

© Yonglin Ren, 2011

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Yonglin Ren

Entitled: *Metaheuristics for multiobjective capacitated location allocation on logistics networks*

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Quality Systems Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

.....Dr.Andrea Schiffareouva (Chair)

.....Dr. Masoumeh Kazemi (External Examiner)

.....Dr. Zhigang Tian (Internal Examiner)

.....Dr. Anjali Awasthi (Supervisor)

Approved by

Chair of Department or Graduate Program Director

_____20_____

Dean of Faculty

ABSTRACT

Metaheuristics for multiobjective capacitated location allocation on logistics networks

Yonglin Ren

Logistics is vital to sustaining many industrial, commercial, and administrative activities. It is often composed of the logistics service providers and the customers being serviced. The goal of service providers is to maximize revenues by servicing customers efficiently within their preferred timelines. To achieve this goal, they are often involved in activities of location-allocation planning, that is, which logistics facilities be opened, where they should be opened, and how customer allocations should be performed to ensure timely service to customers at least delivery costs to logistics operators.

In this thesis, we address the multiobjective capacitated location allocation problem on logistics networks. The distinction between the location allocation problem treated in this thesis and the traditional location allocation problem lies in its multiobjective and dynamic nature. The multiple objectives considered are travel time, travel distance, travel cost etc. and developed based on practical constraints such as presence of congestion, timing and access restrictions imposed by municipal administrations in urban areas etc. The dynamic aspect means the location allocation results are not fixed forever but vary

with change in municipal access or timing regulations, congestion, or land, material and labor costs on logistics networks.

Four metaheuristics namely Genetic algorithms (GA), Simulated annealing (SA), Tabu search (TS), and Ant colony optimization (ACO) based solution approaches are presented to treat the multiobjective facility location allocation problem. Two cases are studied. In the first case, opening costs of the facilities and only one criterion (distance) is used. In the second case, opening costs of the facilities and multiple criteria (distance, travel cost, travel time) are used. The proposed approaches are tested under various problem instances to verify and validate the model results.

I dedicate this work
to Anne Xinyi and
Xinyue.

Acknowledgements

I would like to acknowledge my supervisor Dr. Anjali Awasthi for her guidance, perseverance, hard work and continual support towards successful completion of this thesis. Her patience, diligence, and encouragement will affect my whole life.

I would also like to thank my parents and other family members for always supporting me in my career endeavors. Without their emotional and moral support, this hard work would not have been possible.

I would also like to thank my friends for helping me check the English of the thesis and sharing research discussions in lab and classroom. Their company always kept my spirits high and cheerful.

I would also like to thank Concordia University and CIISE for offering me the opportunity to pursue my Masters in Quality Systems Engineering and providing financial support to accomplish the research pursued in this thesis.

Last but not the least, I am greatly indebted to my wife and two lovely children who always welcome me with warm smile and stand beside me in all aspects of life. I dedicate this work to them for their unconditional support and love.

Table of Contents

List of Figures	xi
List of Tables	xii
List of Acronyms	xiv
 Chapter 1: Introduction	 1
1.1 Background	1
1.2 Problem description	3
1.3 Thesis contribution	4
1.4 Thesis outline	4
 Chapter 2: Literature Review	 5
2.1 Classification of location allocation problem	5
2.1.1 Classifications on customers demand	6
2.1.2 Classifications on facilities	6
2.1.3 Classifications on the physical space or locations	6
2.1.4 Classifications based on location objectives	8
2.2 Models for Location Allocation problem	9
2.2.1 Exact approaches	9
2.2.2 Data analysis	11

2.2.3 Simulation	12
2.2.4 Multi-criteria decision analysis	13
2.2.5 Heuristics	14
2.2.6 Metaheuristics	15
2.2.7 Hybrid Approaches or Combinations of the above	16
Chapter 3: Problem definition	18
Chapter 4: Solution Approach	22
4.1 Genetic algorithms (GA) for location & allocation	24
4.1.1 Method Description	25
4.1.2 High level pseudocode for GA	28
4.1.3 Example	29
4.1.4 Advantages of Genetic Algorithms	30
4.1.5 Limitations of Genetic Algorithms	31
4.2 Simulated annealing (SA) for location allocation	32
4.2.1 Method Description	32
4.2.2 High level pseudocode for SA	34
4.2.3 Example	35
4.2.4 Advantages of Simulated Annealing	36
4.2.5 Limitations of Simulated Annealing	36
4.3 Tabu Search (TS) for location allocation	36

4.3.1 Method Description	37
4.3.2 High level pseudocode for TS	39
4.3.3 Example	40
4.3.4 Advantages of TS	41
4.3.5 Limitations of TS	41
4.4 Ant Colony Optimization (ACO) for location allocation	41
4.4.1 Method Description	42
4.4.2 High level pseudocode for ACO	45
4.4.3 Example	46
4.4.4 Advantages of Ant Colony Optimization	47
4.4.5 Limitations of Ant Colony Optimization	47
Chapter 5 Numerical Application	48
5.1 Location & allocation using single factor	48
5.1.1 Input Data	48
5.1.2 Application of proposed metaheuristics	51
5.1.2.1 Genetic Algorithm (GA)	51
5.1.2.2 Simulated Annealing (SA)	52
5.1.2.3 Tabu Search (TS)	54
5.1.2.4 Ant Colony Optimization (ACO)	56
5.1.2.5 Comparison of GA, SA, TS, and ACO	57
5.2 Location & allocation problem using multifactors	59

5.2.1 Input Data	59
5.2.2 Application of metaheuristics	62
5.2.2.1 Genetic Algorithm (GA)	62
5.2.2.2 Simulated Annealing (SA)	64
5.2.2.3 Tabu Search (TS)	66
5.2.2.4 Ant Colony Optimization (ACO)	67
5.2.2.5 Comparison of GA, SA, TS, and ACO	69
5.3 Model Verification	70
5.4 Model Validation	77
Chapter 6: Conclusions and future works	81
6.1 Conclusions	81
6.2 Future work	82
References	84

List of Figures

Figure 3.1 Network of logistics depots and customers	19
Figure 4.1 Solution Representation for location allocation problem	25
Figure 5.1 Convergence of GA results for single factor location- allocation problem	51
Figure 5.2 Convergence of SA results for single factor location- allocation problem	53
Figure 5.3 Convergence of TS results for single factor location- allocation problem	55
Figure 5.4 Convergence of ACO results for single factor location- allocation problem	56
Figure 5.5 Convergence of GA results for multifactor location- allocation problem	63
Figure 5.6 Convergence of SA results for multifactor location- allocation problem	65
Figure 5.7 Convergence of TS results for multifactor location- allocation problem	66
Figure 5.8 Convergence of ACO results for multifactor location- allocation problem	68

List of Tables

Table 4.1 Initial population for genetic algorithm	30
Table 4.2 Solutions for Simulated Annealing	35
Table 5.1 Input distance, capacity and demand data	49
Table 5.2 Opening costs for logistics depots	49
Table 5.3 Normalized distance values	50
Table 5.4 Normalized opening cost values for logistics depots	51
Table 5.5 Objective function results for single factor location allocation problem using GA	52
Table 5.6 Initial and Final Solution for single factor location allocation problem using GA	52
Table 5.7 Objective function results for single factor location allocation problem using SA	53
Table 5.8 Initial and Final Solution for single factor location allocation problem using SA	54
Table 5.9 Objective function results for single factor location allocation problem using TS	55
Table 5.10 Initial and Final Solution for single factor location allocation problem using TS	55
Table 5.11 Objective function results for single factor location allocation problem using ACO	57
Table 5.12 Initial and Final Solution for single factor location allocation problem using ACO	57
Table 5.13 Comparison of Location allocation results for single factor problem	58
Table 5.14 Comparison of performance results for single factor location allocation problem	59
Table 5.15 Time Matrix	60
Table 5.16 Cost matrix	61
Table 5.17 Normalized data for multifactor cost, time and distance	62
Table 5.18 Objective function results for multifactor location allocation problem using GA	63
Table 5.19 Initial and Final Solution for multifactor location allocation problem using GA	64
Table 5.20 Objective function results for multifactor location allocation problem using SA	65
Table 5.21 Initial and Final Solution for multifactor location allocation problem using SA	66
Table 5.22 Objective function results for multifactor location allocation problem using TS	67
Table 5.23 Initial and Final Solution for multifactor location allocation problem using TS	67
Table 5.24 Objective function results for multifactor location allocation problem using ACO	68

Table 5.25 Initial and Final Solution for multifactor location allocation problem using ACO	69
Table 5.26 Comparison of model results for multifactor location allocation problem	69
Table 5.27 Comparison of model performance for multifactor location allocation problem	70
Table 5.28 Customer Demand data	71
Table 5.29 Opening costs of facilities	72
Table 5.30 Average transit time (in minutes) between the depot i and customer j	72
Table 5.31 Unit shipping cost (in dollars) and demand (in units of products)	73
Table 5.32 Scenarios for verification	73
Table 5.33 Objective function value	74
Table 5.34 Computation Time (in seconds)	75
Table 5.35 Location Allocation Results for Scenario 1 (in seconds)	75
Table 5.36 Location Allocation Results for Scenario 2 (in seconds)	76
Table 5.37 Location Allocation Results for Scenario 3 (in seconds)	77
Table 5.38 Weights scenario description	79
Table 5.39 Transit Time Results	79
Table 5.40 Shipping Cost Results	80

List of Acronyms

GA	Genetic Algorithms
TS	Tabu Search
ACO	Ant Colony Optimization
SA	Simulated Annealing
LA	Location Allocation
AHP	Analytic Hierarchy Process
CST	Central Sales Tax
DC	Distribution Centers
MCDA	Multi-Criteria Decision Analysis
ANP	Analytical Network Process
TOPSIS	Technique for Order Preference by Similarity to an Ideal Solution
SAW	Simple Additive Weighting Model
NSGA II	Non-dominated Sorting Genetic Algorithm II
ESA	Evolutionary Simulated Annealing
HC	Hill Climbing
SVR	Support Vector Ordinal Regression
SSCLA	Single Source Capacitated Location Problem
MACO	Multiple ant colony optimization
UCLA	Uncapacitated continuous location–allocation

Chapter 1:

Introduction

1.1 Background

Location planning of logistics depots and customer allocation are important decisions in supply chain network design (Ambrosino and Scutella 2005, Drezner and Hamacher, 2002). A carefully planned network design positively impacts the economics of business organizations and their competitiveness in national and international markets. Improper planning can lead to poor service quality towards customers, long delivery times, and high investment and maintenance costs for the logistics operators, which is detrimental to their business operations and profitability.

The problem treated in this thesis is motivated by distribution network design in urban areas under congestion. The decisions concerned are location planning of logistics depots and allocation of clients to the opened logistics depots. In urban environment, opening of logistics depots and clients allocation is affected by a number of factors such as presence of congestion, land and labor cost, proximity to clients, presence of municipal regulations such as time restrictions, access restrictions etc. Therefore, the problem of how many logistics depots to be opened, where to locate them, and how to cluster customers and allocate them to logistics depots etc. is multiobjective and dynamic in nature and not a

static or one-time decision as considered in several studies available in literature in this direction. In practice, the LA decision involves consideration of multiple factors such as distance, travel cost, travel time etc. which are continually varying over time and therefore, the location allocation problem we are treating in this thesis is multi-objective and dynamic in nature.

Solutions to location allocation problem have been mainly investigated under two main cases. In the first case, the location planning of logistics depots is performed first and customer allocations are done. In the second case, the customer zones are formed first and then logistical facilities or logistics depots are located at center of zones to ensure better coverage and service for customers.

Distributing goods to customers from several logistics depots produces the problem of optimizing the delivery process. Managers or Logistics operators face the problem of reducing delivery costs, that is, how to ensure efficient delivery processes considering multiple factors such as travel cost, travel time, and travel distance, and how to integrate them altogether in optimizing overall costs for delivery of goods to customers. It is obvious that these problems are multiobjective in nature and therefore compromise solutions have to be found.

Most of the solutions to location & allocation problems have been approached in similar ways as those used for combinatorial optimization problems. If the number of logistics

depots and customers are small, the optimal solutions can be found using exact programming approaches. However, if the scale of problem is large, then exact approaches are not enough to provide satisfactory solutions in reasonable amount of time. Therefore, new types of solution approaches need to be developed to resolve large sized location-allocation problems.

Location-allocation problem is NP-hard problem (Azarmand and Neishabouri, 2009). In literature, metaheuristics have been shown to perform better than exact programming approaches to tackle larger NP-hard problems. In this thesis, we will address the multiobjective capacitated location allocation problem and develop solution approaches based on the following four metaheuristics.

- Genetic algorithms (GA)
- Simulated Annealing (SA)
- Tabu Search (TS)
- Ant Colony Optimization (ACO)

These metaheuristics will be discussed in detail in Chapter 3 and Chapter 4 of the thesis.

1.2 Problem Statement

The problem treated in this thesis is capacitated location allocation planning of logistics depots for distribution network design. This involves location planning of logistics depots and customer allocations considering facility opening costs and distribution costs to

customers under given capacity constraints of logistics depots and customer demands.

1.3 Thesis contribution

The thesis presents four metaheuristics namely GA, SA, TS and ACO for location allocation problem on logistics networks. The proposed metaheuristics were tested for different problem instances and the results were compared with other existing approaches available in literature. The strength of using the proposed metaheuristics is ability to generate good solutions under large problem instances. Besides, consideration of multiple criteria in allocation of clients to logistics depots provides practical solution to the problem under consideration.

1.4 Thesis outline

The rest of the thesis is divided as follows.

In Chapter 2, we present the literature review on the location allocation problem and available solution approaches.

In Chapter 3, we present the problem description with mathematical formulation.

In Chapter 4, we propose four metaheuristics (GA, SA, TS, and ACO) for capacitated location allocation on logistics networks.

In Chapter 5, we present numerical application of the proposed metaheuristics and perform verification and validation of model results.

In Chapter 6, we present the conclusions and directions for future research.

Finally, references conclude the thesis.

Chapter 2:

Literature Review

The Location-allocation (LA) problem involves locating an optimal set of facilities to satisfy customer demand at minimal transportation cost from facilities to customers (Love et al 1988, Ninlawan 2008). They have been applied in a number of areas such as location of warehouses, fast food outlets, gas stations, electric transformers, emergency healthcare facilities, production plants etc.

2.1 Classification of location allocation problem

There are four components that characterize any location allocation problem. According to Revelle and Eiselt (2005), they are (1) customers, who are presumed to be already located at points or on routes, (2) facilities that will be located, (3) a space in which customers and facilities are located, and (4) a metric that indicates distances or times between customers and facilities. Based on the studies by Scaparra and Scutellà (2001), Revelle and Eiselt (2005), Revelle et al (2008), Azarmand and Neishabouri (2009), Beaumont (1981), Love and Juel (1982), we classify the location allocation models into following main categories:

2.1.1 Classifications on Customers demand

Based on the certainty of information available about the customer demands, the models can be classified as deterministic or stochastic. If the number of customers, their locations and demands are known with certainty, the model is called deterministic. If the customer demands are modeled using probability distributions, the models are termed stochastic.

2.1.2 Classifications on Facilities

The Location-allocation models can be classified into single-facility or multi-facility depending upon the number of facilities to be located. In the contrary case, the number of facilities to be placed may not be known in advance. In such case, idea is to find the least number of facilities so that all demand points are covered within a prespecified distance standard (also called as location set covering model first introduced by Toregas et al. (1971).

If the facilities are limited by their capacities to serve customer demands, the models are termed capacitated otherwise called uncapacitated. The models can also be differentiated into single-service and multi-service types, based on whether the facilities can provide only one or many services.

2.1.3 Classification on the Physical Space or Locations

Based on the representation of space in which facilities are located, the location allocation

models can be classified into problems in planar (d -dimensional real space $|^d$) and network location problems each of which can be further sub-divided into continuous or discrete location problems (ReVelle and Eiselt, 2005). Distances in $|^d$ are most often derived from Minkowski distances, which are defined as a family of distances with a single parameter p . In particular, the ℓ_p distance between a point (a_i, b_i) and a point (a_j, b_j) with $i \neq j$ is defined as $d_{ij}^p = \left[|a_i - a_j|^p + |b_i - b_j|^p \right]^{1/p}$. For $p=1$, we obtain the rectilinear (or rectangular or Manhattan or ℓ_1) distance $d_{ij} = |a_i - a_j| + |b_i - b_j|$, and for $p=2$, we obtain the Euclidean (or straight line or ℓ_2) metric with $d_{ij}^2 = \sqrt{(a_i - a_j)^2 + (b_i - b_j)^2}$ and the Chebyshev (or “max”, or ℓ_∞) metric with $d_{ij}^\infty = \max\{|a_i - a_j|, |b_i - b_j|\}$. In contrast, the distances in network location problems are measured on the network itself, typically as the shortest route on the network of arcs connecting the two points.

Both planar problems and network problems can be further subdivided into continuous and discrete location problems. In continuous problems, the points to be sited can generally be placed anywhere on the plane or on the network. For example, placement of a helicopter for trauma pickup is a typical application of a continuous problem on a network. In discrete problems (Marin 2011), in addition to the points to be positioned, the facilities can conceptually be placed only at a limited number of eligible points on the plane or network.

2.1.4 Classifications based on location objectives

Traditionally, the location of facilities is done in a way so that the closer they are to the customers, the better the value of the objective function. Eiselt and Laporte (1995) call this objective to fall into the “pull” category. This normally involves maximizing the demands served (capture problem), minimizing sum of transportation costs (median problem) or minimizing the largest customer-facility distance (center problem). In contrast to facilities where closeness is desirable, there can also be ‘push’ objective where the goal is to “push” undesirable facilities as far from the customers as possible. Finally, a third class of objective is the achievement of equity. In such models, the objectives attempt to locate the facilities in such a way that the customer-to-facility distances are as similar to each other as possible. This equalization gives rise to the term “balancing objectives”. In other words, the distances from clients to the nearest facility may be bounded by some generally recognized distance standard.

