INDIAN INSTITUTE OF MANAGEMENT CALCUTTA

WORKING PAPER SERIES

WPS No. 734/ October 2013

Metaheuristic Applications on Discrete Facility Location Problems: A Survey

by

Sumanta Basu
Assistant Professor, IIM Calcutta, D. H. Road, Joka P.O., Kolkata 700 104 India

Megha Sharma
Assistant Professor, IIM Calcutta, D. H. Road, Joka P.O., Kolkata 700 104 India

&

Partha Sarathi Ghosh
Wipro Technologies

METAHEURISTIC APPLICATIONS ON DISCRETE FACILITY LOCATION PROBLEMS: A SURVEY

Sumanta Basu* Megha Sharma† Partha Sarathi Ghosh‡

Abstract

This paper provides a detailed review of metaheuristic applications on discrete facility location problems. The objective of this paper is to provide a concise summary of solution approaches based on four commonly used metaheuristics: genetic algorithm, tabu search, particle swarm optimization and scatter search for different variants of the discrete facility location problem. Such a concise summary is expected to be useful for researchers interested in any of the major variants of discrete facility location problem as for each metaheuristic the paper provides a comprehensive review of different variants on which this metaheuristic has been applied, and the details of its implementation. Therefore, a research can exploit a method developed for another variant to solve the problem variant at hand. Based on our review of these papers, we also report some interesting observations, identify research gaps and highlight directions for future research.

Facility Location, Genetic Algorithm, Tabu Search, Particle Swarm Optimization, Scatter Search

1 Introduction

Discrete facility location problems are one of the most extensively studied combinatorial optimization problems. Different variants of this problem have been used to model a wide variety of real life problems. For example, real life plant location problems have been modeled as uncapacitated or capacitated facility location problems, p-center problems have been used to determine the location of emergency services such as ambulances, medical services, and police vans etc., and the problem of locating distribution centers to serve customers within a limited service time has been modeled as the maximal covering location problems. Other variants such as the p-median problems and the p-dispersion problems have also been applied extensively in real life. A detailed review of the different variants of the discrete facility location problems and their applications can be found in (Drezner, 1995).

Different solution approaches have been proposed and used for different variants. Since almost all variants of the discrete facility location problems are NP-hard, exact approaches are limited. Hence most real life discrete facility location problems are solved through heuristics and metaheuristics. While heuristic solution approaches are more problem specific, metaheuristics have a problem independent structure consisting of components which exploit problem related information. Review of different exact and heuristic solutions approaches categorized by problem variant can be found in (Drezner, 1995; Drezner and Hamacher, 2004). While there has been a significant amount of research on metaheuristic based solution approaches in the last one decade, and many of these approaches have been found to outperform problem specific heuristics, to the best of our knowledge no comprehensive review of the literature on these approaches is available. Although similar literature survey is available for p-median problems (Mladenovic et al., 2007), it is restricted within a specific context. This paper aims to fill in this gap by providing a comprehensive review of the literature on metaheuristic applications in solving discrete facility location problems.

In particular, we review the published applications of four of the most popular and extensively researched metaheuristics, namely genetic algorithms, tabu search, particle swarm optimization, and scatter search in the field of discrete facility location problems. While most of the review papers on solution approaches to a problem class, present the survey by each problem class, in this paper we adopt a different approach and provide the survey by solution approaches i.e. metaheuristics and within each metaheuristic we subcategorize the literature by problem class it refers to. There are four major reasons behind our approach as given below.

Firstly, since metaheuristics consists of different components, each of which can be implemented in one of many different ways for solving a problem. As the rationale of choosing one implementation of a component over others is not as objective as with any analytical or exact solution approach, many a times, an implementation

*Operations Management Group, Indian Institute of Management Calcutta. Email: sumanta@iimcal.ac.in
†Operations Management Group, Indian Institute of Management Calcutta. Email: megha@iimcal.ac.in
‡Wipro Technologies, Email:partha.silicon@gmail.com

is chosen either due to the ease or convenience of implementing it. As the rationale is very subjective and there are many implementations possible, most papers while presenting a solution approach do not compare the performance of different implementations. Due to this, most often when new solution approaches using that metaheuristic are presented, they choose the implementations already reported in the literature. As a result of this, one can note that some particular implementations are used in the literature again and again, while some other implementations which have given good results in related problem variants are not explored. Therefore, in this paper we organize our survey by metaheuristics and not problem classes to emphasize on the implementations that have been used in the literature on different problem variants. This would help researchers working on a particular problem variant select from implementations that may not have been implemented on this variant but have worked well on other variants. This would enable deeper exploitation of the benefits that each of these metaheuristics offer.

