, we show that Brk-GA is superior and more effective than traditional GA developed for TP.

This thesis proposed realistic TP models which consider exclusionary side constraints, kind of products, inventory, direct shipping route and multiple periods. These elements reflected needs for the modern distribution. Although these proposal TP models have different complexity, in order to obtain stable solution in realistic time, we propose solution methods based on GA. We found the excellent technique by combine improved gene expression and selection technique. Moreover, through the analysis of evaluation result, we clarify the effectiveness and limit of each technique. This article has a great value that proposed effective and stability techniques for modern various TPs. This article greatly contributes to the future distribution optimization. We are planning to consider more concrete inventory-control techniques and a shorter product life cycle. Suggestion of the chromosome representation method that can investigate large solution space quickly and the experiment that used larger scale data sets or real data sets is a future problem.

References

[1]. Kadir Ertogral, S. David Wu and Laura I. Burke, “Coordination Production and Transportation Scheduling in the Supply Chain”, *Technical Report #98T-010, Department of Industrial & Mfg. Systems Egnieering, Lehigh Univerisity,* 1998.

[2]. K. Taniguchi , T. Terano, “A Study of Supply Chain Management Support by means of Evolutionary Computation, *IEICE technical report. Artificial intelligence and knowledge-based processing*, vol.100(89), pp57-61, 2000, [in Japanese].

[3]. T. Enkawa, K. Itoh, T. Kasahara, D. Chen, “A Multi-Period Vehicle Routing Problem Minimizing the Inventory with no Shortage and Its Algorithm”, *Journal of Japan Industrial Management Association*, vol.46(5), pp.492-502, 1995, [in Japanese].

[4]. T. Egawa, H. Konno, “Transportation Problem Optimization with a Concave Production Cost Using Integer Programming”, *Proc. of The 2004 Fall National Conference of Operations Research Society of Japan*, pp.68-69, 2004, [in Japanese].

[5]. S. Higuchi, S. Kataoka, “The Branch Bound Method Using List Expression of The Transportation Problem with Degree Restriction”, *Proc. of The 2006 Fall National Conference of Operations Research Society of Japan*, pp.216-217, 2006, [in Japanese].

[6]. J. Nakano, T. Tokuyama, “Efficient Algorithms for the Hitchcock Transportation Problem”, *Proc. of The 44th Information Processing Society of Japan*, pp.89-90 1992.

[7]. I. Kuwajima, Y. Nojima and H. Ishibuchi, “Genetic Fuzzy Rule Selection with Pareto-optimal Rules as Candidate Rules”, *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, vol.20(2), pp.231-243, 2008, [in Japanese]

[8]. H. Ishibuchi and H. Tanaka, “An Analysis of the Transportation Problem Based on Interval Demand Functions”, *Journal of Japan Industrial Management Association*, vol.39(4), pp.262-269, 1988, [in Japanese].

[9]. Sadiq M. Sait and H. Youssef, “*Iterative Computer Algorithms with Applications in Engineering*”, Wiley-IEEE Computer Society, 2000.

[10]. M. Yanagiura and T. Ibaraki, “Combination Optimization”, *Asakura Publishing*, 2001.

[11]. M. Gen, Y. Li and K. Ida, “Spanning Tree-based Genetic Algorithm for Bicriteria Fixed Charge Transportation Problem”, *Journal of Japan Society for Fuzzy Theory and Systems*, vol.12(2), pp.87-95, 2000, [in Japanese].

[12]. J, Gottlieb, “Genetic Algorithms for the Fixed Charge Transportation Problem”, *Proc. of IEEE International Conference on Evolutionary Computation*, pp.330-335, 1998.

[13]. S.M. Soak, “’Adaptive Link Adjustment’ Applied to the Fixed Charge Transportation Problem”, *IEICE TRANSACTIONS ON FUNDAMENTALS OF ELECTRONICS COMMUNICATIONS AND COMPUTER SCIENCES*, vol.E90-A no.12 pp.2863-2876, 2007.

[14]. L.X. Yang, L.Z. Liu, “Fuzzy Fixed Charge Solid Transportation Problem and Algorithm”, *APPLIED SOFT COMPUTING*, vol.7, no.3, pp.879-889, 2007.