ReVelle et al. (1970) proposed the private sector and public sector category for location problems. The private sector problems seek the sites that optimize some function of the monetary value associated with the location. In contrast, public sector problems seek facility sites that optimize the population's access. Clearly, there are many shades of gray between the extremes of “private” and “public”.

Using the above mentioned classifications, the location allocation problem treated in this thesis can be categorized into deterministic, multifacility, multiobjective, min-sum,

discrete, and capacitated.

2.2 Models for Location Allocation problem

The LA problem was first proposed by Cooper (1963) and spread to a weighted network by Hakimi (1964). The network LA problem and many models were presented by Badri (1999). Numerous approaches (Klose and Drexl 2005, Henrik and Robert 1982, Love et al. 2008, Bischoff and Dachert, 2009) have been developed over years to solve the location allocation problem which can be mainly classified into:

- Exact approaches
- Data analysis
- Simulation
- Muticriteria decision analysis
- Heuristics
- Metaheuristics
- Hybrid approaches or combinations of the above

These approaches are presented in detail as follows.

2.2.1 Exact approaches

The exact approaches or the mathematical programming approaches involve the use of techniques such as linear programming, integer programming, multiobjective optimization

etc. to arrive at optimal solutions. Mathematics, computation and business fields refer to selection of the best element from a set of available alternatives as Optimization or Computational programming. Steuer et al. (1986) simplifies the problem of solving minimization or maximization of real functions by systematically choosing values within an allowed set and proposes three types of optimizations: Multi-objective optimization, multi-model optimization, and dimensionless optimization. The multi-objective optimization (or programming), also known as multi-factors or multi-attribute optimization, is the process of simultaneously optimizing two or more conflicting objectives subject to certain constraints. The multi-modal optimization problems possess multiple good solutions. They could all be globally good (same cost function value) or there could be a mix of globally good and locally good solutions. Obtaining all (or at least some of) the multiple solutions is the goal of a multi-modal optimizer. Dimensionless optimization is used when the variables are dimensionless. In certain optimization problems the unknown optimal solution might not be a number or a vector, but rather a continuous quantity, for example a function or the shape of a body. Such a problem is an infinite-dimensional optimization problem, because, a continuous quantity cannot be determined by a finite number of certain degrees of freedom.

Exact solution methods were for a long time restricted to relatively small problem sizes. Branch and bound algorithms for LA problems were developed by Kuenne and Soland (1972), Ostresh (1973), Drezner (1984) and Rosing (1992), among others. Love and

Morris (1975) concentrated on rectilinear distances, and Love (1976) applied dynamic programming to problems where all demand points are located on a line. Brimberg and Love (1998) later generalized this approach to certain class of planar problems. More recently, the application of global optimization techniques has increased the size of problem instances that can be solved exactly. Examples are a D.C. programming method for the two facility case (Chen et al, 1998) and a column generation approach (Kraus, 1997). Approximation schemes for the problem were developed by Lin and Vitter (1992a and 1992b) and by Arora, Raghavan and Rao (1998), who gave a ϵ -approximation scheme for the Euclidean location allocation problem. Bischoff et al (2006) present a mixed integer programming approach for multifacility location allocation problem with polyhedral barriers. Fazel-Zarendi and Beck (2009) focus on the Location-Allocation Problem with Logic-Based Benders' Decomposition. Kuenne and Soland (1972) present branch-and-bound algorithms for location allocation problem.

2.2.2 Data analysis

Data analysis techniques perform inspecting, cleaning, transforming, and modeling data with the goal of highlighting useful information, suggesting conclusions, and supporting decision making; Data analysis has multi-facets and approaches, encompassing diverse techniques under a variety of names, in different economics, science, and mathematical science domains. Examples of data analysis techniques are cluster analysis, correspondence analysis, regression analysis etc. In location & allocation problems, data

analysis could be used in allocating customers to logistics facilities using distance based clustering.

Hsieh and Tien (2004) use Self-organizing feature maps for solving location–allocation problems with rectilinear distances. Lozano et al (1998) apply Kohonen maps for solving a class of location-allocation problems. Barreto et al (2007) use clustering analysis in a capacitated location-routing problem. Satani et al (1998) developed a commercial facility location model using multiple regression analysis. Tsuchiya et al (1996) present a neural network approach to facility layout problems.

2.2.3 Simulation

Simulation modeling is an experimental and applied methodology used for describing the behavior, constructing theories or hypotheses, and applying these theories to predict future behavior of systems (Shannon 1975, Banks 1998). It is the use of mathematical models to imitate a situation many times in order to estimate the likelihood of various possible outcomes. Simulation has been applied in many fields like science, engineering, business and social management (Poole and Szymankiewicz, 1977). Barton (1970) presents four categories of simulation modeling: analysis, man-model simulation, man-computer simulation, and all-computer simulation.

Armour and Buffa (1965) present a heuristic algorithm and simulation approach for relative location of facilities. Canbolat and van Massow (2011) present a spreadsheet

based simulation model for locating emergency facilities with random demand for risk minimization. Greasley (2008) applied simulation for facility design. Vos and Akkermans (1996) proposes dynamics of facility allocation using system dynamics simulation models that usually comprise a large number of interrelated variables.

2.2.4 Muticriteria decision analysis

Multicriteria decision involves evaluation of a set of alternatives using a pre-defined set of criteria by a committee of decision makers or experts. Examples of MCDA techniques are AHP, ANP, TOPSIS, SAW etc. In location allocation problem, the criteria can be minimum cost, distance and travel time etc. and the alternatives are the potential locations to be evaluated for final site selection. The location problem was first posed by Weber and formed the theory of the Location of Industries (1929). Freek (1999) and Ho et al (2008) investigated location & allocation problem using multi-factors decision analysis.

Farahani and Helmatfar (2009) provide a review on recent efforts and development in multi-factors location allocation problems in three categories including bi-objective, multi-objective and multi-attribute problems and their solution methods. Fortenberry and Mitra (1986) present a multiple criteria approach to the location-allocation problem. Badri (1999) combined the analytic hierarchy process and goal programming for global facility location-allocation problem. Ho et al (2008) used Analytic Hierarchy Process (AHP) to

optimize the facility location-allocation problem in the contemporary customer-driven supply chain.

2.2.5 Heuristics

Heuristics methods yield good solutions at reasonable cost and can be used for providing good initial solutions in other optimizing methods (Anand and Knott, 1986). A well known heuristics approach for the sequential location allocation is by Cooper (1964). The method alternates between a location and an allocation phase until no further improvements can be made. Brimberg et al (1998) propose heuristics based decomposition strategies for large-scale continuous location-allocation problems. A p-Median plus Weber heuristic was proposed by Hansen et al. (1998). Local search methods were developed by Love and Juel (1982), and Brimberg and Mladenovic (1996a). The modification of the objective function was investigated in the location allocation problem by Chen (1983). Gamal and Salhi (2001) present constructive heuristics for the uncapacitated continuous location–allocation (UCLA) problem.

Doerner et al (2009) present a method of multi-factors location planning for public facilities. For the optimal solution of the multi-objectives optimization problem, they propose a heuristic approach based on the NSGA-II algorithm, which is a kind of GA.

2.2.6 Metaheuristics

A metaheuristic is an approach used for optimization by iteration in the neighborhood of solution space. Examples of metaheuristics are simulated annealing, tabu search, genetic algorithms etc. Metaheuristics have been applied in many different areas such as science, engineering, logistics, management, and defense (Glover and Kochenberger, 2002).

Zhou et al. (2002) use genetic algorithm approach for balanced allocation of customers to logistics depots. Zhou et al. (2003) present a genetic algorithm approach to bi-criteria allocation of customers to warehouses. Villegas et al (2006) use genetic algorithm approach for allocation of logistics depots to customers. Cortinhal and Captivo (2003) applied Genetic Algorithms for the Single Source Capacitated Location Problem (SSCLA) and propose three algorithms based on the Nondominated Sorting Genetic Algorithm; the Pareto Archive Evolution Strategy; and mathematical programming. The problem is modeled as a biobjective (cost, coverage) uncapacitated problem under allocation constraints of customers for coffee supply network.

Murray and Church (1996) apply simulated annealing for location allocation problem. Vecihi et al. (2006) present the evolutionary simulated annealing (ESA) for large-scale uncapacitated facility location problem. Land allocation zones for forest management were created using an annealing approach by Mark et al. (2004). Their multiobjective function comprised of landscape-level targets, size, shape, and all ecosystem types.

Tabu search for location allocation problems was investigated by Brimberg and Mladenovic (1996) and Ohlemüller (1997). Crainic et al. (1993) apply tabu algorithm for multi-commodity location & allocation with balancing requirements. Kulturel-Konak et al. (2003) efficiently solve the redundancy allocation problem using tabu algorithm. Junjiro et al (2006) tested tabu search for efficient allocation of SVRs optimizing the rate of operation for distribution systems. Cordeau and Laporte (2005) apply tabu search and models heuristics for the berth-allocation.

Chan and Kumar (2009) apply multi ant colony optimization approach for customers allocation. Hua et al (2010) develop ant colony optimization algorithm for computing resource allocation based on cloud computing environment. Kwang and Weng (2002) apply multiple ant colony optimizations (MACO) for load balancing. Silva et al. (2008) apply ant colonies for distributed optimization of a logistic system and its suppliers. Comparison of genetic algorithms, random restart and two-opt switching for solving large location–allocation problems is presented by Houck et al (1996).

2.2.7 Hybrid Approaches or Combinations of the above

Some hybrid algorithms have been also suggested, such as the one based on simulated annealing and random descent method (Ernst and Krishnamoorthy 1999) and the one utilizing the Lagrange relaxation method and genetic algorithm (Gong et al. 1997). Brimberg et al. (2000) improved present algorithms and proposed variable neighborhood search, which is proved to obtain the best results when the number of facilities to locate is

large. Abdinnour-Helm (1998) developed a hybrid heuristic based on Genetic Algorithms (GAs) and Tabu Search (TS) for the uncapacitated hub location problem. Chen (2007) proposes hybrid heuristics based on simulated annealing, tabu list, and improvement procedures for the uncapacitated single allocation hub location problem. Silva and Cunha (2009) propose multi-start tabu search heuristic for the uncapacitated single allocation hub location problem. A Tabu search and ant colony system approach for the capacitated location-routing problem was proposed by Bouhafs et al.(2008). Qin (2006) put forward an ant colony arithmetic model for logistics distribution centre allocation problem. Kansou and Yassine (2010) use a hybrid approach consisting of ant colony optimization and a heuristic saving method for the multi-depots capacitated arc routing problem.

Chapter 3:

Problem definition

Distribution of goods to customers from logistics depots produces the problem of how to construct the network of logistics depots and customers, and to optimize the delivery process. That is, how to cluster customers and service them through logistics depots considering least distance, cost, time etc. The distinction between the location allocation problem treated in this thesis and the traditional location allocation problem lies in its multiobjective and dynamic nature. The multiple objectives considered are travel time, travel distance, travel cost etc. and developed based on practical constraints such as presence of congestion, timing and access restrictions imposed by municipal administrations in urban areas etc. The dynamic aspect means the location allocation results are not fixed forever but vary with change in municipal access or timing regulations, congestion, or land, material and labor costs on logistics networks.

In our problem, each customer should be serviced by a logistics center. It is possible that a logistics depot gets no customer allocations, in that case it will be closed down. Multiple criteria (factors) such as facility opening costs, travel cost, travel distance, and travel time to customers are considered in deciding the opening of logistics depots and customer allocations. The solution for location allocation problem should therefore be

developed considering these factors, customer demands and capacity constraints of logistics depots. Figure 3.1 shows a logistics network comprising of logistics facilities (depots) and the customers.

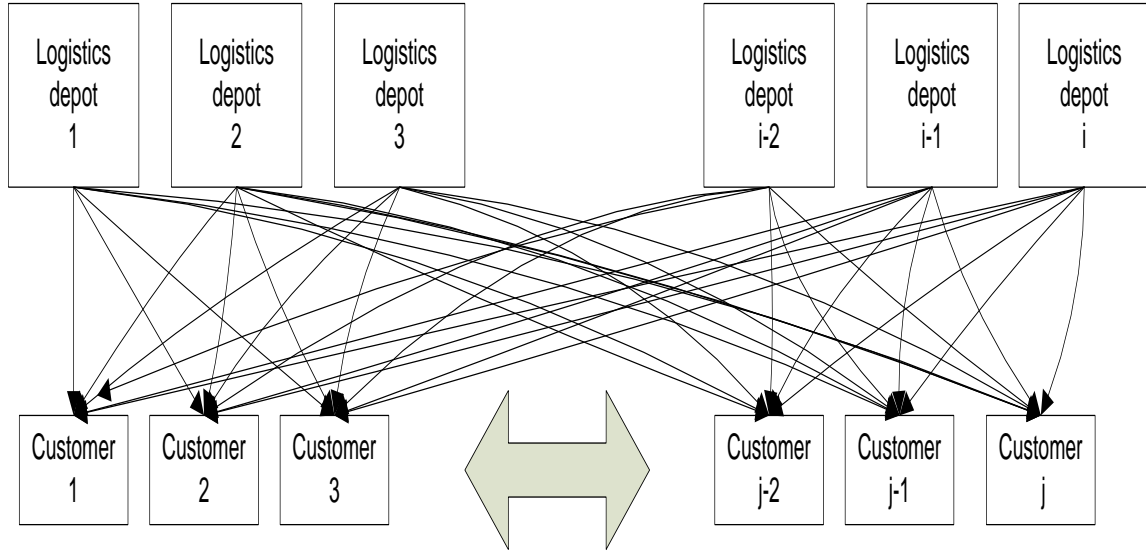


Figure3.1 Network of logistics depots and customers

Let us denote the logistics centers by i , ($i=1,2,...,m$) and customers by j ($j=1,2,...,n$). The maximum number of depots is denoted by m and the maximum number of customers is denoted by n . The cost of opening a facility i is denoted by c_i and its capacity by b_i . The demand for customer j is given by d_j . The distance between depot i and customer j is given by d_{ij} , travel cost by c_{ij} , and travel time by t_{ij} . The binary variable y_i is 1 if facility i is opened, otherwise it is set equal to 0. Similarly, binary variable x_{ij} is equal to 1 if customer j is allocated to depot i and is set equal to 0 in the contrary case. The quantity of goods transported between i and j (if they are connected) is given by q_{ij} . The goal is to

minimize the total costs, that is, opening costs of facilities and delivery costs of goods to customers from logistics depots. The delivery cost for customers is a weighted function of travel distance (d_{ij}), travel cost (c_{ij}) and travel time (t_{ij}) where the weights of travel distance, travel cost and travel time are represented by w_1 , w_2 and w_3 respectively. Since, the facility opening costs, travel distance, travel time, travel costs etc. are in different units, they are normalized before being used in the objective function. Let us denote the normalized values of c_i , d_{ij} , c_{ij} and t_{ij} by $c'_i, d'_{ij}, t'_{ij}, c'_{ij}$ which are computed as follows:

$$\begin{aligned} c'_i &= c_i / \sum c_i \\ d'_{ij} &= d_{ij} / \sum d_{ij} \\ t'_{ij} &= t_{ij} / \sum t_{ij} \\ c'_{ij} &= c_{ij} / \sum c_{ij} \end{aligned}$$

Using the normalized values $c'_i, d'_{ij}, t'_{ij}, c'_{ij}$, the mathematical formulation of the problem is presented as follows:

Objective:

$$\text{Minimize} \quad \sum_{j=1}^n y_i * c'_i + \sum_{i=1}^m \sum_{j=1}^n x_{ij} * (w_1 * d'_{ij} + w_2 * t'_{ij} + (1 - w_1 - w_2) * c'_{ij}) \quad (3.1)$$

s.t

$$\sum_{i=1}^m x_{ij} = 1, \forall j \in 1, 2, \dots, n \quad (3.2)$$

$$\sum_{i=1}^m q_{ij} x_{ij} = d_j, \forall j \in 1, 2, \dots, n \quad (3.3)$$

$$\sum_{j=1}^n q_{ij} x_{ij} \leq b_i y_i, \forall i \in 1, 2, \dots, m \quad (3.4)$$

$$x_{ij} \in \{0, 1\}$$

$$y_i \in \{0, 1\}$$

$$q_{ij} \geq 0$$

It can be seen from (3.1) that the objective function comprises of multiple factors such as facility opening costs (c_i), travel distance (d_{ij}), travel cost (c_{ij}) and travel time (t_{ij}). If $w_1 = 1$ and $w_2 = 0$, then $w_3 = 0$ and the above objective function reduces to a single factor optimization problem based on minimizing the travel distance only. The objective function (3.1) now reduces to (3.5) which is given as follows:

$$\text{Minimize} \quad \sum_{j=1}^n y_i * c_i + \sum_{i=1}^m \sum_{j=1}^n x_{ij} * d_{ij} \quad (3.5)$$

The constraints in single factor optimization model remain the same as in multifactor optimization model. Equation (3.2) ensures that each client is served by exactly one facility. Equation (3.3) shows the demand satisfaction constraint of the customers. Equation (3.4) shows the capacity restriction constraints for the logistics depots. The facility location selection variable x_{ij} and the customer allocation variable to logistics facilities y_i are binary. The quantity allocations q_{ij} are non-negative real numbers.

