CHAPTER 3

METHOD OF APPROACH

This chapter discussed the framework of the study. The study aimed to determine the suitable assignment in a facility location problem. However, there are several types of the facility location problem. Each type of problem has different characteristic. For this thesis, a single-source capacitated facility location problem (SSCFLP) has been focused and the proposed algorithm will be presented to solve the problem. First of all the characteristics of the problem have been discussed and the proposed algorithm has been then constructed. Moreover, the similarity of generalized assignment problem (GAP) and single-source capacitated facility location problem (SSCFLP) has also been described since the application of ant colony optimization (ACO) on generalized assignment problem (GAP) can be applied to solve the facility location problem.

3.1 A Single-Source Capacitated Facility Location Problem

The single-source capacitated facility location problem is a special case of the capacitated facility location model. In the past years, many researchers have paid attention on this kind of problem. Many solution methods have been proposed to solve the problem. While many researchers have been proposed the new method to solve the original form of problem and compared the results with other methods, some of researchers have proposed the modification models of SSCFLP and presented the solution method to solve the proposed model. In this chapter the original model of SSCFLP has been presented and the model of Nozick L. K. and Turnquist M. A. (2001) has been described before adding some variables and concepts into the model in the next chapter.

3.1.1 The Original Single-Source Capacitated Facility Location Model

To formulate the mathematical model of the SSCFLP, the notations will be presented as follow:

m = Number of potential facilities,

n = Number of customers,

a_j = Demand of customer j, $(j = 1, 2, \ldots, n)$

f_i = Fixed cost for using/opening facility i, $(i = 1, 2, \ldots, m)$

b_i = Capacity of facility i ,

c_{ij} = Cost for assigning customer j to facility i,

All coefficients are assumed to be nonnegative and integral. Define the decision variable as follow:

$y_i = \begin{cases} 1 & \text{if a facility } i \text{ is opened} \\ 0 & \text{otherwise} \end{cases}$

$x_{ij} = \begin{cases} 1 & \text{if a facility } i \text{ serves customer } j \\ 0 & \text{otherwise} \end{cases}$

The problem can be stated by the following integer program:

$$\text{Minimize} \quad Z = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} f_i y_i \tag{1}$$

Subject to,

$$\sum_{j=1}^{n} a_j x_{ij} \le b_i y_i, \qquad i = 1,2,\ldots,m \tag{2}$$

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad j = 1,2,\ldots,n \tag{3}$$

$$x_{ij} - y_i \le 0, \quad i = 1, 2, \ldots, m, \ j = 1, 2, \ldots, n \tag{4}$$

$$x_{ij} \in \{0, 1\}, \qquad i = 1, 2, \ldots, m, \ j = 1, 2, \ldots, n \tag{5}$$

$$y_i \in \{0, 1\}, \qquad i = 1, 2, \ldots, m \tag{6}$$

3.1.2 The model of Nozick L. K. and Turnquist M. A. (2001)

Nozick L. K. and Turnquist M. A. (2001) have presented the model of single-source facility location problem by following the fixed-charge location model of Daskin (1995). There model was not concerned the capacity at each facility as constraint. Let

m = Number of customers,

n = Number of candidate sites,

h_i = Demand at location i, $(i = 1, 2, \ldots, m)$

f_i = Fixed cost of creating a facility at candidate site j, $(j = 1, 2, \ldots, n)$

d_{ij} = Distance from demand location i to candidate site j,

α = Cost per unit distance per unit demand,

W = A weight,

$X_j = \begin{cases} 1 & \text{if a facility } i \text{ is located at candidate site } j \\ 0 & \text{otherwise} \end{cases}$

$$Y_{ij} = \begin{cases} 1 & \text{if demands at } i \text{ served by a facility at candidtae site } j \\ 0 & \text{otherwise} \end{cases}$$