[15]. M. Sun, “A Tabu Search Heuristic Procedure for Solving the Transportation Problem with Exclusionary Side Constraints”, *Journal of Heuristic*, vol.3,

pp.305-326, 1998.

[16]. M. Sun, "The Transportation Problem with Exclusionary Side Constraints and Two Branch-and-bound Algorithms", *European Journal of Operational Research,* vol.140(3), pp. 629-647, 2002.

[17]. A. Syarif & M. Gen, "Solving Exclusionary Side Constrained Transportation Problem by Using a Hybrid Spanning Tree-based Genetic Algorithm", *Journal of Intelligent Manufacturing*, vol.14(3/4), pp.389-399, 2003.

[18]. M. Gen & A. Syarif, "A Genetic Algorithm for Two-stage Transportation Problem Using Priority-based Encoding" *OR Spectrum*, vol.28, pp.337-354, 2005.

[19]. N. Jawahar, A.N. Balaji. "A Genetic Algorithm for the Two-stage Supply Chain Distribution Problem Associated with a Fixed Charge", *EUROPEAN JOURNAL OF OPERATIONAL RESEARCH*, vol.194, no.2, pp.496-537, 2009.

[20]. F. Altiparmak, M. Gen, L. Lin & T. Paksoy, "A Genetic Algorithm Approach for Multi-objective Optimization of Supply Chain Networks", *Computers and Industrial Engineering*, vol.51(1), pp.196-215, 2006.

[21]. A. Morita, F. Shimada, Hiroaki Ishii, "The Fuzzy Transportation Problem Aiming at Optimization of the Transportation Cost to Each Demand Point", *Proc. of The 2000 Spring National Conference of Operations Research Society of Japan*, pp.230-231, 2000, [in Japanese].

[22]. M. Moki, Y. Nishioka, "Formulization of the Inventory Transportation Problem Considered the Manufacturing Planning of the Middle Base", *Proc. of The 2003 Fall National Conference of Operations Research Society of Japan*, pp.44-45, 2003, [in Japanese].

[23]. S.H. Zegordi, M.A. Beheshti Nia, “A multi-population Genetic Algorithm for Transportation Scheduling”, *Transportation Research Part E: Logistics and Transportation Review*, vol.45, np.6, pp.946-959, 2009.

[24]. D. Nasoa, M. Suricoa, B. Turchianoa, U. Kaymakb, “Genetic algorithms for supply-chain scheduling: A case study in the distribution of ready-mixed concrete”, *EUROPEAN JOURNAL OF OPERATIONAL RESEARCH*, vol. 177, no.3, pp.2069-2099, 2007.

[25]. M. Sakawa, I. Nishizaki, and Y. Uemura, “Fuzzy Programming and Profit and Cost Allocation for a Production and Transportation Problem in a Housing Material Manufacturer”, *Journal of Japan Society for Fuzzy Theory and Systems*, vol.11(3), pp.145-153,1999, [in Japanese].

[26]. K. Oyama, “The Model and Analysis in a Transportation Problem of Garbage”, *Operations research as a management science research*, vol.29(7), pp.422-430,1984, [in Japanese].

[27]. M. Gen & R. W. Cheng, “*Genetic Algorithms and Engineering Design*”, John Wiley & Sons, 1997.

[28]. M. Gen & R. W. Cheng, “Genetic Algorithms and Engineering Optimization”, New York: John Wiley & Sons, 2000.

[29]. Y. Z. Li, M. Gen, & K. Ida, “Improved Genetic Algorithm for solving multi-objective solid transportation problem with fuzzy number”, *Japanese Journal of Fuzzy Theory and Systems*, vol.4(3), pp.220-229, 1998.

[30]. M. Gen, Y. Z. Li & K. Ida, “Solving Multi-Objective Transportation Problem by Spanning Tree-Base Genetic Algorithm”, *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*,

Vol.E82-A(12), pp.2802-2810, 1999.

[31]. B. Cao, “Transportation Problem with Nonlinear Side Constraints a Branch And Bound Approach”, *Mathematical Methods of Operations Research*, vol.36, pp.185-197, 1992.

[32]. B. Cao, G. Uebe, “Solving Transportation Problems with Nonlinear Side Constraints with Tabu Search”, *Computers and Operations Research*, vol.22, pp.593-603, 1995.