Chapter 4:

Solution Approach

The location & allocation problem treated in this thesis is multi-objective in nature. According to Konak et al (2006) and Sawaragi et al (1985), there are two general approaches to solve multiple-objective optimization problems.

1. Combining individual objective functions into a single composite function or move all but one objective to the constraint set. In the former case, determination of a single objective is possible with methods such as utility theory, weighted sum etc., but the problem lies in the proper selection of the weights which is tricky process as small perturbations in the weights can sometimes lead to quite different solutions. In the latter case, the problem is to move objectives to the constraint set where a constraining value must be established which can again be rather arbitrary. In both cases, an optimization method would return a single solution rather than a set of solutions that can be examined for trade-offs.
2. The second approach consists of determining Pareto optimal solutions where a Pareto optimal set is defined as a set of solutions that are non-dominated with respect to each other. Each Pareto solution dominates other in terms of one objective function value and there is always a certain amount of sacrifice in this

objective value when trying to achieve a certain amount of gain in the other objective(s). Pareto optimal solution sets are often preferred to single solutions since the final solution of the decision-maker is always a trade-off in practice. The Pareto optimal sets can be of varied sizes, usually it increases with the increase in the number of objectives.

In our thesis, since all the functions are minimization type and the weights of the objective functions can be obtained using multicriteria decision making approaches such as AHP, we have used the weighted sum method (Marler and Arora, 2009) for treating the multiobjective problem is used over the Pareto optimal solution. Before applying the weighted sum method, we normalize all the factors used in the model to bring them to a common unit to avoid discrepancies of scale. If s_{ij} represents an element of matrix $S_{m \times n}$ where $i=1,2,\dots,m$ and $j=1,\dots,n$, then the normalized values a_{ij} can be obtained using any of the following four methods:

$$a_{ij}=s_{ij}/\sum(s_{ij}) \quad (4.1)$$

$$a_{ij} = s_{ij} / \max(s_{ij}) \quad (4.2)$$

$$a_{ij} = (s_{ij} - \min s_{ij}) / (\max(s_{ij}) - \min s_{ij}) \quad (4.3)$$

$$a_{ij} = s_{ij} / \sqrt{\sum(s_{ij})^2} \quad (4.4)$$

The normalization method we have chosen for our multiobjective location allocation model is given by eqn (4.1). Four types of metaheuristics based solution approaches are proposed for solving the multiobjective capacitated location allocation problem on

logistics networks. The metaheuristics were developed in Matlab. Jones et al (2002) present a detailed overview of multi-objective meta-heuristics. The details of the metaheuristics proposed in the thesis are presented as follows.

4.1 Genetic Algorithms (GA) for location allocation

Genetic algorithm is a kind of stochastic search and optimization technique based on principles from evolution theory (Holland, 1975). Genetic algorithms form part of the larger class of ‘Evolutionary algorithms’ which generate offsprings for better solution by using techniques inspired from genetic evolution such as crossover, inheritance, selection, mutation etc. Goldberg (1989) defines genetic algorithm as a search heuristic that mimics the process of natural evolution. This heuristic is routinely used to generate useful solutions, search and optimize better solution from neighborhood of solution space. Genetic selection for crossover and mutation is important and should be carefully done because it affects the computational speed and quality of final results of the genetic algorithms.

The application of GA for location allocation problem has been investigated by several researchers. Zhou et al. (2002) use genetic algorithm for balanced allocation of customers to logistics depots. Zhou et al. (2003) present a genetic algorithm approach to bi-criteria allocation of customers to warehouses. Villegas et al. (2006) use genetic algorithm approach for allocation of logistics depots to customers. Cortinhal and Captivo (2003) applied Genetic Algorithms for the Single Source Capacitated Location Problem.

4.1.1 Method Description

The various steps of the genetic algorithm for location allocation problem are presented as follows.

Representation Scheme

The representation scheme for the chromosome is a n -bit string where n represents the number of customers. A non-zero value for the i^{th} bit implies that a depot is allocated to that customer. If a depot is not present in the string, it implies that this depot was not opened or closed for non-feasibility reasons (allocation of zero customers). Let us consider a network comprising of 21 customers and 7 logistics depots. The representation of an individual chromosome (solution) is illustrated as follows:

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
-----------	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----

Figure 4.1: Solution Representation for Location Allocation problem

Using case (a), we can say that logistics depot 1 is allocated to customers (1,2,3), logistics depot 2 to customers (4,5,6), etc. On analyzing results for case (b), we see that logistics depot 1 is allocated to customers (1,7,20), logistics depot 3 to (4,6,15,16,17,21). However, in case (b) the logistics depot 7 is absent which means it was not opened for the reasons of zero allocation of customers.

Case (a)

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7

Case (b)

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots	1	6	2	3	5	3	1	4	2	2	2	2	6	4	3	3	3	4	2	1	3

Fitness Function

The fitness function is same as eqn (3.5) for single factor and eqn (3.1) for multifactor subject to constraints (3.2-3.4) (Chapter 3).

Parents Selection Procedure

Few methods of parent selection as described in Beasley and Chu (1996) and Talbi (2009) are ranking (picking the best individuals every time), Roulette wheel or proportionate (probability of selection is proportional to fitness), Tournament (initial large number are selected via roulette wheel, then the best ranked are chosen), Stochastic (various methods of replenishment of less fit stock (useful) or initial selection (not useful)) and Elite (in combination with other selection schemes, always keep the fittest individual around). To select the parents for crossover, we have chosen the ranking method.

Crossover Operator

Different cross-over operators are one-point, two-point, uniform, arithmetic, heuristic etc.

We chose the one-point cross-over in our approach which involves randomly generating

one cross-over point and then swapping segments of the two parent chromosomes to generate two child chromosomes. Let P1 and P2 be the parent strings $P1[1], \dots, P1[n]$ and $P2[1], P2[2], \dots, P2[n]$ respectively. Choose a cross-over point k , where $0 \leq k \leq 1$. Then the child chromosomes C1 and C2 are given by:

$$C1 = P1[1], \dots, P1[k], P2[k+1], \dots, P2[n]$$

$$C2 = P2[1], \dots, P2[k], P1[k+1], \dots, P1[n]$$

Mutation Operator

Mutation is applied to each child after crossover. It works by inverting each bit in the solution with some small probability. In our thesis, the mutation operator works by selecting randomly one of the customers in the child chromosome and allocating to another logistics facility picked at random.

Replacement population method

The newly generated child solutions are put back into the original population to replace the “less fit” members. The average fitness of the population increases as child solutions with better fitnesses replace the less fit solutions (“incremental replacement”). Note that when replacing a solution, care must be taken to avoid duplicate solutions from entering the population as it will severely limit the GAs ability to generate new solutions. Another commonly used method is the “generational replacement”, which generates a new population of children and replaces the whole parent population (Beasley et al., 1993). In

fact, the simple or generational GAs replace entire population per the dictates of the selection scheme whereas the steady state or online GAs use different replacement schemes such as Replace worst, Replace best, Replace parent, Replace random and Replace most similar (crowding).

In our GA, we have used the incremental replacement method.

Population size

The performance of GA is influenced by the population size. Small populations run the risk of seriously under-covering the solution space, while large populations are computationally intensive [Jaramillo et al, 2002]. Alander [1992] suggests that a value between n and $2n$ is optimal for the problem type considered, where n is the length of a chromosome. In our case, we chose a population size equal to n which is equal to the number of customers in the LA problem.

4.1.2 High level pseudocode for GA

The high level pseudocode for implementing GA is presented as follows:

1. Set iteration counter $t = 0$.
2. Generate the initial population, $P(t)$, randomly.
3. Evaluate fitness of the population $P(t)$ using the objective function.
4. While (number of iterations $t \leq$ Maximum value) or (improvement in objective function value $\leq 10^{-5}$) do

4.1. Set $t = t + 1$

4.2. Select two solutions P1 and P2 from the population using the ranking method.

4.3. Apply genetic operators to P1 and P2

4.3.1. If crossover, then combine P1 and P2 using single point crossover to generate offspring O1.

4.3.2. If O1 is identical to any of its parents, then apply mutation operator to the parent with the best fitness.

4.3.2.1. If mutation, then apply mutation operator to the parent with the best fitness to form a offspring O1.

4.3.3. Evaluate the fitness of the new child set using the objective function

4.3.4. If fitness of chromosome is improved or objective value is reduced (in case of minimization) then utilize the incremental replacement method to create $P(t)$ and update population size.

5. Stop. Print final results.

4.1.3 Example

Let us consider 2 logistics depots D1 and D2 and 6 customers C1, C2, C3, C4, C5, C6 respectively. The initial population consists of four chromosomes P1, P2, P3, P4 generated at random (Table 4.1). Please note that each of these chromosomes (solutions) satisfies the demand and capacity constraints.

Depot	Solution String						Objective Function
P1	1	1	1	2	2	2	30
P2	2	2	2	1	1	1	45
P3	1	1	2	2	1	2	27
P4	1	1	2	2	2	1	80

Table 4.1: Initial population for genetic algorithm

Let us select P1 and P3 for cross-over since they have the least objective function value (ranking method). One point cross-over is used to generate offspring (s) O1 (1 1 1 2 1 2) and O2 (1 1 2 2 2 2) both of which have objective function equal to 25 (lower than the parent chromosomes). Therefore, the new offsprings O1 and O2 are returned back to the original population to replace P1 and P3. The crossover probability ranges from 0.4-0.7 and the mutation probability is near 0.1. This process of crossover, mutation, generation of offsprings and renewal of parent population continues until the new population size is same as the initial population size. Then, the whole procedure of evaluating population fitness, chromosome generation, population replenishment etc. continues until number of iterations \leq Maximum value or improvement in objective function value $\leq 10^{-5}$. At this point the results are said to be stabilized over time or the algorithm converges and the final results are generated.

4.1.4 Advantages of Genetic Algorithms

The advantages of Genetic algorithms are that it supports multi-objective optimization, can be applied to new problems with exploratory type of solutions, always improves

solutions over time, and can be easily parallelized or distributed.

4.1.5 Limitations of Genetic Algorithms

Genetic algorithms require careful selection of chromosomes, cross-over and mutation operators to generate better results over time. If they are not carefully planned, there is risk of getting trapped into local optima and the algorithm may involve high computational times for generating final results.

4.2 Simulated annealing for location allocation

Simulated annealing is a generalization of the Monte Carlo method for examining the equations of state and frozen states of n -body systems (Metropolis et al., 1953). The concept is based on the manner in which liquids freeze or metals recrystallize in the process of annealing. In an annealing process a melt, initially at high temperature and disordered, is slowly cooled so that the system at any time is approximately in thermodynamic equilibrium. If the initial temperature of the system is too low or cooling is done insufficiently slowly the system may become quenched forming defects or freezing out in metastable states. Therefore, the process of optimization should have an appropriate speed, or it couldn't get the satisfied solution by simulated annealing.

Application of simulated annealing for LA problem has been studied by several researchers. Murray and Church (1996) apply simulated annealing for location allocation problem. Vecihi et al. (2006) present the evolutionary simulated annealing (ESA) for

large-scale uncapacitated facility location problem.

4.2.1 Method description

The various steps of simulated annealing algorithm for location allocation problem are presented as follows:

Generation of initial solution

The initial solution is generated by opening all facilities and performing random allocation of clients to them.

Initialization of annealing parameters

The initialization process involves the following parameters:

- An initial temperature
- A temperature function to determine how the temperature will be changed as the algorithm will proceed.
- The number of iterations to be performed at each temperature
- A termination condition to stop the algorithm such as maximum number of iterations of difference between the old and new objective function values.

Updation of temperature

There is always a compromise between the quality of the obtained solutions and the speed of the cooling schedule. If the temperature is decreased slowly, better solutions are obtained but with more computation time. Different ways for updating temperature T are

as presented as follows (Talbi, 2009):

Linear: $T_i = T_0 - i \times \beta$,

Geometric: $T_i = \alpha \times T_{i-1}$,

Logarithmic: $T_i = \frac{T_0}{\log(i)}$,

Very slow decrease: $T_{i+1} = \frac{T_i}{1 + \beta T_i}$

In the above formulas, α and β are constants, T_0 represents the initial temperature, and T_i represents the temperature at iteration i .

We have used the linear function $T_i = T_0 - i \times \beta$ for temperature update in our SA for location allocation problem.

Generation of neighborhood solutions

The neighborhood of a solution is generated by some suitable mechanism such as moving customers from one logistics depot to another in our location allocation problem and recording the change in the objective function value.

Accept solutions

If a reduction in the objective function occurs, the current solution is replaced by the neighborhood solution, otherwise, the neighborhood solution is accepted with a certain probability. The probability of accepting an uphill move is normally set to $\exp(-\Delta E/T)$ where T is a control parameter which corresponds to temperature in the analogy with physical annealing, and ΔE is the change in the objective function value. The SA starts

with a relatively high temperature, attempts a certain number of moves at each temperature, and then drops the temperature parameter gradually until a minimal temperature T_{min} has been reached (Al-khedhairi, 2008).

4.2.2 High level pseudocode for SA

The high level pseudocode for simulated annealing is presented as follows:

1. Set initial solution $s = s_0$, initial temperature $T = T_{max}$, maximum number of iterations $= L$, iteration counter $n = 0$, temperature change counter $t = 0$.
2. Initialize temperature decreasing rate R and minimal acceptable temperature T_{min} .
3. While ($T \geq T_{min}$)
 - 3.1. While (number of iterations $n \leq \text{Maximum value } L$) or (improvement in objective function value $\Delta E \leq 10^{-5}$) do
 - 3.1.1: Generate a neighboring solution s' .
 - 3.1.2: Calculate $\Delta E = f(s') - f(s)$.
 - 3.1.3 If $\Delta E \leq 0$ then
 - 3.1.3.1 Set $s = s'$
 - else
 - 3.1.3.2 Select a random number R from $U(0,1)$. If $R < e^{-\frac{\Delta E}{T}}$ accept $s = s'$ else update s with next best neighboring solution s'' with $\Delta E \leq 0$.
 - 3.1.4 Set $n = n+1$;
 - 3.2: Set $t = t+1$ and $T = R \cdot (T_{max}/L)$;

4. Stop. Print final results.

4.2.3 Example

Let us consider 2 logistics depots D1 and D2 and 6 customers C1, C2, C3, C4, C5, C6. We generate an initial solution say S1 at random with overall objective function value equal to 35 (Table 4.2). Set the initialization parameters $\beta = 0.1^{(-10)}$, $\alpha = 0.997$, and $T_0 = 2$. Let us generate a neighboring solution S2 with objective function 25. Since $\Delta E = 25 - 35 \leq 0$, the solution is accepted. Then, we repeat the process to generate another solution S3 in neighborhood of S2. Since the objective function value of S3 (=28) is greater than that of S2, we generate a random number R from $U(0,1)$ and check if $R < e^{-\frac{\Delta E}{T}}$. Since this condition holds to be true, solution S3 is accepted. This process continues for a pre-defined number of iterations N or until very small change in magnitude of $\Delta E (= 10^{-5})$ is observed. The best solution is recorded at this stage and the temperature T is lowered by a fixed amount $= R - (R * t/L)$ and the whole process is repeated again until a pre-defined minimum temperature T_{min} has been reached. Of course, the solution considered at any stage of the algorithm must satisfy the demand and capacity constraints of location allocation problem.

Depot	Objective Function	Solution String					
S1	30	1	1	1	2	2	2
S2	25	1	1	2	1	2	2
S3	28	1	2	1	1	2	2

Table 4.2: Solutions for Simulated Annealing

4.2.4 Advantages of Simulated Annealing

Simulated Annealing is a simple, effective, and flexible approach, which could be easily understood and applied in many fields without relative inner structure. It can deal with arbitrary systems and cost functions and statistically guarantees finding an optimal solution. It is relatively easy to code even for complex problems and generally gives a “good” solution.

4.2.5 Limitations of Simulated Annealing

When the speed of decreasing temperature is too fast, the algorithm possibly can't get the optimized result or a satisfied solution; another disadvantage is the use of large CPU time in generating solutions, and lastly, there is the lack of memory which does not prevent the procedure from repeating a solution evaluated previously.

4.3 Tabu Search for location allocation

Glover (1989) proposed the tabu search or tabu algorithm for optimizing problems by tracking and guiding. It begins by setting up a set of feasible solutions, choosing certain solutions in the feasible neighborhood subject to constraints of tabu list for searching the objective solution, and finally generating the solution. Tabu search enhances the performance of a local search method by using memory structures: once a potential solution has been determined, it is marked as "taboo" (tabu) so that the algorithm does not visit that possibility repeatedly. TS focuses on how to cut off large computation in the

solution space so as to avoid long computation times and make the search quicker. The tabu list length is an important factor in TS for the reason that its length will affect the computation speed or the efficiency of the searching process and therefore be decided by the condition of problem or other factors that affect the TS process.

Tabu search for location allocation problems was investigated by Brimberg and Mladenovic (1996), and Ohlemüller (1997). Crainic et al. (1993) apply tabu algorithm for multi-commodity location & allocation with balancing requirements.

4.3.1 Method Description

Generate Initial Solution

This step involves generating initial solution (configuration) which comprises of opening all facilities, random allocation of clients, and evaluation of objective function for that solution.

Initialize memory structures

This step involves initialization of all memory structures used during the run of the tabu search algorithm. The memory structures involved are tabu list, medium-term and long-term memories. The difference between short term and long term memory is that the short-term memory restricts the neighborhood $N(s)$ of solution s to a subset $N'(s) \subseteq N(s)$ whereas the long-term memory may extend $N(s)$ through the inclusion of additional solutions (Glover and Laguna, 1997).

Generate admissible solutions

Generate a set of candidate moves from the current configuration. A move describes the process of generating a feasible solution to the problem. For example, Add, Drop, Swap etc. In our case, all these three kind of moves are involved in allocating customers to logistics facilities to generate admissible solutions that satisfy capacity and demand constraints.