The mathematical model is to minimize cost of establishing facilities and the cost of assigning customers to facilities can be expressed as follows:

$$\text{Minimize} \sum_{j=1}^{n} f_j X_j + \alpha \sum_{i=1}^{m} \sum_{j=1}^{n} h_i d_{ij} Y_{ij} \tag{7}$$

Subject to

$$\sum_{j=1}^{n} Y_{ij} = 1 \qquad i = 1, 2, \ldots, m \tag{8}$$

$$Y_{ij} \leq X_j \qquad i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n \tag{9}$$

$$X_j \in (0, 1) \qquad j = 1, 2, \ldots, n \tag{10}$$

$$Y_{ij} \in (0, 1) \qquad i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n \tag{11}$$

Generally, the objective function of SSFLP to be minimized has consisted of the fixed costs and variable costs. The fixed cost is the cost of establishing facilities and variable cost is the cost of assigning customers to opened facilities or transportation cost. Moreover, this thesis has also included the objective function that will minimize uncovered demand cost into the fixed-charge location model.

A mathematical model tries to maximize the coverage to ensure that the proportion of demand within a specified "coverage" distance of a facility will be met. Delivery is then guaranteed for the set of customers within a certain radius (e.g., 400 k.m.) of the facility. The objective of maximizing the proportion of total demand is covered by a set of N facilities has been first described by Church and ReVelle (1974). An equivalent model which minimizes uncovered demand has been formulated by Hillsman (1984). This formula has involved the integration of coverage maximization and cost minimization. Defining the following variable:

$$q_{ij} = \begin{cases} 1 & \text{if a facility located at candidtae site } j \text{ cannot cover demand at } i \\ 0 & \text{otherwise} \end{cases}$$

the minimization of uncovered demand can be expressed as follows:

$$\text{Minimize} \sum_{i=1}^{m} \sum_{j=1}^{n} h_i q_{ij} Y_{ij} \tag{12}$$

Subject to

$$\sum_{j=1}^{n} Y_{ij} = 1 \qquad i = 1, 2, \ldots, m \tag{13}$$

$$\sum_{j=1}^{n} X_j = N. \tag{14}$$

$$Y_{ij} \leq X_j \qquad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n \tag{15}$$

$$X_j \in (0,1) \qquad j = 1, 2, \ldots, n \tag{16}$$

$$Y_{ij} \in (0,1) \qquad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n \tag{17}$$

$$q_{ij} \in (0,1) \qquad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n \tag{18}$$

Constraints (13) and (15) - (17) in this model are identical to constraints (8) - (11) in the fixed-charge location model, allowing integration of the total cost objective and the coverage objective. More specifically

$$\text{Minimize} \sum_{j=1}^{n} f_j X_j + \sum_{i=1}^{m} \sum_{j=1}^{n} \{ \alpha d_{ij} + \mathrm{W}\, q_{ij} \} h_i Y_{ij} \tag{19}$$

Subject to

$$\sum_{j=1}^{n} Y_{ij} = 1 \qquad i = 1, 2, \ldots, m \tag{20}$$

$$Y_{ij} \leq X_j \qquad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n \tag{21}$$

$$X_j \in (0,1) \qquad j = 1, 2, \ldots, n \tag{22}$$

$$Y_{ij} \in (0,1) \qquad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n \tag{23}$$

$$q_{ij} \in (0,1) \qquad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n \tag{24}$$

The objective function (19) to be minimized consists of the cost of establishing facilities, the cost of assigning customers to open facilities and the cost of uncovered demand. Constraint set (20) can be referred to as the demand constraints (or the customers constraints), and ensures that each customer is assigned to exactly one facility. Constraints set (21) ensure that the assignments are made only to opened facilities. Constraints set (22), (23) and (24) are the integrality constraints.

3.2 Ant Colony Optimization Algorithm

For this study, ant colony optimization (ACO) has been proposed to solve the problem. Therefore, the general concept of ACO algorithm and the applications have to be described here. However, applying ant colony optimization on single-source capacitated facility location problem will be presented in next chapter.