[33]. G. Kannan, P. Sasi Kumar, V.P. Vinay, "Comments on "Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm”by Jung-Bok Jo, Yinzhen Li, Mitsuo Gen, Computers & Industrial Engineering (2007)", *Computers & Industrial Engineering*, vol.55, no.2, pp.533-534, 2008.

[34]. S. Ataka and M. Gen “Solution Method of Multi-Product Two-Stage Logistics Problem with Constraints of Delivery Course”, *IEEJ Transactions on Electronics, Information and Systems*, vol.128(3), pp.456-461, 2008, [in Japanese].

[35]. M. Gen, R. W. Cheng & L. Lin, “Network Models and Optimization: Multiobjective Genetic Algorithm Approach”, Springer, 2008.

[36]. M. Yokoyama, “Integrated Optimization of an Inventory-Distribution System with Cooperation among Distribution Centers”, *Journal of Japan Industrial Management Association*, vol.50, no.5, pp.316-325, 1999, [in Japanese].

[37]. H. Kasigai, S. M. Kim, Tadayuki Masui, “A Study on Multi-Stage Inventory Model”, *Journal of Japan Industrial Management Association*, vol.26, no.3, pp.231-237, 1975, [in Japanese].

[38]. T. Masui, “A Study on Multi-Stage Inventory Model with Restrictions on

Storage Capacities: A Study on Multi-Stage Inventory Model(Part II)", *Journal of Japan Industrial Management Association*, vol.31(3), pp.271-277 ,1980, [in Japanese].

[39]. T. Masui, "A Study on Sensitivity Analysis for a N-Stage Series-Type Inventory Model: A Study on Multi-Stage Inventory Model (Part III)", Journal of Japan Industrial Management Association, vol.34(2), pp.112-118, 1983, [in Japanese].

[40]. M De La Maza, B Tidor, "Boltzmann Weighted Selection Improves Performance of Genetic Algorithms", *Massachusetts Institute of Technology, Technical Report*, AIM-1345, 1991.

[41]. N. Mori, J. Yoshida, H. Tamaki, H. Kita and Y. Nishikawa, "A Thermodynamical Selection Rule for the Genetic Algorithm", *IEEE Proc. Int. Conf. on Evolutionary Computation (ICEC-95)*, pp.188-192, 1995.

[42]. D. Ambedkar, M. N. Murty, and S. Bhatnagar, "Cauchy Annealing Schedule: An Annealing Schedule for Boltzmann Selection Scheme in Evolutionary Algorithms", *Proc. of IEEE Congress on Evolutionary Computation (CEC2004)*, vol.1, pp.55-62, 2004.

[43]. T.H. Wu, C.C. Chang, J.Y. Yeh, "A Hybrid Heuristic Algorithm Adopting Both Boltzmann Function and Mutation Operator for Manufacturing Cell Formation Problems", *INTERNATIONAL JOURNAL OF PRODUCTION ECONOMICS*, vol.120, no.2, pp.669-688, 2009.

[44]. P. Chen, M. Shahandashti, Hybrid of Genetic Algorithm and Simulated Annealing for Multiple Project Scheduling with Multiple Resource Constraints., *Automation in Construction*, vol.18(4), pp.434-443, 2009.

[45]. L. Lin, M. Gen, and X.Wang, "A Hybrid Genetic Algorithm for Logistics

Network Design with Flexible Multistage Model", *International Journal of Information Systems for Logistics and Management*, vol.3(1), pp.1-12, 2007.

[46]. L. Lin, M. Gen, and X. Wang, "Integrated Multistage Logistics Network Design by Using Hybrid Evolutionary Algorithm", *Computers & Industrial Engineering*, vol.56(3), pp.854-873, 2009.

Acknowledgments

I received Prof. Mitsuo Gen's careful guidance from the foundation of GA to application. I thank for that I was able to finish the doctoral dissertation by his guidance. I would like to express my deep and sincere gratitude to my supervisor, Prof. Mizuho Iwaihara who provided helpful comments and suggestions. I have been belonged to Iwaihara laboratory toward the end of Ph.D. program, and even in my short time I have learned a lot. He taught me how to analyze problems and express my ideas. He suggested me different ways to approach a research problem.

I warmly thank my helpful co-adviser, Prof. Kotaro Hirasawa and Prof. Takayuki Furutsuki, whose meticulous comments were numerous help to me. They gave me advisement how to continue my research also brought out the excellent and wonderful ideas in me.