Select best solution

This step returns the best admissible move (solution) from the list of candidate moves. If the best of these moves is not tabu or if the best is tabu but satisfies the aspiration criteria, the pick that move and consider it to be the new current configuration, else pick the best move that is not tabu and consider it to be the new current configuration. Repeat the procedure for a certain number of iterations. On termination, the best solution obtained so far is the solution obtained by the algorithm.

The tabu status of solution approaches is maintained for tl number of iterations, the parameter tl being called the tabu tenure or tabu list length. Normally, $tl=(n*(n-1))^{0.5}$. Unfortunately, setting tl in advance may forbid moves towards attractive, unvisited solutions. To avoid such an undesirable situation, an aspiration criteria is used to override the tabu status of certain moves, that is, if a certain move is forbidden by tabu restriction, then the aspiration criteria, when satisfied, can make this move allowable.

Update memory structures

To increase the efficiency of simple TS, long-term memory strategies can be used to intensify or diversify the search. Intensification strategies are intended to explore more carefully promising regions of the search space either by recovering elite solutions (i.e., the best solutions obtained so far) or attributes of these solutions. Diversification refers to the exploration of new search space regions through the introduction of new attribute combinations (Glover and Laguna 1997, Dorigo and Stutzle, 2004).

Parameter setting

Following parameters need to be set before running the TS:

- The number of random solutions to be generated from the current one.
- The tabu list size.
- The probability threshold, whose value affects the probability assigned to every facility to change its status.
- Maximum number of non-improving iterations before termination.

4.3.2 High level pseudocode for TS

The high level pseudocode for the proposed tabu search is presented as follows:

1. Generate initial solution s_0 .
2. Initialize the tabu list, medium-term and long-term memories
3. Set $s_{best} = s_0$.

4. While (number of iterations \leq Maximum value) or (improvement in objective function value $\leq 10^{-5}$) do

4.1. Generate admissible solutions (s)

4.2. Select best solution s' from the list of admissible solutions (s)

4.3. Update tabu list, aspiration conditions, medium and long term memories;

4.3.1. If intensification criterion holds, then intensification;

4.3.2. If diversification criterion holds, then diversification;

4.4. If ($f(s') < f(s_{best})$ and (s' is non-tabu)) or ($f(s') < f(s_{best})$ and (s' is tabu and aspiration criteria holds)) then

Set $s_{best} = s'$.

Pick the best move s'' that is non-tabu and set $s_{best} = s''$.

5. Stop. Print final results.

4.3.3 Example

Let us consider 2 logistics depots D1 and D2 and 6 customers C1, C2, C3, C4, C5, C6.

An initial solution is generated at random say $S_0 = \{C1(D1), C2(D1), C3(D1), C4(D2), C5(D2), C6(D2)\}$ whose overall objective function value is 35. Let us generate a neighboring solution $S_1 = \{C1(D1), C2(D1), C6(D1), C4(D2), C5(D2), C3(D2)\}$ with objective 29. Since, the new solution is better than the previous one and is not present in the tabu list = $\{C1(D1), C3(D1), C5(D1), C2(D2), C4(D2), C6(D2)\}$, the new solution is accepted and updated as the best solution. The solution S_1 is also added to the tabu list to

avoid repetitive solutions from entering into the tabu list. Next, we generate solutions in the neighborhood of S1 and repeat the whole process again updating best solution each iteration. This process continues until maximum number of iterations have been reached or very minimal improvement in objective function value (say $\leq 10^{-5}$) is observed.

4.3.4 Advantages of Tabu Search

The advantage of Tabu Search is that it searches over all the solutions space to find the optimized solution. Due to the presence of Tabu list, only limited solutions in neighborhood are searched which saves lot of computation time and also avoids low quality solutions.

4.3.5 Limitations of Tabu Search

Since Tabu Search repeatedly checks solutions for presence in the Tabu List, it wastes much of time as well, or TS process will slow the speed of computation if the computing unit entering tabu list requires sorting. Therefore, how to set up the tabu space is very important in TS.

4.4 Ant Colony Optimization (ACO) for location allocation

Marco Dorigo (1992) developed the ant colony approach. The ant colony optimization (ACO) is a probabilistic technique for solving computational problems which can be reduced to finding good paths through neighborhoods. Ant colony optimization originally is a biological swarm intelligent or an evolutionary approach where ants find their food

using the shortest route by cooperation. ACO are just like other population-based metaheuristics that could be used to find approximate solutions to difficult optimization problems.

Alaya et al (2007) apply ant colony optimization for multi-objective optimization problems. Chan and Kumar (2009) apply multi ant colony optimization approach for customer allocation. Silva et al. (2008) apply ant colonies for distributed optimization of a logistic system and its suppliers.

4.4.1 Method description

The various steps of ACO are described as follows:

Initialization of ACO parameters and pheromone trails

The first step involves setting the initial values of ACO parameters, such as α , β , q_0 , etc. where α and β are parameters used for controlling the relative weight of the pheromone trail and the heuristic value (Dorigo and Stutzle, 2004), $q_0 \in [0,1]$ is a tunable parameter for determining the relative importance between exploitation and exploration. We also compute the value of the initial pheromone trail τ_0 , and construct the tabu lists of all ants, which contain all the unvisited nodes for each ant and the list of optimum paths traversed by the ant colonies. The initial pheromone intensity τ_{ij} or the path from nodes i to j is set equal to τ_0 , that is $\tau_{ij} = \tau_0$ and $\Delta\tau_{ij} = 0$.

Solution (Tour) construction

In the second step, ant k currently at node i determines the node to visit next, node j , by applying the state transition rule

$$j = \begin{cases} \arg \max_{u \in S_k(i)} [\eta_{iu}]^\alpha [\tau_{iu}]^\beta & \text{if } q \leq q_0; \\ J, & \text{if } q > q_0 \end{cases} \quad (4-1)$$

where η_{iu} is a heuristic value which equals to the inverse of the length d_{iu} from node i to node u , τ_{iu} is the amount of pheromone trail of the path from node i to node u , α and β are two parameters used for controlling the relative weight of the pheromone trail and the heuristic value (Dorigo and Stutzle, 2004), $S_k(i)$ is a tabu list containing those unvisited nodes for ant k currently at node i , q is a random number uniformly distributed in $[0,1]$, $q_0 \in [0,1]$ is a tunable parameter for determining the relative importance between exploitation and exploration, and J is the node randomly chosen from the list $S_k(i)$ according to the pseudo random proportional distribution rule

$$p_{ij}^k = \begin{cases} \frac{[\eta_{iu}]^\alpha [\tau_{iu}]^\beta}{\sum_{u \in S_k(i)} [\eta_{iu}]^\alpha [\tau_{iu}]^\beta} & \text{if } j \in S_k(i); \\ 0, & \text{if } j \notin S_k(i); \end{cases} \quad (4-2)$$

where p_{ij}^k is the probability that ant k chooses to move from node i to node j .

Update of pheromone trails

In this step, local and global update of pheromone trails is performed. The local update of the pheromone trail on each edge is performed by applying the rule

$$\tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \rho \cdot \tau_0 \quad (4-3)$$

where $\rho \in (0,1)$ is the pheromone evaporating rate used in the local update and τ_0 is the

initial value of the pheromone trail. In step 1 of the ACO algorithm, $\tau_0 = 1/d_{\text{total}}$ where d_{total} is the distance between the customer and the logistics depot, which is derived from a randomly generated tour of an ant, using the least distance objective function, and subject to satisfaction of customer demand and capacity constraints of logistics depots. With this updating rule, the chance for ants to stick to a few previously visited paths can be reduced.

After all ants have completed the tours, the global update of pheromone trails is performed. The purpose of the global update is to increase the pheromone trails of the paths on the tour with the best performance so far. The global update is performed by applying the rule

$$\tau_{ij} = (1 - \rho_g) \cdot \tau_{ij} + \rho_g \cdot \Delta \tau_{ij}^{bs} \quad (4-4)$$

where $\rho_g \in (0,1)$ is the pheromone evaporating rate for global updating and $\Delta \tau_{ij}^{bs}$, the amount of pheromone for the best tour found so far, is given by the equation

$$\Delta \tau_{ij}^{bs} = \begin{cases} \frac{1}{C^{bs}}, & \text{if the path from nodes } i \text{ to } j \text{ belongs to } T^{bs}; \\ 0, & \text{otherwise} \end{cases} \quad (4-5)$$

In the above equation, C^{bs} is the cost of T^{bs} , the best tour found so far. The cost of a tour is the travel distance between the allocated customers to the logistics depots, each of which is derived from a randomly generated tour of an ant, using the least distance objective function, and subject to satisfaction of demand and capacity constraints. The initial value

of $\Delta\tau_{ij}^{bs}$ is equal to 0 in step 1 of the ACO algorithm.

4.4.2 High level pseudocode for ACO

The high level pseudocode for the proposed ACO is presented as follows:

1. Set iteration counter $iter = 0$.
2. Initialize values of ACO parameters, such as α , β , q_0 , etc.
3. While ($iter \leq iter_max$) or (improvement in objective function value $\leq 10^{-5}$)

5.1 Set the value of the initial pheromone trail τ_0 , and initial pheromone intensity

$$\tau_{ij} = \tau_0 \text{ for the path from nodes } i \text{ to } j \text{ and } \Delta\tau_{ij} = 0$$

5.2 Construct the tabu lists of all ants, which contain all the unvisited nodes for each ant and the tabu list for the best path found by the ant colonies.

5.3 Randomly place the m ants on the n nodes.

5.4 For $k = 1$ to m do

5.4.1 Generate a random number q .

5.4.2 If $q \leq q_0$, choose the node j to move to according to the state transition rule defined by equation (4-1).

5.4.3 If $q > q_0$, choose the node j to move to with the highest probability p_{ij}^k given by equation (4-2).

5.4.4 Delete the chosen node j from the tabu list S_k of ant k .

5.4.5 Continue moving until the ant k finishes the whole tour.

5.4.6 Update the pheromone trail locally with equation (4-3).

5.5 Evaluate all the feasible tours constructed with respect to objective function value and satisfaction of demand and capacity constraints

5.6 Select the tour with the minimum cost.

5.7 Perform the global update for pheromone trails using equations (4-4) and (4-5).

5.8 Re-construct all tabu lists.

5.9 $iter := iter + 1$.

6 Stop the ACO search process and output the best tour.

4.4.3 Example

Let us consider the case of 2 logistics depots and 6 customers. Set initialization parameters $\alpha=1.0$, $\beta=0.5$, $q=0.8$. At the beginning, a random solution is generated, say, $S_0 = \{C_1(D_1), C_2(D_1), C_3(D_1), C_4(D_2), C_5(D_2), C_6(D_2)\}$ whose objective function value is $\{35, 34, 39, 25, 28, 22, 40\}$. Now we set up six ants, one at each of the customer node. The new generated solution S_1 for the six ants after the first move is given by $S_1 = \{C_1(D_2), C_2(D_2), C_3(D_2), C_4(D_1), C_5(D_1), C_6(D_1)\}$ with objective function value $\{31, 35, 34, 22, 28, 25\}$. Since improvement in objective function value is observed for all ants but except ants on node C_2 and C_5 . Therefore, local update of pheromones takes place on all arcs but except those linking $C_2(D_2)$ and $C_5(D_1)$ and list of unvisited nodes is updated in the tabu list 1. The solution tour now becomes $S_3 = \{C_1(D_2), C_2(D_1), C_3(D_2), C_4(D_1), C_5(D_2), C_6(D_1)\}$ with objective function values

{31,34,34,22,22,25} respectively. Since, each customer can be allocated to only one depot, and only two depots are available for the ants to visit, any further moves leads to violation of tabu list elements. We now move directly to the global update of pheromones and best path S3 in the tabu list 2. Since any more changes will lead to no improvement in objective function value, the algorithm is terminated and best solution S3 = {C1(D2),C2(D1),C3(D2),C4(D1),C5(D2),C6(D1)} is printed.

4.4.4 Advantages of Ant Colony Optimization

The ACO approach is applicable to a broad range of optimization problems and can be used in dynamic applications. Compared to GA, it retains memory of entire colony instead of previous generation only and is less affected by poor initial solutions.

4.4.5 Limitations of Ant Colony Optimization

Like most metaheuristics, sometimes it is difficult to estimate the theoretical speed of convergence. Because of probability rule, mistakes can be made by ant colony algorithms. Also, if the parameters are not correctly chosen, the approach may result in local optimum.

Chapter 5:

Numerical Application

In this section, we will present numerical examples for location & allocation problem

under (a) single factor (objective function $\sum_{j=1}^n c_i y_i + \sum_{i=1}^m \sum_{j=1}^n d_{ij} x_{ij}$) and (b) multi-factor

(objective $\sum_{j=1}^n c_i y_i + \sum_{i=1}^m \sum_{j=1}^n (w_1 * d_{ij} / \sum d_{ij} + w_2 * t_{ij} / \sum t_{ij} + (1 - w_1 - w_2) * c_{ij} / \sum c_{ij}) x_{ij}$).

The normalization method used is $a_{ij} = s_{ij} / \sum(s_{ij})$ where s_{ij} represents an element of matrix $S_{m \times n}$ where $i=1,2,...,m$ and $j=1,...,n$, and the normalized values is given by a_{ij} . Four metaheuristics namely GA, SA, TS, and ACO are applied and tested for solving the location allocation problem. The results are verified and validated against existing models to ensure correctness and assess performance of the proposed approaches.

5.1 Location & allocation using single factor

5.1.1 Input Data

Let us consider a logistics network comprising of 7 depots (D1, D2... D7) and 21 customers (C1, C2 ... C21). The demand, distance, and capacity data for location allocation problem using single factor “distance” is presented in Table 5.1. The distance matrix is presented at the center of the Table 5.1. The customer demands are presented in the last column and depot capacities are present in the last row of Table 5.1.

Customers	Depots							Demand
	D1	D2	D3	D4	D5	D6	D7	
C1	3.4	3.74	4.2	3.2	3.3	4.8	2.1	120
C2	3.10	3.28	3.3	2.7	4.0	3.1	5.8	200
C3	3.8	3.4	3.2	2.9	3.0	2.4	4.8	80
C4	3.5	3.6	3.5	4.9	3.6	2.5	4.9	110
C5	3.7	3.0	3.2	4.6	2.0	3.2	4.6	130
C6	3.6	3.7	3.6	4.7	3.7	3.8	4.7	90
C7	2.88	2.97	7.3	3.31	3.5	3.6	4.5	140
C8	2.5	2.9	3.0	2.83	2.7	3.0	3.2	170
C9	2.6	2.7	4.82	3.2	3.6	3.7	10.8	90
C10	5.8	2.8	3.2	5.3	4.74	4.2	6.1	115
C11	3.1	2.9	6.7	3.0	3.28	3.3	4.4	100
C12	2.4	2.7	2.9	5.0	3.24	6.5	2.0	125
C13	3.5	3.30	3.5	3.6	9.04	2.8	4.5	85
C14	4.2	2.96	2.7	1.0	3.03	3.0	2.3	180
C15	3.1	3.2	2.6	2.74	2.82	3.7	4.1	130
C16	3.2	4.3	2.8	2.88	3.2	3.3	2.8	95
C17	2.7	5.0	3.1	2.92	5.7	6.0	3.1	175
C18	5.9	3.0	2.4	2.47	2.9	3.0	2.4	150
C19	3.5	1.6	3.5	1.30	3.5	3.6	7.5	190
C20	2.7	3.0	5.2	2.96	2.7	3.0	1.2	95
C21	2.6	3.7	4.8	3.28	3.6	6.7	3.8	160
Capacity	800	800	1100	1000	700	1100	900	

Table 5.1: Input distance, capacity and demand data for single factor location allocation problem

Table 5.2 presents the opening costs for the logistics depots.

Depots	D1	D2	D3	D4	D5	D6	D7
Opening costs	14	21	17	15	25	13	22

Table 5.2 Opening costs for logistics depots

The goal is to minimize the facility opening costs and the allocation costs for customers under the demand and capacity constraints of customers and the logistics depots. The

objective function used is $\sum_{j=1}^n c_i y_i + \sum_{i=1}^m \sum_{j=1}^n d_{ij} x_{ij}$. Since the cost and distance data are in different units, they will be normalized before application of metaheuristics GA, SA, TS, and ACO. The formula used for normalization is $a_{ij} = s_{ij} / \sum s_{ij}$ where s_{ij} represents the original data value in matrix_{ij} and a_{ij} represents the normalized value. The normalized values for input distance and facility opening costs are presented in Table 5.3 and Table 5.4 respectively.