Secondly, most of the current research in the domain of discrete facility location focuses on extensions of the basic facility location models such as facility location with vehicle routing (Nagy and Salhi, 2007), facility location with inventory management (Shen et al., 2003), facility location with stochastic demand or transportation costs (Snyder, 2006), reliable facility location (Shen, 2011) etc. Here again, the choice of a metaheuristic over another for solving these extended problems is often arbitrary, and within a metaheuristic the choice of an implementation of a component is generally based on that used for solving the basic variant in the literature. Sometimes, different implementations from other problem variants are also explored. As the choice of a metaheuristic is based on its success as reported in the literature, classifying our survey based on metaheuristics seems to be an obvious choice.

Thirdly, for a researcher new to the area of discrete facility location our work provides a complete overview of the basic variants of the discrete facility location problems, four of the most commonly used metaheuristics, and reports applications of these metaheuristics to different problem variants. Our scheme clearly indicates the less or unexplored areas within each of the four metaheuristics for solving the facility location problem. It will help researchers identify research gaps in terms of metaheuristics based solution approaches for discrete facility location problems. Moreover, as we present four of the most popular metaheuristics, our survey will also help researchers develop new hybrid metaheuristics for solving different problem variants. One may note that recently hybrid metaheuristics have emerged as the solution method of choice not only for facility location problems (Resende and Werneck, 2004, 2006) but also for different combinatorial optimization problems such as the travelling salesman problem (Tsai et al., 2004) etc.

Lastly, as a problem specific metaheuristic implementation is likely to perform better, many problem specific metaheuristics have been proposed in the literature. Unfortunately, most of these implementations never get used in practice due to assumptions inherent in the model. To overcome, this research-practice divide researchers have started focusing on generalized metaheuristics which are problem independent (see for example, Campos et al. (2005)). These generalized metaheuristics do not always perform as good as a problem specific metaheuristic, overall on a problem class they perform acceptably well. To the best of our knowledge, there is no such generalized metaheuristic for discrete facility location problems. This review by presenting different problem variants and different metaheuristic based solution approached provides a stepping stone for such a generalized metaheuristic.

Moreover, we also provide a list of benchmark problems used by different papers surveyed in this review. As we note different papers on the same problem variant use different benchmark problems to quantify the performance of their solution approach. This makes comparison between different solution approaches difficult. By providing a list of commonly used benchmark problems, we hope future research in the area would use common benchmark problems, thus making the comparison between different approaches easier.

The paper is organized as follows. In Section 2, we briefly describe the problem variants that we review in this paper. Section 2 also provides an overview of different problem types segregated by size of the problem addressed in published literature. A complete list of all benchmark problems segregated by major problem categories is provided to facilitate comparative research in this area. Section 3.2 elaborates on application and implementation issues of genetic algorithm on various discrete location problems. In section 3.1, we provide details of tabu search implementation. A short discussion on Particle swarm optimization is found in section 3.3. Scatter search components along with location problems addressed are summarized in Section 3.4. We conclude the paper in Section 4 by highlighting some important observations and major open issues in this area of research.

2 Discrete Facility Location: Basic Problem Variants

Facility location problems are problems of choosing a subset of locations from a given set of potential locations to establish facilities so as to optimize a given function of these chosen locations while satisfying certain constraints. Based on the set of potential locations, facility location problems can be classified into discrete facility location problems and continuous facility location problems depending upon whether the set of potential locations is discrete or continuous. In discrete facility location problems an exhaustive list of potential locations for facilities is available, while in case of continuous facility location a continuous space as usually specified by its co-ordinates is provided. The treatment of problems from these two classes differs significantly and their solution approaches are also significantly different. Discrete facility location problems, which are the focus of this work, are combinatorial optimization problems. Discrete facility location problems have attracted more focus in published literature in recent years because of their wider application areas (Mirchandani and R.L., 1990). A detailed list of real life problems modelled as discrete facility location problems can be found in the book by Drezner (Drezner, 1995). In this section, we briefly describe the major variants of the discrete facility location problems.

In all the problem variants, one is given a set D of demand points with the demand d_i for each demand point $i \in D$, a set F of potential locations for facilities, the cost c_{ij} of transporting one unit of supply from facility $j \in F$ to demand point $i \in D$ for all possible facility - demand point combinations. Different problem variants differ in terms of other characteristics such as fixed cost f_j of opening a facility at potential location $j \in F$, the capacity K_j of a facility if opened at potential location $j \in F$ etc. provided to the decision maker. Problem variants also differ in terms of their objective function and the constraints they impose on facilities etc. Given this basic nomenclature, we define the major variants of the problem as follows.