My grateful thanks also go to my friends especially those who study together at Graduate school of Information, Production and Systems, Waseda University – Ms. Jeong-eun Lee, Mr. Feng Wen, Mr. Byungki Kim and Ms. Uanaramvilai Voravee. They kept me in good spirits.

The last, but not least, I would also like to express my gratitude to my parents, Makoto Ataka and Yasuko Ataka, for their support and warm encouragements. With out them, I could not be in this position. They always give me with the great opportunity to be where I am. I love both my cats, Goro and Riri.

The responsibility for the final formulation, and any errors that it may concern, are entirely mine.

List of Publications

Journals:

1. S. Ataka B. Kim & M. Gen, “Optimal Design of Two-stage Logistics Network Considered Inventory by Boltzman random key-based GA”, 電気学会 共通英語論文誌「Electronics, Information and Systems」特集, 2010 年３月（採録決定）.
2. 安高真一郎, 玄光男, “ハイブリッド型遺伝的アルゴリズムによる配送経路に制約を伴う多品種 2 段階配送計画モデルの解法”, 電気学会論文誌 C, Vol. 128, No. 3 pp.456-461, 2008 年３月.

International Conferences (with Review Process):

1. S. Ataka, B. K. Kim and M. Gen, “Study on Two-Stage Transportation Planning with Inventory and Exclusionary Side Constraints by Boltzmann Random Key-based GA”, *Proc. of Artificial Neural Networks in Engineering 2009*, USA, pp.307-314, November 2009.
2. S. Ataka, B. Kim and M. Gen, “Study on Flexible Logistics Network Model with Inventory using Boltzmann Random Key-based Genetic Algorithm”, *International Conference on Intelligent Manufacturing & Logistics Systems IML2009 & Symposium on Group Technology and Cellular Manufacturing GT/CM2009*, Japan, pp.82-89, February 2009.
3. S. Ataka and M. Gen, “Study for Flexible Logistics Network Model with Inventory”, *Proc. of Artificial Neural Networks in Engineering 2008*, USA, pp.413-420, November 2008.
4. S. Ataka and M. Gen, “The Study for Transportation Planning Considered the Inventory Using Hybrid Genetic Algorithm”, *Proc. of GECCO2008*, USA, pp.1689, July 2008.
5. S. Ataka and M. Gen, “Study for Two-stage Transportation Planning with Inventory and Exclusionary Side Constraints by Genetic Algorithm”, *Proc. of*

International Symposium on Management Engineering, Japan, pp.127-133, March 2008.

6. S. Ataka and M. Gen, “A Hybrid Genetic Algorithm for Flexible Logistics Network Design Problem”, *Proc. of The 8th Asia Pacific Industrial Engineering & Management System*, Taiwan, December 2007.
7. S. Ataka and M. Gen, “Study for Multi-Product Logistics Network Design Problem with Constraints of Delivery Course”, Proc. of Artificial Neural Networks in Engineering 2007, USA, pp. 247-252, November 2007.
8. S. Ataka and M. Gen, “Solving Exclusionary Side Constrained Logistics Network by Using Priority-based GA”, *Proc. of IEEE Congress on Evolutionary Computation*, Singapore, pp.4483-4488, September 2007.

National Publications:

1. 安高真一郎, 玄光男, “在庫と時間を考慮したフレキシブル・ロジスティクスネットワークモデルの研究”, 日本, 日本知能情報ファジィ学会 第 24 回ファジィシステムシンポジウム pp.192-195, 2008 年９月.
2. 安高真一郎, 玄光男, ”在庫と排他的制約を考慮した配送計画モデルの一解法”, 日本, 日本経営工学会平成 20 年度春季大会, pp136-137, 2008 年５月.
3. 安高真一郎, 玄光男, “ハイブリッド型遺伝的アルゴリズムを用いた配送経路に制約を伴う多品種ロジスティクスネットワーク問題の研究”, 平成 19 年 電気学会 電子・情報・システム部門大会, 日本, pp.441-446, 2007 年９月.
4. 安高真一郎, 玄光男, “遺伝的アルゴリズムを用いた排他的制約を伴う輸送問題の一解法”,日本, 第 17 回ソフトサイエンス・ワークショップ, pp.154-157, 2007 年３月.