	D1	D2	D3	D4	D5	D6	D7
C1	0.0064	0.0070	0.0079	0.0060	0.0062	0.0090	0.0039
C2	0.0058	0.0061	0.0062	0.0051	0.0075	0.0058	0.0109
C3	0.0071	0.0064	0.0060	0.0054	0.0056	0.0045	0.0090
C4	0.0066	0.0067	0.0066	0.0092	0.0067	0.0047	0.0092
C5	0.0069	0.0056	0.0060	0.0086	0.0037	0.0060	0.0086
C6	0.0067	0.0069	0.0067	0.0088	0.0069	0.0071	0.0088
C7	0.0054	0.0056	0.0137	0.0062	0.0066	0.0067	0.0084
C8	0.0047	0.0054	0.0056	0.0053	0.0051	0.0056	0.0060
C9	0.0049	0.0051	0.0090	0.0060	0.0067	0.0069	0.0202
C10	0.0109	0.0052	0.0060	0.0099	0.0089	0.0079	0.0114
C11	0.0058	0.0054	0.0126	0.0056	0.0061	0.0062	0.0082
C12	0.0045	0.0051	0.0054	0.0094	0.0061	0.0122	0.0037
C13	0.0066	0.0062	0.0066	0.0067	0.0169	0.0052	0.0084
C14	0.0079	0.0055	0.0051	0.0019	0.0057	0.0056	0.0043
C15	0.0058	0.0060	0.0049	0.0051	0.0053	0.0069	0.0077
C16	0.0060	0.0081	0.0052	0.0054	0.0060	0.0062	0.0052
C17	0.0051	0.0094	0.0058	0.0055	0.0107	0.0112	0.0058
C18	0.0111	0.0056	0.0045	0.0046	0.0054	0.0056	0.0045
C19	0.0066	0.0030	0.0066	0.0024	0.0066	0.0067	0.0141
C20	0.0051	0.0056	0.0097	0.0055	0.0051	0.0056	0.0022
C21	0.0049	0.0069	0.0090	0.0061	0.0067	0.0126	0.0071

Table 5.3 Normalized distance values

Depots	D1	D2	D3	D4	D5	D6	D7
Normalized Costs	0.1102	0.1654	0.1339	0.1181	0.1969	0.1024	0.1732

Table 5.4 Normalized opening cost values for logistics depots

5.1.2 Application of proposed metaheuristics

5.1.2.1 Genetic Algorithm (GA):

Figure 5.1 presents the results obtained from GA for the numerical example on location allocation problem under distance constraints. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized distance for customer allocation over time. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time (9287920 iterations) after which the best objective function values (1.1118) for opening logistics depots and customer allocations are said to have been obtained.

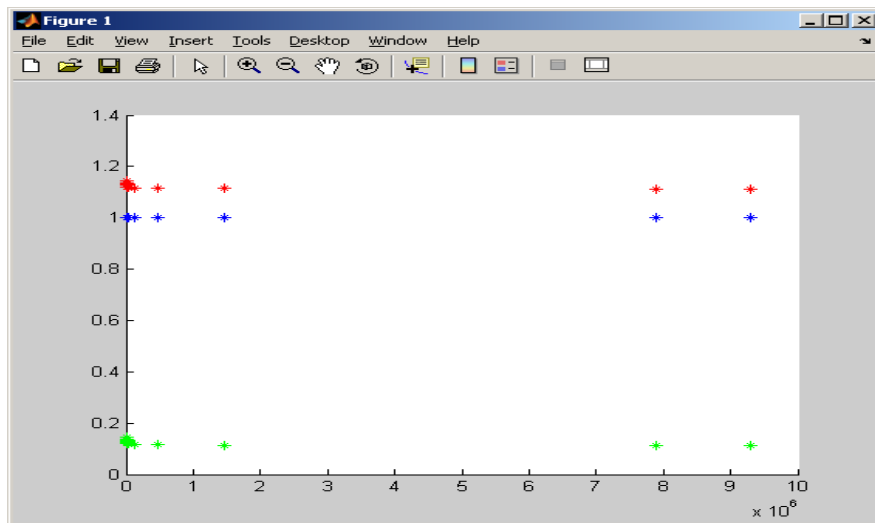


Figure 5.1: Convergence of GA results for single factor location- allocation problem

Table 5.5 depicts the computation results of GA over number of iterations.

Iteration Number	29	254	255	1460753	460754	7894324	9287920
Normalized distance value	0.1441	0.1374	0.1346	0.1137	0.1137	0.1132	0.1118
Normalized cost value	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Normalized distance plus cost distance	1.1441	1.1374	1.1346	1.1137	1.1137	1.1132	1.1118

Table 5.5 Objective function results for single factor location allocation problem using GA

Table 5.6 shows the difference between the initial and final solutions obtained from GA.

It can be seen that all logistics are opened and have customer allocations in the final results.

Customers		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial Solution)		1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots (Final Solution)		7	3	6	6	7	7	2	1	5	6	5	7	7	4	4	7	1	7	2	7	1

Table5.6 Initial and Final Solution for single factor location allocation problem using GA

5.1.2.2 Simulated Annealing (SA)

Figure 5.2 presents the results obtained from SA for the numerical example on location allocation problem under distance constraints. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized distance for customer allocation over time. The red colored dots (upper) represent the total value of the objective function. It can be seen that the results

for costs and distance stabilize over time (819721 iterations) after which the best objective function values (1.1048) for opening logistics depots and customer allocations is said to have been obtained.

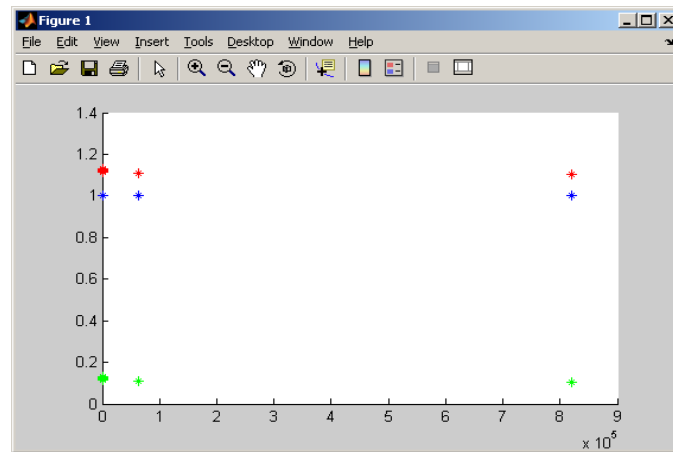


Figure5.2 Convergence of SA results for single factor location- allocation problem

Table 5.7 provides the numerical values for the objective function results for single factor location allocation problem using SA.

Iteration Number	25	27	466	1866	63590	63591	819721
Normalized distance value	0.1320	0.1237	0.1202	0.1123	0.1091	0.1072	0.1048
Normalized cost value	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Normalized distance plus cost distance	1.1320	1.1237	1.1202	1.1123	1.1091	1.1072	1.1048

Table 5.7 Objective function results for single factor location allocation problem using SA

Table 5.8 provides the difference between the initial and final solution obtained using SA.

It can be seen that all logistics depots are opened and allocated to customers.

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial Solution)	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots (Final Solution)	1	4	5	6	5	1	4	5	2	2	5	5	6	4	1	1	3	3	4	7	1

Table 5.8 Initial and Final Solution for single factor location allocation problem using SA

5.1.2.3 Tabu Search (TS)

Figure 5.3 presents the results obtained from Tabu Search (TS) for the numerical example on location allocation problem under distance constraints. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized distance for customer allocation over time. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time (1247890 iterations) after which the best values of objective function (1.1114) for opening logistics depots and customer allocations are said to have been obtained. Table 5.9 provides the objective function results for single factor location allocation problem using TS.

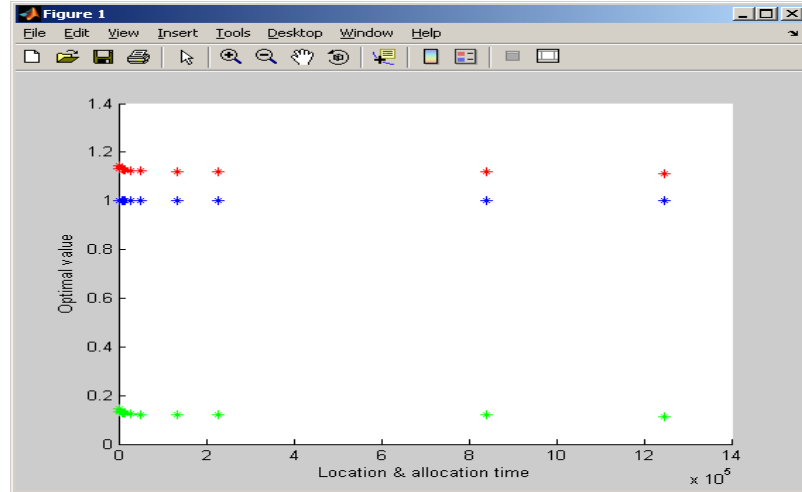


Figure 5.3 Convergence of TS results for single factor location- allocation problem

Iteration Number	1117	2088	2089	1247887	1247888	1247889	1247890
Normalized distance value	0.1433	0.1326	0.1326	0.1114	0.1114	0.1114	0.1114
Normalized cost value	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Normalized distance plus cost distance	1.1433	1.1326	1.1326	1.1114	1.1114	1.1114	1.1114

Table 5.9 Objective function results for single factor location allocation problem using TS

Table 5.10 provides the difference between the initial and final solution for single factor location allocation problem obtained using TS. It can be seen that all the logistics depots are opened and allocated to customers.

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial Solution)	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots (Final Solution)	7	6	6	6	3	7	2	5	1	2	7	7	7	7	1	7	1	4	4	7	5

Table 5.10 Initial and Final Solution for single factor location allocation problem using TS

5.1.2.4 Ant Colony Optimization (ACO)

Figure 5.4 presents the results obtained from ACO for the numerical example on location allocation problem under distance constraints. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized distance for customer allocation over time. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time (4110026 iterations) after which the best values of objective function (0.7704) for opening logistics depots and customer allocations are said to have been obtained.

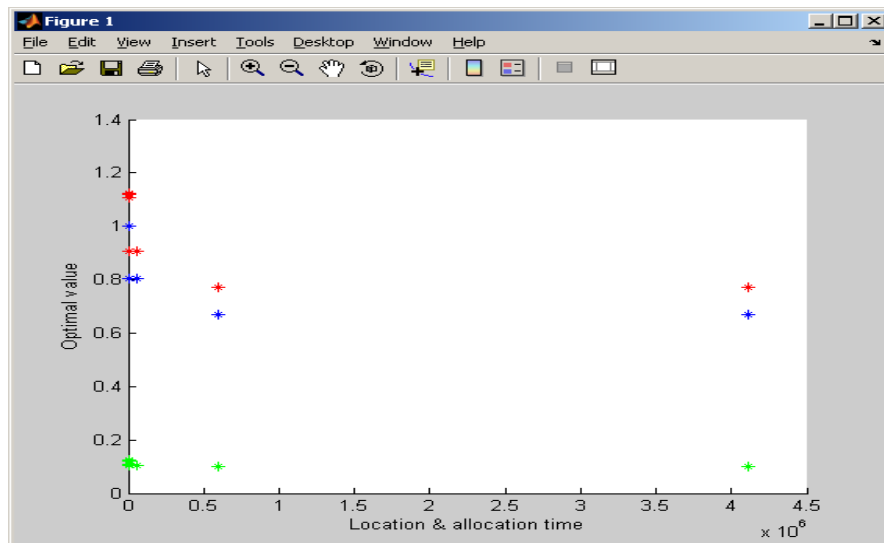


Figure 5.4 Convergence of ACO results for single factor location- allocation problem

Table 5.11 provides the numerical values of the objective function for single factor location allocation problem using ACO.

Iteration Number	2	10	17	3195	59308	598023	4110026
Normalized distance value	0.1244	0.1191	0.1072	0.1047	0.1039	0.1018	0.1011
Normalized cost value	1.0000	1.0000	1.0000	0.8031	0.8031	0.6693	0.6693
Normalized distance plus cost distance	1.1244	1.1191	1.1072	0.9079	0.9070	0.7711	0.7704

Table 5.11 Objective function results for single factor location allocation problem using ACO

Table 5.12 provides the difference between the initial and final solution obtained using ACO. It can be seen that some logistics depots (for example, depot 3 and depot 5) do not get any customer allocations and therefore considered as closed.

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial Solution)	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots	7	4	6	6	2	1	4	7	1	2	2	7	6	4	2	4	7	4	2	7	1

Table 5.12 Initial and Final Solution for single factor location allocation problem using ACO

5.1.2.5 Comparison of GA, SA, TS, and ACO for single factor location allocation

Table 5.13 presents a relative comparison of the final results obtained from the four metaheuristics for the single factor location allocation problem under distance constraints. It can be seen that ACO results propose opening of only 5 logistics depots followed by GA (6 depots), whereas all the depots are open in SA and TS which also justifies their

least total objective function value shown in Table 5.14. .

	Initial Solution	GA	SA	TS	ACO
D1	C1, C2, C3	C8, C17, C21	C21	C9, C15, C17	C6, C9, C21
D2	C4, C5, C6	C7, C19	C10, C19	C7, C10	C5,C10,C11 ,C15, C19
D3	C7, C8, C9	C2	C4	C5	-
D4	C10,C11,C12	C14,C15	C9, C14, C17	C18, C19	C2,C7,C14, C16, C18
D5	C13,C14, C15	C9,C11	C5,C15	C8,C21	-
D6	C16,C17, C18	C3,C4,C10	C2,C3,C6,C7, C8, C13	C2,C3,C4	C3,C4,C13
D7	C19,C20, C21	C1,C5,C6,C12, C13,C16,C18,C20	C1,C11,C12, C16, C18, C 20	C1,C6,C11,C12, C13,C14,C16,C20	C1,C8,C12, C17,C20

Table 5.13 Comparison of Location allocation results for single factor problem

Table 5.14 presents a relative comparison of the computation time and objective function values for the final results obtained from the four metaheuristics for the single factor location allocation problem under distance constraints. It can be seen that ACO performs the fastest in terms of computation time used for generating the results and gives least value for the total objective function equal to 0.7704 when compared to other three metaheuristics.

	GA	SA	TS	ACO
Number of Iterations	10000000	10000000	10000000	10000000
Objective function value distance by normalized	0.1079	0.1050	0.1550	0.1011
Objective function value cost by normalized	0.8898	1.0000	1.0000	0.6693
Objective function value cost plus distance by normalized	0.9977	1.1050	1.1550	0.7704
Computation time	4696.801189 seconds	4965.391465 seconds	5093.268974 Seconds	4296.178412 seconds

Table 5.14 Comparison of performance results for single factor location allocation problem

5.2 Location and allocation using multifactors

5.2.1 Input Data

Let us consider the input data for location allocation problem using multiple factors “distance”, “time” and “cost” for a logistics network comprising of 7 logistics depots (D1, D2... D7) and 21 customers (C1, C2 ... ,C21). The distance matrix, customer demands, and depot capacities are same as presented in Table 5.1. The time matrix is presented in Table 5.15 and the cost matrix in Table 5.16. The depot opening costs are same as in Table 5.2.

Customers	Depots						
	D1	D2	D3	D4	D5	D6	D7
C1	1.0	4.83	3.00	2.7	5.0	3.0	2.1
C2	3.0	2.88	2.86	3.3	4.9	4.1	5.8
C3	4.6	3.76	3.74	4.2	6.1	3.6	4.8
C4	3.4	3.10	3.28	3.3	4.4	2.6	4.9
C5	4.0	3.38	3.24	4.2	6.0	3.4	4.6
C6	13.3	3.4	3.1	3.04	3.63	8.5	4.0
C7	3.1	3.3	3.8	3.31	3.28	3.9	1.8
C8	7.2	3.3	2.8	2.88	2.97	3.3	9.4
C9	2.7	3.0	7.1	2.92	3.0	2.8	2.7
C10	2.9	4.0	8.4	2.47	2.65	2.9	3.5
C11	1.5	2.6	3.5	6.30	3.38	3.7	5.2
C12	2.7	3.0	3.0	2.6	3.1	3.2	2.6
C13	3.6	9.7	5.8	3.28	3.2	4.3	6.8
C14	3.0	3.2	8.6	3.18	2.7	3.0	8.1
C15	3.1	3.2	3.4	3.1	3.04	3.13	3.5
C16	3.2	13.3	1.3	13.8	13.31	3.28	3.2
C17	6.7	13.0	0.3	6.8	2.88	2.97	7.3
C18	2.92	3.05	23.0	3.1	2.92	3.05	2.8
C19	8.47	2.65	3.0	12.4	12.47	2.65	2.9
C20	3.30	3.38	3.6	3.5	3.30	3.38	3.7
C21	9.96	8.14	3.0	3.2	2.96	23.1	3.2

Table 5.15 Time Matrix

Customers	Depots						
	D1	D2	D3	D4	D5	D6	D7
C1	2.9	3.2	3.5	3.14	3.15	3.0	2.1
C2	3.9	4.0	4.3	3.62	3.60	4.1	5.8
C3	3.5	3.6	3.5	3.14	3.12	3.6	4.8
C4	3.5	3.6	3.6	3.19	3.17	3.6	4.9
C5	3.3	3.4	3.0	2.99	3.07	3.4	4.6
C6	3.3	3.4	3.1	3.04	3.13	3.5	4.7
C7	3.1	3.3	3.8	3.31	3.28	3.2	1.8
C8	2.5	2.9	3.0	2.83	3.00	2.7	3.0
C9	3.2	3.3	3.0	2.88	2.86	3.3	4.1
C10	4.0	4.1	4.6	3.76	3.74	4.2	6.1
C11	3.1	3.3	3.4	3.10	3.28	3.3	4.4
C12	3.1	3.4	4.0	3.38	3.24	3.2	2.0
C13	2.8	3.0	3.2	2.95	3.04	2.8	2.5
C14	3.0	3.2	3.6	3.18	3.03	3.0	2.3
C15	3.1	3.2	2.6	2.74	2.82	3.2	4.1
C16	3.2	3.3	2.8	2.88	2.97	3.3	4.4
C17	2.7	3.0	3.1	2.92	3.05	2.8	2.7
C18	2.9	3.0	2.4	2.47	2.65	2.9	3.5
C19	3.5	3.6	3.5	3.30	3.38	3.7	5.2
C20	2.7	3.0	3.2	2.96	3.14	3.0	3.5
C21	3.6	3.7	3.8	3.28	3.27	3.7	5.1

Table 5.16 Cost matrix

The goal is to minimize the facility opening costs and the allocation costs for customers under multifactors and the demand and capacity constraints of customers and the logistics depots. The objective function used for the multifactor location allocation problem is

$$\text{given by } \sum_{j=1}^n c_i y_i + \sum_{i=1}^m \sum_{j=1}^n (w_1 * d_{ij} / \sum d_{ij} + w_2 * t_{ij} / \sum t_{ij} + (1 - w_1 - w_2) * c_{ij} / \sum c_{ij}) x_{ij} .$$

Since the cost, travel time and distance data are in different units, they will be normalized before application of metaheuristics GA, SA, TS, and ACO. The formula used for

normalization is $a_{ij} = s_{ij} / \sum s_{ij}$ where s_{ij} represents the original data value in matrix_{ij} and a_{ij} represents the normalized value. Using $w_1=0.2$, $w_2=0.3$, and $w_3=1-w_1-w_2$, the total objective function values obtained are presented in Table 5.17.