The ant colony optimization (ACO) paradigm has been studied by Colorni, Dorigo and Maniezzo (1991a, 1991b), Dorigo, Maniezzo, and Colorni (1996) and Dorigo and Di Caro (1999). ACO is a cooperative algorithm which is inspired by the foraging behavior of real ants. Ants lay down some quantity of an aromatic substance, known as pheromone, on their way to food and on their way back to the nest. Ants choose to follow a pheromone trail with a probability proportional to the pheromone trail intensity. Research on real ants has shown that such a trail following behavior allows the ants to identify shortest path between a food source and their nest. The applications of ACO have been successfully studied in many areas such as TSP, QAP, GAP, etc.

3.2.1 Introduction to Ant Colony Optimization

In ant algorithms a colony of relatively simple agents, called ants, efficiently carries out complex tasks such as resource optimization and control. Ants act according to a sequential decision making scheme in which a stochastic decision policy is adaptively built and evaluated by the ants themselves.

As Dorigo et al. (2000) have pointed out in the first paper of this special issue, the indirect stigmergic communication among the ants is the key characteristic of ant algorithms. *Stigmergy* defines a paradigm of indirect and asynchronous communication mediated by an environment. While carrying out their own tasks, ants deposit some chemical substance (called pheromones) or induce some other physical modifications of the environment. These modifications change the way of environment (and in a way, the problem under consideration) is sensed by the other ants in the colony, and implicitly act as signals triggering other ants' behaviors that again generate new modifications hat will stimulate other ants and so on.

The use of a colony of agents communicating in stigmergic way turned out to be very effective in distributed and network environments, because of its asynchronous and indirect nature, and it revealed to be a very effective way to communicate 'memory' information when solving combinatorial optimization problems. In fact, currently there are many successful applications of ant algorithms to adaptive routing in communications networks and combinatorial optimization problem.

Ant colony optimization (ACO) is a recently developed, population-based approach which has been successfully applied to several *NP*-hard combinatorial optimization problems. As the name suggests, ACO has been inspired by the behavior of real ant colonies, in particular, by their foraging behavior. One of its main ideas is the indirect communication among the individuals of a colony of agents, called (*artificial*) ants, based on an analogy with trails of a chemical substance, called pheromone, which real ants use for communication. The (artificial) pheromone trails are a kind of distributed numeric information (called stigmergic information) which is modified by the ants to reflect their experience accumulated while solving a particular problem. The concept of stigmergic is a particular form of indirect communication used by social insects to coordinate their activities. By exploiting the stigmergic approach to coordination, researchers have been able to design a number of successful algorithms in such diverse application fields as combinatorial optimization, routing in communication networks, task allocation in a multi-robot system, exploratory data analysis, graph drawing and partitioning and so on.

3.2.2 Literature review on Ant Colony Optimization

An ant colony optimization approach has been proposed to solve the SSCFL model. From the past researches, ant colony optimization has been studied in several areas. The table below presents the current applications of ant colony optimization derived from Dorigo et al. (2000).

Table 3.1 Current applications of ACO algorithms (From Dorigo et al., 2000)

Problem name	Authors	Algorithm name	Year
Traveling salesman	Dorigo, Maniezzo and Coloni	AS	1991
	Gambardella and Dorigo	Ant-Q	1995
	Dorigo and Gambardella	ACS and ACS-3-opt	1996
	Stützle and Hoos	AS	1997
	Bullnheimer, Hartl and Strauss	AS_{rank}	1997
Quadratic assignment	Maniezzo,Coloni and Dorigo,	AS-QAP	1994
	Gambardella, Taillard and Dorigo	HAS-QAP	1997
	Stützle and Hoos	AS-QAP	1997
	Maniezzo	ANTS-QAP	1998
	Maniezzo and Coloni	AS-QAP	1999
Scheduling problems	Coloni, Dorigo and Maniezzo,	AS-JSP	1994
	Stützle	AS-FSP	1997
	Bauer et al.	ACS-SMTTP	1999
	Den Besten, Stützle and Dorigo	ACS-SMTWTP	1999
Vehicle routing	Bullnheimer, Hartl and Strauss	AS-VRP	1997
	Gambardella, Taillard and Agazzi	HAS-VRP	1999
Connection-oriented network routing	Schoonderwoerd et al.	ABC	1996
	White, Pagurek and Oppacher	ASGA	1998
	Di Caro and Dorigo	AntNet-FS	1998