Uncapacitated Facility Location Problem (UFLP): Given a set D of demand points with the demand d_i for each demand point $i \in D$, a set F of potential locations for facilities, fixed cost f_j of opening a facility at potential location $j \in F$, the cost c_{ij} of transporting one unit of supply from facility $j \in F$ to demand point $i \in D$ for all possible facility - demand point combinations, the UFLP is the problem of choosing a subset $F' \subseteq F$ to open the facilities at and assigning each demand point to at least one open facility, such that the sum of the fixed costs of opening these facilities and the total transportation cost of supplying d_i units of demand at each demand point $i \in D$ from its assigned facility in F' is minimized. UFLP is also referred to as the Simple Plant Location (SPL) Problem. Since there are no capacity constraints, each demand point is assigned to its closest opened facility, and the problem reduces to that of simply choosing the locations to open the facilities at. UFLP has been extensively used in practice when one is free to choose the plant capacity as well. Applications of SPL are elaborated in papers like (Melo et al., 2009; Verter, 2011).

p-Median Problem (PMP): Given a set D of demand points with the demand d_i for each demand point $i \in D$, a set F of potential locations for facilities, the cost c_{ij} of transporting one unit of supply from facility $j \in F$ to demand point $i \in D$ for all possible facility - demand point combinations, and a number p, the PMP is the problem of choosing a subset $F' \subseteq F$, where $|F'| = p$, to open the facilities at and assigning each demand point to an opened facility in F', such that the total transportation cost of supplying d_i units of demand at each demand point $i \in D$ from facilities in F' is minimized. Since there are no capacity constraints, each demand point is assigned to its closest opened facility, and the problem reduces to that of simply choosing the locations to open the facilities at. PMP is used to model those situations wherein the number of facilities to be opened is pre-specified. Applications of PMP can be found in (Tansel et al., 1983).

p-Center Problem (PCP): Given a set D of demand points, a set F of potential locations for facilities, the cost c_{ij} of transporting one unit of supply from facility $j \in F$ to demand point $i \in D$ for all possible facility - demand point combinations, the PCP is the problem of choosing a subset $F' \subseteq F$, where $|F'| = p$, to open the facilities at and assigning each demand point to an opened facility in $|F'|$, such that the maximum cost among all the demand points and their assigned facilities is minimized. Applications of PCP include location decisions for emergency facilities such as ambulances, police vans etc. For more applications of p-median problems, interested readers are advised to see papers like (Tansel et al., 1983; Huang et al., 2010).

Capacitated Facility Location Problem (CFLP): Given a set D of demand points with the demand d_i for each demand point $i \in D$, a set F of potential locations for facilities, fixed cost f_j of opening a facility at potential location $j \in F$, the capacity k_j of each facility if opened at potential location $j \in F$, the cost c_{ij} of transporting one unit of supply from facility $j \in F$ to demand point $i \in D$ for all possible facility - demand point combinations, the CFLP is the problem of choosing a subset $F' \subseteq F$ to open the facilities at and assigning demand points to opened facilities without violating capacity constraints for the facilities, such that the sum of the fixed costs of opening these facilities and the total transportation cost of supplying d_i units of demand at each demand point $i \in D$ from facilities in F' is minimized. CFLP is generally used while choosing suppliers from existing set of suppliers or in plant location when there are constraints on their maximum capacities.

Maximal Covering Location Problem (MCLP): Introduced by Church and Revelle (1974), the problem context is defined with a set D of demand points with the demand d_i for each demand point $i \in D$, a set

F of potential locations for facilities, the cost c_{ij} of transporting one unit of supply from facility $j \in F$ to demand point $i \in D$ for all possible facility - demand point combinations. Objective of this problem is to choose a subset $F' \subseteq F$ to open the facilities at and assigning demand points within a service distance S to opened facilities. Applications of MCLP are summarized in a paper by Chung (1986).

Hub Location Problem (HLP): While all the other problem variants described above are single stage, i.e. plants supply directly to the demand points, HLP is a two stage facility location problem wherein there is a set of facilities that supply to a set of distribution centers which in turn supply to a set of demand points. Papers like (Contreras et al., 2011; Kim and O'Kelly, 2009) elaborate on some application areas of hub location problem.