	D1	D2	D3	D4	D5	D6	D7
C1	0.0052	0.0068	0.0068	0.0058	0.0066	0.0067	0.0039
C2	0.0066	0.0068	0.0071	0.0062	0.0074	0.0071	0.0109
C3	0.0071	0.0067	0.0065	0.0061	0.0067	0.0061	0.0090
C4	0.0066	0.0066	0.0066	0.0070	0.0066	0.0059	0.0092
C5	0.0066	0.0062	0.0058	0.0069	0.0060	0.0063	0.0086
C6	0.0094	0.0066	0.0061	0.0066	0.0064	0.0082	0.0086
C7	0.0057	0.0060	0.0091	0.0062	0.0063	0.0065	0.0049
C8	0.0061	0.0056	0.0056	0.0053	0.0055	0.0054	0.0077
C9	0.0055	0.0058	0.0079	0.0056	0.0058	0.0063	0.0111
C10	0.0082	0.0069	0.0090	0.0075	0.0073	0.0075	0.0107
C11	0.0054	0.0058	0.0083	0.0067	0.0062	0.0063	0.0085
C12	0.0053	0.0059	0.0066	0.0070	0.0060	0.0079	0.0039
C13	0.0059	0.0078	0.0070	0.0060	0.0091	0.0057	0.0071
C14	0.0063	0.0059	0.0078	0.0048	0.0056	0.0056	0.0061
C15	0.0058	0.0060	0.0051	0.0053	0.0054	0.0063	0.0075
C16	0.0060	0.0098	0.0048	0.0087	0.0088	0.0062	0.0070
C17	0.0063	0.0098	0.0050	0.0066	0.0072	0.0071	0.0067
C18	0.0071	0.0057	0.0107	0.0048	0.0052	0.0056	0.0058
C19	0.0081	0.0054	0.0064	0.0078	0.0091	0.0066	0.0104
C20	0.0053	0.0058	0.0073	0.0057	0.0057	0.0058	0.0053
C21	0.0081	0.0083	0.0075	0.0061	0.0062	0.0144	0.0083

Table 5.17 Normalized data for multifactor cost, time and distance

5.2.2 Application of metaheuristics

5.2.2.1 Genetic Algorithm (GA)

Figure 5.5 presents the results obtained from Genetic algorithm for the multifactor location allocation problem where multifactors considered are cost, time and distance.

The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized multifactors used for customer allocation to logistics facilities. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time (4854673 iterations) after which the best values of objective function (1.1182) for opening logistics depots and customer allocations are said to have been obtained. Table 5.18 provides the numerical values for the costs and distances over time.

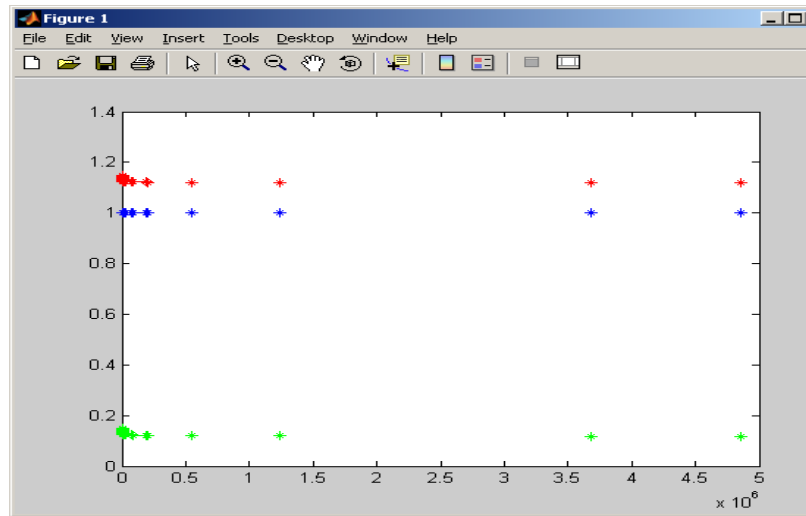


Figure 5.5 Convergence of GA results for multifactor location- allocation problem

Iteration Number	6	8	9	1243586	3685608	4854673
Normalized multifactor value	0.1453	0.1426	0.1399	0.1197	0.1184	0.1182
Normalized cost value	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Normalized cost plus multifactor value	1.1453	1.1426	1.1399	1.1197	1.1184	1.1182

Table 5.18 Objective function results for multifactor location allocation problem using GA

Table 5.19 provides the difference between the initial and final solution obtained using GA for the multifactor case. It can be seen that all logistics depots are opened and allocated to customers using the proposed GA.

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial solution)	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots (final result)	7	4	4	6	2	3	7	5	5	6	5	7	1	2	5	3	1	4	2	7	5

Table 5.19 Initial and Final Solution for multifactor location allocation problem using GA

5.2.2.2 Simulated Annealing (SA)

Figure 5.6 presents the results obtained from Simulated Annealing algorithm for the multifactor location allocation problem where multifactors considered are cost, time and distance. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized multifactors used for customer allocation to logistics facilities. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time after (4194943 iterations) after which the best objective function values (1.1259) for opening logistics depots and customer allocations are said to have been obtained.

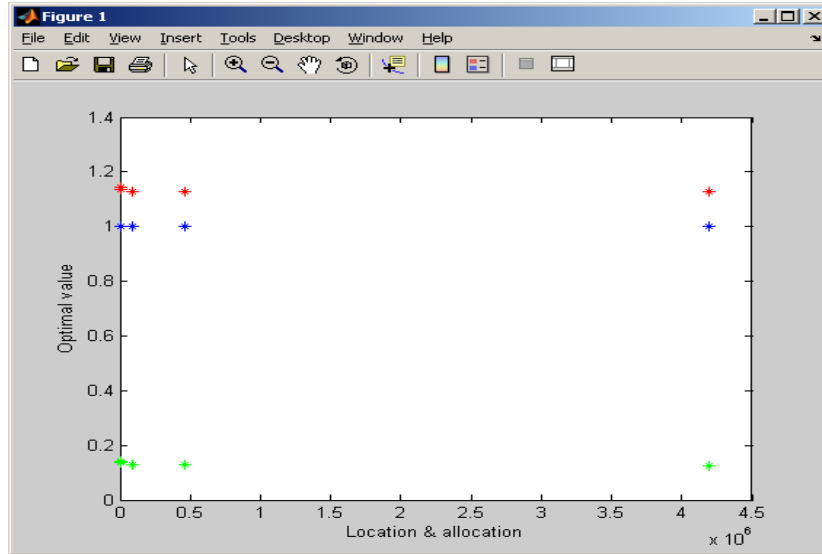


Figure 5.6 Convergence of SA results for multifactor location- allocation problem

Table 5.20 provides the objective function values for multifactor location allocation problem over time.

Iteration Number	29	112	89070	466276	4194943
Normalized multifactor value	0.1432	0.1353	0.1290	0.1280	0.1259
Normalized cost value	1.0000	1.0000	1.0000	1.0000	1.0000
Normalized cost plus multifactor value	1.1432	1.1353	1.1290	1.1280	1.1259

Table 5.20 Objective function results for multifactor location allocation problem using SA

Table 5.21 provides final solution obtained using SA for the multifactor case. It can be seen that all logistics depots are opened and allocated to customers.

Customers		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial solution)		1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots (final solution)		7	3	6	1	6	7	7	5	1	6	5	7	7	4	2	7	1	7	2	7	4

Table 5.21 Initial and Final Solution for multifactor location allocation problem using SA

5.2.2.3 Tabu Search

Figure 5.7 presents the results obtained from Tabu Search algorithm for the multifactor location allocation problem where the multifactors are cost, time and distance. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized multifactors used for customer allocation to logistics depots. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time (6003819 iterations) after which the best values of objective function (0.9483) for opening logistics depots and customer allocations are said to have been obtained.

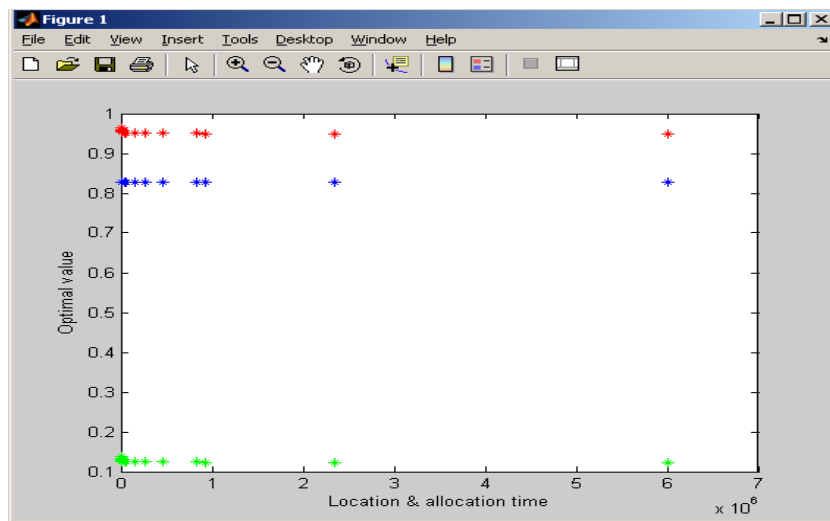


Figure 5.7 Convergence of TS results for multifactor location- allocation problem

Table 5.22 provides the numerical values for the total costs over time.

Iteration Number	1	10	935084	935085	2349407	6003819
Normalized multifactor value	0.1388	0.1388	0.1229	0.1229	0.1227	0.1215
Normalized cost value	0.8268	0.8268	0.8268	0.8268	0.8268	0.8268
Normalized cost plus multifactor value	0.9656	0.9656	0.9497	0.9497	0.9494	0.9483

Table 5.22 Objective function results for multifactor location allocation problem using TS

Table 5.23 provides the difference between the initial and final solution obtained using TS for the multifactor case. It can be seen that some logistics depots (example, depot 7) are closed since they do not get any customer allocations.

Customers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial solution)	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7
Logistics depots (final solution)	1	6	2	3	5	3	1	4	2	2	2	2	6	4	3	3	3	4	2	1	3

Table 5.23 Initial and Final Solution for multifactor location allocation problem using TS

5.2.2.4 Ant Colony Optimization (ACO)

Figure 5.8 presents the results obtained from ACO algorithm for the multifactor location allocation problem where multifactors considered are cost, time and distance. The blue color dots (middle curve) shows the normalized facility opening cost values and the green colored dots (lower curve) show the normalized multifactors used for customer allocation

to logistics depots. The red colored dots (upper curve) represent the total value of the objective function. It can be seen that the results for costs and distance stabilize over time (6959966 iterations) after which the best objective function values (1.000) for opening logistics depots and customer allocations are said to have been obtained.

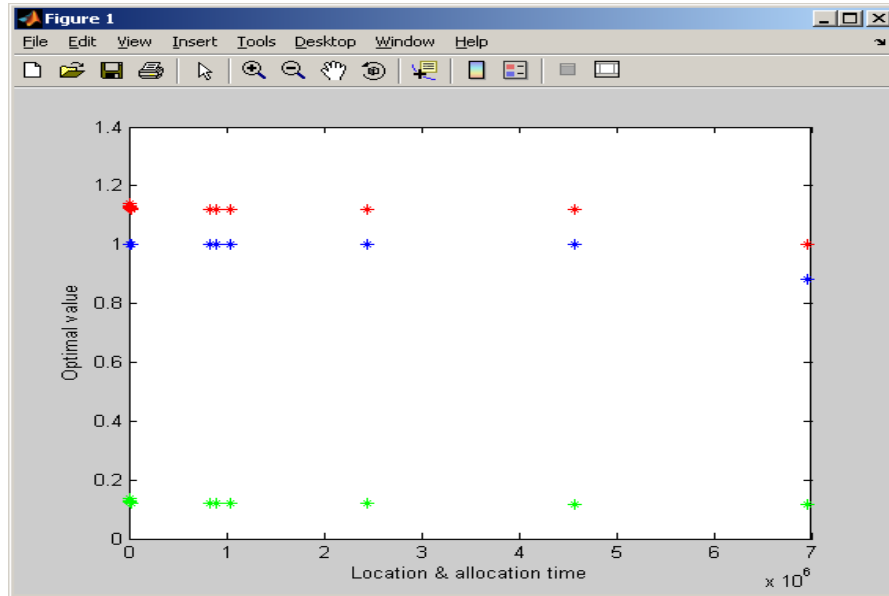


Figure 5.8 Convergence of ACO results for multifactor location- allocation problem

Table 5.24 provides the numerical values for the total objective function cost values over time.

Iteration Number	1	2	1044459	2441079	4580238	6959966
Normalized multifactor value	0.1386	.1307	0.1197	0.1188	0.1182	0.1181
Normalized cost value	1.0000	1.0000	1.0000	1.0000	1.0000	0.8819
Normalized cost plus multifactor value	1.1386	.1307	1.1197	1.1188	1.1182	1.0000

Table 5.24 Objective function results for multifactor location allocation problem using ACO

Table 5.25 provides the difference between the initial and final solution obtained using

ACO for the multifactor case. It can be seen that some logistics depots (example, depot 4) do not get any customer allocations and are therefore closed.

Customers		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Logistics depots (Initial solution)	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6	7	7	7	
Logistics depots (Final solution)		7	2	6	6	3	3	7	2	5	5	1	7	3	6	5	3	3	6	2	5	5

Table 5.25 Initial and Final Solution for multifactor location allocation problem using ACO

5.2.2.5: Comparison of GA, SA, TS, and ACO for multifactor location allocation

Table 5.26 presents a relative comparison of the final results obtained from the four metaheuristics for the multifactor location allocation problem. It can be seen that only 6 depots are opened in ACO and TS making them least costly solutions (Table 5.27) for location allocation as compared to GA and SA.

Depots	Initial Solution	GA	SA	TS	ACO
D1	C1,C2,C3	C2,C8,C11	C4,C9,C17	C1,C7 ,C20	C11
D2	C4,C5,C6	C10,C15	C15,C19	C3,C9,C10, C11,C12, C19	C2,C8,C19
D3	C7,C8,C9	C5	C2	C4,C6,C15, C16,C17, C21	C5,C6,C13, C16,C17
D4	C10,C11,C12	C3,C14	C14,C21	C8,C14,C18	-
D5	C13,C14,C15	C4,C21	C8,C11	C5	C9,C10,C15, C20,C21
D6	C16,C17,C18	C9,C17,C19	C3,C5,C10	C2,C13	C3,C4,C14, C18
D7	C19,C20,C21	C1,C6,C7C2, C13,C16,C18 C20	C1,C6,C7,C12 ,C13,C16,C1, C20	-	C1,C7,C12

Table 5.26 Comparison of model results for multifactor location allocation problem

Table 5.27 presents a relative comparison of the computation time and objective function values for multifactor location allocation problem. It can be seen that TS performs the best followed by ACO in terms of objective function value. The metaheuristic SA is the fastest in terms of computation time followed by GA.

	GA	SA	TS	ACO
Iteration times	10000000	10000000	10000000	10500000
Objective multifactor value	0.1271	0.1259	0.1215	0.1181
Objective normalized cost	1.0000	1.0000	0.8268	0.8819
Objective normalized cost plus multifactor value	1.1271	1.1259	0.9483	1.0000
Computation time	4844.64 seconds	3142.55 seconds	6702.69 seconds	10867.25 Seconds

Table 5.27 Comparison of model performance for multifactor location allocation problem

5.3 Model Verification

To verify the model results, we tested our model under three difference scenarios for the same numerical example presented in section 5.1 (multifactor case).

- *Scenario 1:* In the scenario 1, the opening cost is same for all the 7 logistics facilities and is equal to \$100,000. Demand and capacity constraints are not considered.
- *Scenario 2:* In the scenario 2, we ignore the facility opening costs by setting them equal to 0, in other words all facilities are considered open and customers are allocated to them using different metaheuristic approaches. Demand and capacity

constraints are not considered.

- *Scenario 3:* In the scenario 3, the opening costs for facilities are different.

Demand and capacity constraints are considered.

The demand data for customers, opening costs of logistics facilities and their capacities and average transit time between the logistics facilities and the customers is presented in Table 5.28-5.30.