Table 3.1 Current applications of ACO algorithms (From Dorigo et al., 2000, continued)

Problem name	Authors	Algorithm name	Year
Connection-oriented network routing	Bonabeau et al.	ABC-smart ants	1998
Connection-less network routing	Di Caro and Dorigo	AntNet and AntNet-FA	1997
Connection-less network routing	Subramanian, Druschel and Chen	Regular ants	1997
	Heusse et al.	CAF	1998
	Ven der Put and Rothkrantz	ABC-backward	1998
Sequential ordering	Gambardella and Dorigo	HAS-SOP	1997
Graph coloring	Costa and Hertz	ANTCOL	1997
Shortest common supersequence	Michel and Middendorf	AS-SCS	1998
Frequency assignment	Maniezzo and Carbonaro	ANTS-FAP	1998
Generalized assignment	Ramalhinho Lourenço and Serra	MMAS-GAP	1998
Multiple knapsack	Leguizamón and Michalewicz	AS-MKP	1999
Optical network routing	Navarro Varela and Sinclair	ACO-VWP	1999
Redundancy allocation	Liang and Smith	ACO-RAP	1999

Recently, the ACO metaheuristic has been proposed to provide a unifying framework for most applications of ant algorithms to combinatorial optimization problems. Algorithms which actually are instantiations of the ACO metaheuristic will be called ACO algorithms. The first ACO algorithm, called ant system (AS), has been applied to the traveling salesman problem (TSP). It has given encouraging results, yet its performance was not competitive with state-of-the-art algorithms for the TSP. Therefore, one important focus of research on ACO algorithms has been the introduction of algorithmic improvements to achieve a much better performance.

Applications of such ant algorithms can be divided into two classes:

1) Ant algorithms for static problem. In static problems the key-points of the problem are defined at the beginning and do not change while the problem is being solved. Examples of applications of static combinatorial optimization problems are:

(a) ***Traveling Salesman Problem*** where a salesman must find the shortest route to visit a given number of cities, and each city exactly once.

(b) ***Quadratic Assignment Problem***, the problem of assigning n facilities to n locations so that the costs of the assignment are minimized.

(c) ***Job-Shop Scheduling Problem*** where given a set of machines and a set of jobs–operations must be assigned to time intervals in such a way that no two jobs are processed at the same time on the same machine and the maximum of the completion times of all operations are minimized.

(d) ***Vehicle Routing Problem***, the object is to find minimum cost vehicle routes such that (a) every customer is visited exactly once by exactly one vehicle, (b) for every vehicle the total demand does not exceed the vehicle capacity, (c) the total tour length of each vehicle does not exceed a given limit, and (d) every vehicle starts and ends its tour at the same position (the depot).

(e) ***Shortest Common Supersequence Problem*** where given a set of strings over an alphabet a string of minimal length that is a supersequence of each string of the given set has to be found (a supersequence S of string A can be obtained from A by inserting zero or more characters in A).

(f) ***Graph-Coloring Problem*** which is the problem of finding a coloring of a graph so that the number of colors used is minimal.

(g) ***Sequential Ordering Problem*** which consists of finding a minimum weight Hamiltonian path on a direct graph with weights on the arcs and on the nodes, subject to precedent constraints among the nodes.

(h) ***Generalized Assignment Problem*** which consists of assigning a set of tasks to a set of agents with minimum cost. Each agent has a limited amount of a single resource and each task must be assigned to one only one agent, requiring a certain amount of the agent's resource.