Customers	Depots						
	D1	D2	D3	D4	D5	D6	D7
Wallingford (C1)	104795	102450	93708	95787	95414	106554	113644
Ankeny (C2)	33370	35380	39670	39887	40267	113644	253360
Posen (C3)	101682	106509	115836	117401	118249	104356	82507
W.Chicago (C4)	99334	104161	113488	114988	115901	102008	80159
Indianapolis (C5)	94008	98196	110421	107535	104309	96611	75274
Louisville (C6)	147009	153570	170860	165540	160131	149225	116064
Boston (C7)	134765	1311487	119613	122454	124202	137169	329263
Baltimore (C8)	396064	382395	345242	176126	165874	190847	162106
Westland (C9)	175591	183075	197843	199866	201282	179738	151417
Blaine (C10)	55259	58877	65821	66995	67631	57264	40833
Charlotte (C11)	126005	128567	126005	126213	117697	123028	97758
Auburn (C12)	52472	51189	46318	47543	50589	53435	63643
Kenvil (C13)	376864	374128	344212	351326	340564	388174	367397
Menands (C14)	25422	249092	229615	234783	248297	261547	276387
Columbus (C15)	103026	107812	119972	116538	113610	106235	85180
W.Chester (C16)	98128	102294	114455	111077	107812	100774	79662
Philadelphia (C17)	136459	132669	121044	246385	234508	271781	122560
Pittsburgh (C18)	126983	131658	138986	285553	269759	259651	106198
Nashville (C19)	77868	80993	88301	852203	82203	76961	57305
Richmond (C20)	148193	146717	131820	134912	126340	142361	119524
Milwaukee (C21)	75505	78587	85880	87319	87832	76892	60096
Total	2918303	2939805	2919111	3033453	2972370	3019095	2612318

Table 5.28: Customer Demand data

S.No.	Depots	Opening Costs	Capacity
1	D1 (Baltimore)	3215569	2000000
2	D2 (Williamsport)	3327844	2000000
3	D3 (Wheeling)	3000000	2000000
4	D4 (Pittsburgh)	3197605	2000000
5	D5 (Erie)	3094311	2000000
6	D6 (Harrisburg)	3251497	2000000
7	D7 (Boston)	1500000	2000000

Table 5.29: Opening costs of facilities

	D1	D2	D3	D4	D5	D6	D7
Wallingford (C1)	269	323	474	440	485	251	127
Ankeny (C2)	955	885	734	733	722	916	1154
Posen (C3)	652	582	440	429	419	613	927
W.Chicago (C4)	699	629	487	476	466	660	974
Indianapolis (C5)	548	478	291	344	395	515	877
Louisville (C6)	565	495	318	371	427	541	904
Boston (C7)	389	444	595	560	524	372	0
Baltimore (C8)	0	76	263	232	347	78	386
Westland (C9)	495	425	284	273	262	456	770
Blaine (C10)	1050	980	839	827	817	1011	1121
Charlotte (C11)	426	376	419	428	554	451	808
Auburn (C12)	434	489	640	606	561	417	81
Kenvil (C13)	188	215	367	332	368	144	247
Menands (C14)	322	356	507	462	358	285	172
Columbus (C15)	384	314	127	180	227	350	713
W.Chester (C16)	194	248	399	365	415	177	195
Philadelphia (C17)	100	174	324	290	392	104	298
Pittsburgh (C18)	230	160	61	0	127	191	554
Nashville (C19)	658	597	475	528	584	671	1034
Richmond (C20)	149	169	347	325	440	227	531
Milwaukee (C21)	749	679	537	529	516	710	1024

Table 5.30: Average transit time (in minutes) between the depot i and customer j

The unit shipping cost between the logistics facilities/depots and customers is shown in Table 5.31.

	D1	D2	D3	D4	D5	D6	D7
Wallingford (C1)	2.9	3.2	3.5	3.14	3.15	3	2.1
Ankeny (C2)	3.9	4	4.3	3.62	3.6	4.1	5.8
Posen (C3)	3.5	3.6	3.5	3.14	3.12	3.6	4.8
W.Chicago (C4)	3.5	3.6	3.6	3.19	3.17	3.6	4.9
Indianapolis (C5)	3.3	3.4	3	2.99	3.07	3.4	4.6
Louisville (C6)	3.3	3.4	3.1	3.04	3.13	3.5	4.7
Boston (C7)	3.1	3.3	3.8	3.31	3.28	3.2	1.8
Baltimore (C8)	2.5	2.9	3	2.83	3	2.7	3
Westland (C9)	3.2	3.3	3	2.88	2.86	3.3	4.1
Blaine (C10)	4	4.1	4.6	3.76	3.74	4.2	6.1
Charlotte (C11)	3.1	3.3	3.4	3.1	3.28	3.3	4.4
Auburn (C12)	3.1	3.4	4	3.38	3.24	3.2	2
Kenvil (C13)	2.8	3	3.2	2.95	3.04	2.8	2.5
Menands (C14)	3	3.2	3.6	3.18	3.03	3	2.3
Columbus (C15)	3.1	3.2	2.6	2.74	2.82	3.2	4.1
W.Chester (C16)	3.2	3.3	2.8	2.88	2.97	3.3	4.4
Philadelphia (C17)	2.7	3	3.1	2.92	3.05	2.8	2.7
Pittsburgh (C18)	2.9	3	2.4	2.47	2.65	2.9	3.5
Nashville (C19)	3.5	3.6	3.5	3.3	3.38	3.7	5.2
Richmond (C20)	2.7	3	3.2	2.96	3.14	3	3.5
Milwaukee (C21)	3.6	3.7	3.8	3.28	3.27	3.7	5.1

Table 5.31: Unit shipping cost (in dollars) and demand (in units of products)

The data used for the three scenarios are shown in Table 5.32.

	D1	D2	D3	D4	D5	D6	D7
Scenario 1	100,000	100,000	100,000	100,000	100,000	100,000	100,000
Scenario 2	0	0	0	0	0	0	0
Scenario 3	100,000	70,000	20,000	40,000	80,000	120,000	60,000

Table 5.32: Scenarios for verification

The scenarios were run for 100,000 iterations. The results for the three scenarios are presented in Table 5.33-5.35 respectively. It can be seen from the results of Table 5.33 that Ant Colony provides the least cost objective function value for the three scenarios. Besides, the results of the four meta-heuristics for scenario 1 and 2 follow identical pattern since they consider equal facility opening costs or zero costs and therefore do not contribute towards the total objective function value. This verifies the correctness of our model results with respect to the objective function.

Scenario	Initial Solution	GA	SA	TS	ACO
1	1.1541	1.105	1.1084	0.96	0.9454
2	0.1631	0.0999	0.1025	0.1073	0.0879
3	1.1542	1.1135	1.1141	0.9602	0.9264

Table 5.33: Objective function value

Table 5.34 presents the computation time for the four metaheuristics. It can be seen that SA takes the least computation time in first two scenarios. Besides, the results of the four meta-heuristics for scenario 1 and 2 follow identical pattern in terms of computation times since they consider equal facility opening costs or zero costs and therefore do not contribute towards the total objective function value. The GA algorithm performs fastest in scenario 3. This verifies the correctness of our model results with respect to the computation time.

Scenario	GA	SA	TS	ACO
1	378.22	203.37	1779.21	528.91
2	351.49	208.57	1431.79	1083.93
3	492.528	505.389	1251.28	1145.89

Table 5.34 Computation Time (in seconds)

The location-allocation results for scenario 1, 2 and 3 can be seen in Table 5.35-5.37 respectively. It can be seen in Table 5.35 that only 6 logistics facilities (depots) are open when applying Tabu Search and Ant Colony Optimization which also confirms their least cost objective function values (Row 1, Table 5.33)

Depots	Initial Solution	Final Solution			
		GA	SA	TS	ACO
D1	C3,C17	C13, C16	C1,C2,C8, C17	C20	C8,C11, C16,C20
D2	C4,C8	C5,C8	C3,C4,C5,C6, C11,C14	-	-
D3	C5,C10	C11,C19	C15,C18,C19	C5,C17	C5,C6,C9, C15,C19
D4	C2,C7	C18,C21	C9	C9	C2,C21, C18
D5	C1,C6,C9, C11,C13, C15	C2,C3,C4,C6, C9,C10	C10	C2,C3,C4,C6, C10,C15,C19 ,C21	C3,C4,C10
D6	C12,C14	C1,C12,C14 C15,C17,C20	C13,C16,C20, C21	C11,C13,C18	C13,C14, C17
D7	C16,C18, C19,C20, C21	C7	C7,C12	C1,C7,C8, C12 ,C14,C16	C1,C7,C12

Table 5.35 Location Allocation Results for Scenario 1 (in seconds)

Form the results of scenario 2 (Table 5.36), it can be seen that only 6 logistics facilities(depots) are open when applying Ant Colony Optimization which also confirms its least cost objective function value (Row 2, Table 5.33).

Depots	Initial Solution	Final solution			
		GA	SA	TS	ACO
D1	C2,C9,C19	C8,C13,C20	C18,C20	C8,C16,C20	C8,C11,C17,C20
D2	C4,C8	C2,C3	C4,C11,C13	C7,C11,C19	-
D3	C6,C18	C4,C19	C3,C5,C15,C19	C6	C5,C6,C15
D4	C12,C14,C20	C11,C18,C21	C9,C10	C4,C15,C21	C9,C10,C18,C21
D5	C11,C13,C16	C9,C10,C15	C21	C2,C3,C9,C14,C18	C2,C3,C4,C19
D6	C1,C5,C7,C10	C5,C6,C16,C17	C2,C6,C8,C16	C1,C5,C10,C13,C17	C13
D7	C3,C15,C17,C21	C1,C7,C12,C14	C1,C7,C12,C14,C17	C12	C1,C7,C12,C14,C16

Table 5.36 Location Allocation Results for Scenario 2 (in seconds)

From the results of scenario 3 (Table 5.37), it can be seen that only 6 logistics facilities(depots) are open when applying Ant Colony Optimization which also confirms its least cost objective function value (Row 2, Table 5.33).

Depots	Initial Solution	Final solution			
		GA	SA	TS	ACO
D1	C3,C6,C16	C8,C19,C20	C4,C5,C8	C13,C15, C17	C8,C16,C17, C20
D2	C7,C11	C2,C3	C2,C20	C5,C11,C16, C18,C19, C20,C21	-
D3	C17	C6	C6	-	C5,C15
D4	C8,C14	C15,C21	C15,C19	C4,C10	C4,C6,C10, C18,C19
D5	C5,C21	C4,C9	C5	C3,C6,C9	C2,C3,C9, C21
D6	C18,C19, C20	C10,C11, C18	C11,C17,C18	C2,C8	C11,C13
D7	C1,C2,C4, C9,C10,C12, C13,C15	C1,C5,C7, C12,C13,C1 4,C16,C17	C1,C7,C9, C10,C12,C13, C14, C16,C21	C1,C7,C12, C14	C1,C7,C12, C14

Table 5.37 Location Allocation Results for Scenario 3 (in seconds)

5.4 Model Validation

To perform validation of model results, we took the numerical case study presented in Zhou et al (2003) which is described as follows:

Let us denote V as a set of nodes representing m customers, U as a set of nodes representing r warehouses, and E as a set of edges representing a connection between customers and warehouses. On each edge (i, j) there are two objective coefficients c_{ij} denoting unit shipping cost and t_{ij} denoting transit time between warehouse j and its customer i . At each customer node i , customer demand is denoted as v_i , and at each warehouse j , its capacity is denoted as q_j . Using the above notations, the bi-criteria

multiple warehouse allocation problem is formulated as follows:

$$\text{Minimize } f(\mathbf{x}) = w_1 \cdot f_1(\mathbf{x}) + w_2 \cdot f_2(\mathbf{x})$$

where

w_1 and w_2 are constants representing weights for $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$, respectively.

$$f_1(x) = \sum_{i=1}^m \sum_{j=1}^r v_i c_{ij} x_{ij} \quad (\text{Minimize shipping costs})$$

$$f_2(x) = \sum_{i=1}^m \sum_{j=1}^r t_{ij} x_{ij} \quad (\text{Minimize total transit time between warehouses and customers allocated to them})$$

Subject to

$$\sum_{j=1}^r x_{ij} = 1, i = 1, 2, \dots, m \quad (\text{Each customer is allocated to only one warehouse})$$

$$\sum_{i=1}^m v_i x_{ij} \leq q_j, j = 1, 2, \dots, r \quad (\text{Total demand of customers does not exceed the capacity of warehouses serving them})$$

$$x_{ij} = \begin{cases} 1 & \text{If customer } i \text{ is allocated to warehouse } j, i=1,2,\dots,m; j=1,2,\dots,r \\ 0 & \text{otherwise} \end{cases}$$

The shipping cost data and transit time data is same as in Table 5.31 and Table 5.30. The demand of customers and capacity of warehouses is obtained from Table 5.28 and Table 5.29 respectively. The 4 metaheuristics and Zhou et al (2003) were compared under 7 problem scenarios where each scenario allocates different weight values to the shipping cost and transit time functions. Table 5.38 presents the details of these scenarios.

Scenario	Weight w1	Weight w2
1	0.1	0.9
2	0.25	0.75
3	0.4	0.6
4	0.55	0.45
5	0.7	0.3
6	0.85	0.15
7	0.95	0.05

Table 5.38 Weights scenario description

In their approach, Zhou et al (2003) propose 7 Pareto optimal solutions. We have compared our results with each one of them in terms of cost and time. The metaheuristics were run for 100000 iterations and the results obtained are presented in Table 5.39 and 5.40 respectively. It can be seen in Table 5.39 that our four metaheuristics perform better than the results of Zhou et al (2003) in terms of transit time (objective function f1) for all the 7 Pareto optimal solutions with ACO performing best in 5/7 scenarios.

Scenario	Initial Soln	GA	SA	TS	ACO	Zhou et al (2003)
1 (w1 =0.1, w2=0.9)	175.29	110.41	104.71	103.20	97.95	128.49 126.13 125.97 125.65 123.62 123.30 121.09
2 (w1 =0.25, 2=0.75)	175.29	103.77	104.71	101.90	101.21	
3 (w1 =0.4, w2 =0.6)	175.29	101.21	102.02	110.17	97.22	
4 (w1 =0.55, w2 =0.45)	175.29	105.50	103.98	109.22	101.17	
5 (w1 =0.7, w2 =0.3)	175.29	106.10	100.85	115.36	98.46	
6 (w1 =0.85, w2=0.15)	175.29	102.06	98.64	105.85	98.99	
7 (w1 =0.95, w2 =0.05)	175.29	99.60	98.67	96.47	102.16	

Table 5.39: Transit Time Results (in seconds)

From the results of Table 5.40, we can say that that our four meta-heuristics perform better than the results of Zhou et al (2003) in terms of shipping cost (objective function f2) for all the 7 Pareto optimal solutions with SA performing best in 6/7 scenarios.

Scenario	Initial Soln	GA	SA	TS	ACO	Zhou et al (2003)
1 (w1 =0.1,w2 =0.9)	8783300	7162000	6836400	7038000	6884000	7,924,037.50 7,930,047.50 7,931,645.50 7,943,386.00 7,952,062.50 7,963,803.00 7,977,486.50
2 (w1 =0.25, 2=0.75)	8783300	7240200	6863400	6962100	6870500	
3 (w1 =0.4, w2 =0.6)	8783300	6870500	6844100	7048100	6898700	
4 (w1 =0.55, w2=0.45)	8783300	7209200	6836400	7129100	6868200	
5 (w1 =0.7, w2 =0.3)	8783300	7054500	6863000	7073900	6890800	
6 (w1 =0.85, w2=0.15)	8783300	6862900	6894800	7268600	6904500	
7 (w1 =0.95, w2=0.05)	8783300	6877500	6876500	6910900	6949900	

Table 5.40: Shipping Cost Results

The comparison of our results with Zhou et al (2003) for all the 7 Pareto optimal solutions in Table 5.39-5.40 show better performance of the proposed metaheuristics in terms of transit time and shipping costs under the seven weight scenarios listed in Table 5.38. This validates the results of our study.

Conclusions and future works

6.1 Conclusions

In this thesis, we address the problem of multiobjective capacitated location allocation problem on logistics networks. The distinction between the location allocation problem treated in this thesis and the traditional location allocation problem lies in its multiobjective and dynamic nature. The multiple objectives considered are travel time, travel distance, travel cost etc. and developed based on practical constraints such as presence of congestion, timing and access restrictions imposed by municipal administrations in urban areas etc. The dynamic aspect means the results of location allocation are not fixed forever but vary with change in municipal access or timing regulations, congestion, or land, material and labor costs on logistics networks.

The multiobjective capacitated location allocation problem can be categorized into two sub-problems firstly, the location problem, that is which logistics facilities should be opened and where and secondly, the allocation problem, that is how to perform customer allocations to logistics depots to ensure timely service for customers. The problem is studied under two cases. In the first case, opening costs of the facilities and only one criterion (distance) is used. In the second case, opening costs of the facilities and multiple criteria (distance, travel cost, travel time) are used.

Four metaheuristics namely Genetic algorithms (GA), Simulated annealing (SA), Tabu search (TS), and Ant colony optimization (ACO) are proposed to address the problem. Since, the problems involve multiple criteria (factors), normalization is performed before aggregating them into the objective function using the weighted sum method. The models are tested under various problem instances and results compared with some existing models to ensure validity of results. From our computational experiments, it emerged that no metaheuristic performs best under all circumstances; it depends upon the nature of the problem, its size and the level of details involved. However, in majority of the test cases considered in our study, Ant colony optimization (ACO) showed better performance over others.

6.2 Future work

To extend the research work performed in this thesis, we propose the following future works:

- Testing of proposed metaheuristics on real problem instances.
- More rigorous model verification and validation on large network sizes
- Develop hybrid approaches based on the proposed metaheuristics and other approaches available in the literature. For example, screening of facility locations using multicriteria decision making approaches such as AHP and then allocation using heuristics, metaheuristics or exact approaches.
- Combining routing with location-allocation problem

- Modeling location allocation under stochastic demand
- Integration of barriers in location planning of logistics facilities.

References:

1. Abduinnour-Helm S. (1998), A hybrid heuristic for the uncapacitated hub location problem , European Journal of Operational Research, 106(2-3), 489-499.
2. Alander J.T. (1992), On optimal population size of genetic algorithms, Proceedings of computer systems and software engineering, 6th annual European computer conference, IEEE Computer Society, 65–70.
3. Al-khedhairi A (2008), Simulated annealing metaheuristic for solving P-median problem, Int. J. Contemp. Math. Sciences, 3(28), 1357 – 1365.
4. Ambrosino D., Scutella M. G. (2005), Distribution network design: new problems and related models, European Journal of Operational Research, 165(3), 610-624.
5. Anand S., Knott K. (1986), Computer assisted models used in the solution of warehouse location-allocation problems, Computers & Industrial Engineering, 11(1-4), 100-104.
6. Armour G, Buffa E (1965), A Heuristic Algorithm and Simulation Approach to Relative Location of Facilities, Management Science, 9(2), 294-309.
7. Arora, S., Raghavan, P., Rao, S. (1998), Approximation schemes for Euclidean k -medians and related problems. In: Proceedings of the 30th Annual ACM Symposium on Theory of Computing, 106–113.
8. Azarmand Z., Neishabouri E.(2009), Location Allocation Problem, Facility Location : Contributions to Management Science, 93-109.
9. Badri M A (1999), Combining the analytic hierarchy process and goal programming

- for global facility location-allocation problem, *International Journal of Production Economics*, 62, 237–248.
10. Banks J (1998), *Handbook of simulation principle, methodology, advances, applications, and practice*, p 864, Wiley, ISBN 0-471-13403-1, p324-329.
 11. Barreto S, Ferreira C, Paixão J, Santos B.S. (2007), Using clustering analysis in a capacitated location-routing problem, *European Journal of Operational Research*, 179(3), 968-977.
 12. Barton, Richard F. (1970), *A Primer on Simulation and Gaming*, 1970, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
 13. Beasley J.E., Chu P.C. (1996), A genetic algorithm for the set covering problem. *European Journal of Operational Research*, 94, 392-404.
 14. Beaumont J. R. (1981), Location-allocation problems in a plane: a review of some models, *Socio-Economic Planning Sciences*, 15(5), 217-229.
 15. Bischoff M., Dächert K. (2009), Allocation search methods for a generalized class of location–allocation problems, *European Journal of Operational Research*, 192(3), 793-807.
 16. Bischoff M., Fleischmann T., Klamroth K. (2006), *The Multi-Facility Location-Allocation Problem with Polyhedral Barriers*, <http://www2.math.uni-wuppertal.de/~klamroth/publications/biflkl06.pdf>.
 17. Beasley J.E., Bull D.R., Martin R.R. (1993), An overview of genetic algorithms: Part I fundamentals. *University Comp* 15, 170–181.

18. Bouhafs L, Hajjam A, Koukam A (2008), A Tabu Search and Ant Colony System Approach for the Capacitated Location-Routing Problem, SNPD 2008, 46-50.
19. Brimberg J, Mladenovic N (1996), Solving the continuous location–allocation problems with tabu search, *Studies in Locational Analysis*, 8, 23–32.
20. Brimberg J., Love R.F. (1994), A location problem with economies of scale, *Studies in Locational Analysis* 7, 9–19.
21. Brimberg J., ReVelle C. (1998). Solving the plant location problem on a line by linear programming, *Top*, 6, 277–286.
22. Brimberg J., Love R.F. (1998), Solving a class of two-dimensional uncapacitated location- allocation problems by dynamic programming, *Operations Research*, 46, 702–709.
23. Brimberg J, Hansen P, Mladenović N, Taillard É (2000), Improvements and comparison of heuristics for solving the multisource Weber problem, *Operations Research*, 48(3), 444–460.
24. Canbolata M.S., von Massow M (2011), Locating emergency facilities with random demand for risk minimization, *Expert Systems with Applications*, Article in Press, Corrected Proof.
25. Chan Felix T.S, Kumar Niraj (2009), Effective allocation of customers to distribution centres: A multiple ant colony optimization approach, *Robotics and computer-integrated manufacturing*, 25, 1-12.
26. Chen P. , Hansen P. , Jaumard B. , Tuy H. (1998), Solution of the Multisource Weber

- and Conditional Weber Problems by D.-C. Programming, Operations Research, 46 (4) , 548–562.
27. Chen R. (1983), Solution of minisum and minimax location-allocation problems with Euclidean distances, Naval Research Logistics Quarterly, 30, 449–459.
 28. Chen J.-F.(2007), A hybrid heuristic for the uncapacitated single allocation hub location problem, Omega, 35(2), 211-220.
 29. Cooper L (1963), Location-Allocation problems, Operations Research, 11(3), 331–343.
 30. Cooper L (1964), Heuristic methods for location-allocation problems, SIAM Rev 6(1):37–53.
 31. Cordeau J-F, Laporte G (2005), Models and tabu search heuristics the berth-allocation problem, 39(4), 526-538.
 32. Cortinhal M. J., Eugenia M (2003), Genetic algorithms for the single source capacitated location problem, Metaheuristics: Computer, Decision-Making, pp. 187-216, Captivo, 2003 Kluwer Academic Publishers B.V.
 33. Crainic T.G., Gendreau M., Soriano P., Toulouse M. (1993), A tabu search procedure for multicommodity location/allocation with balancing requirements, Annals of operations research, 41,359-383.
 34. Doerner Karl F., Gutjahr Walter J., Nolz Pamela C. (2009), Multi-criteria location planning for public facilities in tsunami-prone coastal areas, OR Spectrum, 31(3), 651-678, DOI: 10.1007/s00291-008-0126-7.

35. Dorigo M. (1992), Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italy.
36. Dorigo M., Stützle T. (2004), Ant Colony Optimization, MIT Press, Cambridge, MA.
37. Drezner Z. (1984), The planar two-center and two-median problems, Transportation Science 18, 351–361.
38. Drezner Z, Hamacher HW (2002), Facility location: applications and theory, Campbell JF, Ernst AT, Krishnamoorthy M. Hub location problems, editors, Berlin: Springer, 373–408
39. Eiselt H.A., Laporte G. (1995), Objectives in location problems, In: Drezner, Z., Editor, Facility Location: A Survey of Applications and Methods, Springer-Verlag, Berlin, 151–180.
40. Ernst AT, Krishnamoorthy M, 1999, Solution algorithms for the capacitated single allocation hub location problem, Annals of Operations Research, 86, 141–159.
41. Farahani R.Z., Hekmatfar M (2009), Facility Location: Concepts, Models, Algorithms and Case Studies VIII, 549 p. 73 illus., Hardcover ISBN: 978-3-7908-2150-5.
42. Fazel-Zarandi M.M., Beck, J.C., Solving a Location-Allocation Problem with Logic-Based Benders' Decomposition, Proceedings of the Fifteenth International Conference of Principles and Practice of Constraint Programming (CP09), 344-351, 2009.
43. Fortenberry J.C., Mitra A. (1986), A multiple criteria approach to the location-allocation problem, Computers & Industrial Engineering, 10(1), 77-87.

44. Freek A. (1999), Multi-factors decision analysis via ratio and difference judgment.
45. Gamal M D H, Salhi S (2001), Constructive heuristics for the uncapacitated continuous location–allocation problem, *Journal of the Operational Research Society*, 52:821–9.
46. Glover F., Laguna M.(1997) *Tabu Search*. Kluwer Academic Publishers, Norwell, MA.
47. Glover F. (1989), *Tabu Search - Part I*," *ORSA Journal on Computing*, Vol. 1, No. 3, 190-206, 1989.
48. Glover F, Kochenberger G.A. (2002), *Handbook of metaheuristics*, International series in operations research and management science, 57. Dordrecht: Kluwer Academic Publishers; 321–53.
49. Goldberg, D.E. (1989), *Genetic algorithms in search, optimization and machine Learning*, Kluwer academic publishers, Boston, MA.
50. Gong D, Gen M, Yamazaki G, Xu W (1997), Hybrid evolutionary method for capacitated location-allocation problem, *Computers & Industrial Engineering*, 33(3-4), 577-580.
51. Greasley A. (2008), Using simulation for facility design: A case study, *Simulation Modelling Practice and Theory*, 16(6), 670-677.
52. Hakimi S.L. (1964), Optimal location of switching centers and the absolute centers and medians of a graph, *Operations Research* 12, 450–459.

53. Hansen P., Jaumard B., Taillard E. (1998), Heuristic solution of the multisource Weber problem as a p-median problem, *Operations Research Letters*, 22, 55–62.
54. Henrik J, Robert FL, 1982, Properties and solution methods for large location-allocation problems, *J Oper Res Soc*, 33(5):443–452.
55. Ho W., Carman Ka Man Lee, Ho George To Sum Ho (2008), Optimization of the facility location-allocation problem in a customer-driven supply chain, *Operations Management Research*, 1, 69–79.
56. Holland J. H. (1975), *Adaptation in natural and artificial systems*, University of Michigan press.
57. Houck C R, Joines J A, Kay M G (1996), Comparison of genetic algorithms, random restart and two-opt switching for solving large location–allocation problems. *Computers Operations Research*, 23, 587–96.
58. Hsieh K.H., Tien F.C. (2004), Self-organizing feature maps for solving location–allocation problems with rectilinear distances, *Computers & Operations Research*, 31(7), 1017-1031.
59. Hua Xia-yu, Zheng Jun, Hu Wen-xin (2010), Ant colony optimization algorithm for computing resource allocation based on cloud computing environment, *Journal of East China, Normal University*.
60. Alaya I., Solnon C., Ghedira K. (2007), Ant colony optimization for multi-objective optimization problems, *Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence - 01*, 450-457.

61. Jaramillo J.H., Bhadury J., Batta R. (2002), On the use of genetic algorithms to solve location problems, *Computers and Operations Research*, 29, 761–779.
62. Jones. D. F., Mirrazavi S.K., Tamiz M. (2002), Multi-objective meta-heuristics: An overview of the current state-of-the-art, *European Journal of Operational Research*, 137(1), 1-9.
63. Junjiro Sugimoto, Ryuichi Yokoyama, Takao Shinji (2006), Tabu Search-based Optimal Allocation of Step Voltage Regulators considering the Rate of Operation, *WSEAS Transactions on Power Systems*, 8(1), 1550-1557.
64. Kansou Ali, Yassine Adnan (2010), New upper bounds for the multi-depot capacitated arc routing problem, *International Journal of Metaheuristics*, 1(1), 81 - 95.
65. Klose A., Drexl A. (2005), Facility location models for distribution system design, *European Journal of Operational Research*, 162(1), 4-29.
66. Konaka A., Coit D.W., Smith A.E. (2006), Multi-objective optimization using genetic algorithms: A tutorial, *Reliability Engineering & System Safety*, 91(9),992-1007.
67. Krau S. (1997), Extensions du problème de Weber. Ph.D. Thèse, École Polytechnique de Montréal (under direction of P. Hansen and B. Jaumard).
68. Kuenne R.E., Soland P.M. (1972), Exact and approximate solutions to the multisource weber problem. *Mathematical Programming* 3, 193–209.
69. Kulturel-Konak S., Coit D.W., Baheranwala F. (2008), Pruned Pareto-optimal sets for the system redundancy allocation problem based on multiple prioritized objectives, *Journal of Heuristics*, 14(4), 335-357, DOI: 10.1007/s10732-007-9041-3.

70. Kwang M. S., Weng H. S. (2002) Multiple ant-colony optimization for network routing.. Proceedings of the First International Symposium on Cyber Worlds pp. 277-281, Hosei University, Tokyo.
71. Lin J.-H., Vitter, J.S. (1992), Approximation algorithms for geometric median problems. Information Processing Letters 44, 245–249.
72. Love R F, Morris J G (1975), A computation procedure for the exact solution of location–allocation problems with rectangular distances, Naval Research Logistics Quarterly, 22, 441–53.
73. Love R F, Juel H (1982), Properties and solution methods for large location–allocation problems, Journal of the Operational Research Society, 33, 443–452.
74. Love R.F. (1976). One-dimensional facility location-allocation using dynamic programming, Management Science 22, 614–617.
75. Love R.F., Juel H. (1982), Properties and Solution Methods for Large Location-Allocation Problems, Journal of Operational Research Society, 33, 443-452.
76. Love R F, Morris J G, Wesolowsky G (1988), Facility location: models and methods. In: Publications in operations research, 57, New York: North-Holland.
77. Lozano S, Guerrero F, Onieva L, Larraneta J. (1998), Kononen, maps for solving a class of location–allocation problems, European Journal of Operational Research, 108:106–117.
78. Mark B., John N., Fred L.B. (2004), Creating land allocation zones for forest

- management: a simulated annealing approach, *Canadian Journal of Forest Research.*, 34: 1669–1682.
79. Marler R. Timothy, Arora Jasbir S.(2009), The weighted sum method for multi-objective optimization: new insights, *Structural and Multidisciplinary Optimization*, 41(6), 853-862
80. Matlab, www.mathworks.com
81. Metropolis N., Rosenbluth A., Rosenbluth M., Teller A., Teller E. (1953), Equation of State Calculations by Fast Computing Machines, *Journal of Chemical Physics*, 21(6), 1087-1092.
82. Mitra A., Fortenberry J.C. (1986), The location–allocation problem using multi-objectives, *Engineering Costs and Production Economics*, 10, 113-120, Elsevier Science Publishers.
83. Murray AT, Church RL, (1996), Applying simulated annealing to location-planning models, *Journal of Heuristics*, 2, 31–53.
84. Ninlawan C. (2008), Location decision in distribution centers, *Department of industrial engineering*, 2548, 237.
85. Marín A (2011),The discrete facility location problem with balanced allocation of customers, *European Journal of Operational Research*, 210, 27–38
86. Ohlemueller M (1997), Tabu search for large location-allocation problems, *Journal of the Operational Research Society*, 48(7), 745-750.

87. Ostresh L.M. (1973), SPA: A shortest path algorithm. In: G. Rushton, Editor, Computer Programs for Location-allocation Problems, Department of Geography, The University of Iowa, Iowa City, Iowa Monograph 6.
88. Poole T.G., Szymankiewicz (1977), Using simulation to solve problems, ISBN 0-07-084472-0, 36-112.
89. Qin G. (2006), Logistics Distribution Center Allocation based on Ant Colony Optimization, Systems Engineering-theory and Practice, (4): 120-124.
90. ReVelle C., Marks D., Liebman, J. (1970), Analysis of private and public sector location problems, Management Science 16, 692–707.
91. ReVelle C.S., Eiselt H.A. (2005), Location analysis: A synthesis and survey, European Journal of Operational Research 165, 1–19.
92. ReVelle C.S., Eiselt H.A., Daskin M.S. (2008), A bibliography for some fundamental problem categories in discrete location science, European Journal of Operational Research, 184, 817–848.
93. Rosing K.E., ReVelle C.S., Rosing-Vogelaar H. (1992), Fractions in the location set-covering problem. Environment and Planning, B 19, 125–130.
94. Satani N., Uchida A., Deguchi A., Ohgai A., Sato S., Hagishima S. (1998), Commercial facility location model using multiple regression analysis, Computers, Environment and Urban Systems, 22(3), 219-240.

95. Sawaragi. Y., Nakayama, H., and Tanino, T., 1985, Theory of Multiobjective Optimization, 176, Mathematics in Science and Engineering, Orlando, FL: Academic Press Inc, ISBN 0126203709.
96. Scaparra M P, Scutell`a M G (2001), Facilities, locations, customers: Building blocks of location models: A survey, Technical report TR-01–18, Computer Science Department, University of Pisa, Italy.
97. Shannon Robert E. (1975), System simulation: The Art and Science, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, ISBN 0-13-881839-8, p2.
98. Silva C.A., Sousa J.M.C., Runkler T.A. (2008), Rescheduling and optimization of logistic processes using GA and ACO, Engineering Applications of Artificial Intelligence, 21(3), 343-352.
99. Silva M.R., Cunha C.B.(2009), New simple and efficient heuristics for the uncapacitated single allocation hub location problem, Computers & Operations Research, 36(12), 3152-3165.
100. Steuer, R.E. (1986), Multiple Criteria Optimization: Theory, Computations, and Application, New York, John Wiley & Sons, Inc, ISBN 047188846X.
101. Talbi E-G. (2009), Metaheuristics: from design to implementation, Wiley, 624 p, ISBN: 978-0-470-27858-1.
102. Toregas, C., Swain, R., ReVelle, C., Bergman, L. (1971), The Location of emergency service facilities, Operations Research, 19, 1363–1373.

103. Tsuchiya K., Bharitkar S., Takefuji Y (1996), A neural network approach to facility layout problems, *European Journal of Operational Research*, 89(3), 556-563.
104. Vecihi Y., Aydin M., Turkbey O (2006), Solving large-scale uncapacitated facility location problems with evolutionary simulated annealing, *International journal of production research*, 44(22), 4773–4791.
105. Villegas J.G, Palacios F., Medaglia A. L. (2006), Solution methods for the bi-objective (cost-coverage) unconstrained facility location problem with an illustrative example, *Annals of operations research*, 109-141, DOI: 10.1007/s10479-006-0061-4
106. Vos B., Akkermans H. (1996), Capturing the dynamics of facility allocation. *International Journal of Operations & Production Management* 16(11), 57–70
107. Weber A. (1929), *Location of Industries*, The University of Chicago Press, translated by Carl J. Friedrich from Weber's 1909 book.
108. Zhou G., Min H., Gen M. (2002), The balanced allocation of customers to multiple distribution centers in the supply chain network: a genetic algorithm approach, *Computers and Industrial Engineering archive*, 43(1-2), 251 – 261.
109. Zhou G., Min H., Gen M. (2003), A genetic algorithm approach to the bi-criteria allocation of customers to warehouses, *International Journal of Production Economics*, 86, 35–45.