2) Ant algorithms for dynamic combinatorial optimization problems. In dynamic problems the problem changes as a function of itself, thus the algorithms have been used to solve such problems must be able to adapt "online" to changes. The main focus of applications to dynamic combinatorial optimization problems is on communication networks especially routing problems. Routing answers the question, how to direct data traffic (e.g. phone calls) through a network (i.e. which node to choose next by a data packet entering the network). Routing mainly consists of building, using and updating routing-tables. Implementations for communication networks can be divided in two classes:

(a) ***Connection-Oriented Network Routing*** where all packets of the same session follow the same path are selected by a preliminary setup phase, and

(b) ***Connectionless Network Routing*** where data packets of the same session can follow different paths (Internet-type network).

3.2.3 Solving the Problem by Ant Colony Optimization

The applications of ant colony optimization have been proposed to solve many areas of problem. The first application has been reported by Dorigo, Maniezzo and Coloni, it was called ant system and applied to the traveling salesman problem (TSP). After the first application has been studied, many researchers have applied the concept of ant colony optimization to solve many kinds of problem. They have developed the new idea to solve their problem/model based on the original concept on ant colony optimization. Some technique has been added into the general concept of ACO for improving the effectiveness. Ant colony optimization has been inspired by the behavior of real ants. The concept of ant colony optimization comes from the way to search food and find the way back to the nest of ants. The transposition of this food searching behavior into an algorithmic framework for solving combinatorial optimization problems has been obtained through an analogy between:

- The search arc of the real ants and the set of feasible solutions to the combinatorial problem;
- The amount of food associated with a source and the objective function;
- The pheromone trail and an adaptive memory.

A standard ant system is schematically described as follows:

1. Initialize the pheromone trail.

2. While a stopping criterion is not met do:

2.1 for each ant construct a new solution using the current pheromone trail and an evaluation of the partial solution being constructed;

2.2 update the pheromone trail.

The most important component of an ant system is the management of pheromone trails. In a standard ant system, pheromone trails are used in conjunction with the objective function to guide the construction of new solutions. Once a solution has been produced, a standard ant system updates the pheromone trail as follows: first all trails are weakened to stimulate the evaporation of pheromone then, pheromone trails that correspond to components that were used to construct the resulting solution are reinforced, taking into consideration the quality of this solution.

3.3 Generalized Assignment Problem

From literatures, the generalized assignment problem (GAP) is similar to a single-source capacitated facility location problem (SSCFLP). Therefore, ant colony optimization

(ACO) on generalized assignment problem (GAP) has been focused for solving SSCFLP. In this section the general concepts of GAP have been described for guiding to solve the SSCFLP.

3.3.1 Overview of Generalized Assignment Problem

The generalized assignment problem (GAP) has considered the minimum cost assignment of *n* jobs to *m* agents such that each job is assigned to one and only one agent subject to capacity constraints on the agents. The GAP has applications in areas like computer and communication networks, location problems, vehicle routing and machine scheduling. Initial interest scopes of Ramalhinho Lourenço and Serra (1998) have been arisen from two real applications in the areas of health services, resource assignment problems and pure integer capacitated plant location problems.

The GAP consists in assigning at minimum total cost a set of tasks to a set of agents, with limited resource capacity. Each task must be assigned to one and only one agent requiring a certain amount of the agent's resource. Fisher et al. (1986) have proved that the generalized assignment problem (GAP) is a *NP*-completed problem. Moreover, the problem of deciding if there exits a feasible solution is *NP*-hard, Sahni and Gonzalez (1976). Osman (1995) has presented the survey of many real-life applications.

The generalized assignment problem can be formulated as an integer program, as presented below. Use the following notation:

I = The set of tasks (i =1,2,…,n),

J = The set of agents (j = 1,2,…,m),

b_j = Resource capacity of agent j,

a_{ij} = Resource needed if task i assigned to agent j,

c_{ij} = Cost of task i if assigned to agent j.

$$x_{ij} = \begin{cases} 1 & \text{if task } i \text{ is assigned to agent } j \\ 0 & \text{otherwise} \end{cases}$$

Assume that $a_{ij} \le b_j$ and $\sum_{i=1}^{n} a_{ij} > b_j$. The problem of GAP is as follow:

$$\text{Minimize} \quad f(x) = \sum_{j=1}^{m}\sum_{i=1}^{n} c_{ij}x_{ij} \tag{25}$$

Subject to,

$$\sum_{i=1}^{n} a_{ij}x_{ij} \le b_j, \quad j = 1, 2, \ldots, m \tag{26}$$

$$\sum_{j=1}^{m} x_{ij} = 1, \qquad i = 1, 2, \ldots, n \tag{27}$$

$$x_{ij} \in \{0, 1\}, \qquad i = 1, 2, \ldots, m, \;\; j = 1, 2, \ldots, n \tag{28}$$

Constraints (26) are the resource capacity of the agents, constrains (27) guarantee that each task is assigned to one agent, and constraints (28) are the integrality constraints.

In the other hand, the generalized assignment problem (GAP) has also used to find a maximal profit assignment of n tasks (indexed by j_1, $j_2, \ldots, j_n$) to m agents (indexed by $i_1, i_2, \ldots, i_m$). Let $I = \{i_1, i_2, \ldots, i_m\}$ and $J = \{j_1, j_2, \ldots, j_n\}$. Let b_i be the resource availability of agent i. Let a_{ij} be the amount of resource required by agent i to perform task agent j, let p_{ij} be the profit assigned, and let x_{ij} be a binary variable that indicates whether task j is assigned to agent i. Each task must be assigned to one agent without exceeding his resource availability. The mathematical model of GAP is:

$$\text{Maximize } f(x) = \sum_{i \in I}^{m} \sum_{j \in J}^{n} p_{ij} x_{ij} \tag{29}$$

Subject to,

$$\sum_{j \in J}^{n} a_{ij} x_{ij} \le b_i, \qquad i \in I \tag{30}$$

$$\sum_{i \in I}^{m} x_{ij} = 1, \qquad j \in J \tag{31}$$

$$x_{ij} \in \{0, 1\}, \qquad i \in I, \quad j \in J \tag{32}$$

For single-source capacitate facility location problem (SSCFLP), the first case of generalized assignment problem (GAP) has been considered since the objective function is to minimize the total cost. In the next section the correlation between SSCFLP and the generalized assignment problem (GAP) will be presented.

3.3.2 Generalized Assignment Problems and SSCFLP

From the previous section, the general idea of generalized assignment problem (GAP) has been presented. The first case of GAP is similar to SSCFLP. Therefore, the SSCFLP can be considered as a GAP. The similarity of GAP and SSCFL has been presented as following:

1) An objective function of SSCFLP is similar to GAP. From the equation (1) the objective function is to minimize the total cost of establishing facility, cost of assigning facility to customer. Therefore, equation (1) is similar to equation (25) in GAP, the objective function is also to minimize the total cost of assigning tasks to agents.

2) The special characteristic of single-source capacitated facility location problem (SSCFLP) is the condition which each customer can only be supplied from one facility in equation (3). This condition as same as the condition of GAP in equation (27) which each task is assigned to one agent. These two equations are the same.

3) The GAP, also consider capacity constraint as same as the single-source capacitated facility location problem (SSCFLP) on the other hand the difference is equation (2) of SSCFLP has been referred to as the capacity constraints (or the facility constraints) which ensures that the customer demand is served by a certain facility does not exceed its capacity. The equation (26) of GAP has represented the resource capacity of the agents.

4) The variables in GAP and SSCFLP are the same. From equation (5) and equation (28), the meaning of these two equations is the integrality constraints. x_{ij} = 1 if facility (task) i assigned to customer (agent) j and 0, otherwise.

From the similarity of basic characteristics of GAP and SSCFLP presented above, it can be concluded that SSCFLP can be considered as a GAP although some conditions of SSCFLP are different from GAP constrains.