While there are some other variants such as the p-Dispersion Problem, Obnoxious Facility Location Problem, etc., in this paper we restrict our attention to the variants described above. We chose these categories for two reasons. Firstly, these models are the most basic and most extensively studied models of facility location. Secondly, these models are the basis of many advanced models such as those including the inventory decisions, vehicle routing decisions, or those including the reliability of facilities etc. In this paper, we review metaheuristic based solution approaches for these categories, in particular, we survey papers that report solution approaches based on Genetic Algorithms, Tabu Search, Particle Swarm Optimization, and Scatter Search. While the choice of genetic algorithms and tabu search is motivated by the number of applications reported in the literature, scatter search has been chosen because it has given many promising results in recent years with its more structured methodology. Similarly, particle swarm optimization has been selected despite the limited number of applications reported in the literature, because of its growing popularity and success reported in other problem domains.

In the next section, we briefly describe each of the four metaheuristics and present their basic structure. Within each metaheuristic, we classify the applications reported in the literature by the problem type they attempt to solve. Based on these applications on different problem variants we then summarize our observations and identify scope for future research. Before describing the solution approaches in Table 1 we provide a bibliography for each of the problem variant categorized by the problem size on which those solution approaches are tested. While the size of a problem instance can be specified in terms of many problem parameters, such as the number of potential locations for facilities, number of commodities etc., for the sake of comparison among different problem variants, we use the number of demand points to denote the problem size for two reasons, a) it has the largest contribution to define problem size and number of facilities increases proportionately with number of demand nodes in most of the papers and b) it gives a precise and simpler picture of problem size distribution across surveyed papers. It is important to note that in most of the benchmark problems, the set of demand points also serves as the set of potential locations for facilities. In most of the papers, authors attempt problems with varying sizes to see the effectiveness of their algorithms of which we chose the maximum size (in terms of number of demand points) to report.

As can be observed from the table, more research has been done on the simple plant location problems which includes the UFLP and the CFLP, and the p-median problem, with more than 50% of the papers focusing on these problems. One can also note that the testbed problem instances for these variants are quite evenly distributed within the entire range compared to other variants which have mainly focused on smaller instances. One exception to this is the MCLP, research on which has either focused on smaller instances with 50 to 500 demand points or larger instances with 1000 to 2500 demand points. While papers with tabu search implementation addressed smaller problem instances in MCLP, GA based heuristics are successful to address larger problem instances. Typically the problem sizes considered for hub location problems are lower because of its complex network structure with multiple stages. While 10 papers out of the 45 listed in the table focus on testbed problem instances between 1000 and 2500, a majority of these deal with problem instances with less than 1500 demand points.

As mentioned in Section 1, while there has been a lot of research in terms of metaheuristic applications for discrete facility location problems, different papers use different benchmark problems or testbeds to quantify the performance of their proposed solution approach. Usage of different benchmark problems makes comparison between different solution approaches difficult. Therefore, before providing a comprehensive review of these metaheuristic applications, in Table 2 we present a list of the benchmark problems used by the papers we surveyed for a ready reference for future researchers to compare their results against standard problem types. The first column in each row refers to the paper that we reviewed and which used some benchmark problem, and the subsequent columns give the reference to the paper(s), where the benchmark problem(s) used in the paper was first introduced, categorized by the problem variant.

It is evident from the table that some benchmark problems have been used by multiple papers. For example, for p-median problem, problem instances used by Beasley (Beasley, 1988, 1990, 1996) are considered for benchmarking in many of the reviewed papers. Similarly, for MCLP problem instances available in papers by Daskin (Daskin, 1983, 1995) are used and for HLP, problem instances designed by O'Kelly (based on airlines

Problem Size	SPL	PMP	PCP	MCLP	HL	Total
≤ 50	(Du and Evans, 2008; Vaithyanathan et al., 1996; Yapicioglu et al., 2007)	(Correa et al., 2004)			(Calik et al., 2009; Topcuoglu et al., 2005; Abdinnour-Helm, 1998; Cunha and Silva, 2007)	8
$(50, 500]$	(Al-Sultan and Al-Fawzan, 1999; Delmaire et al., 1999; Ohlemuller, 1997; Arostegui et al., 2006; Lai et al., 2010; Contreras and Diaz, 2008),	(Chaudhry et al., 2003; Chi et al., 2011; Scheuerer and Wendolsky, 2006; Rolland et al., 1996)	(Pacheco and Casado, 2005; Mozos and Mesa, 2000)	(Jia et al., 2007; Lee and Lee, 2010; Adenso-Diaz and Rodriguez, 1997)	(Topcuoglu et al., 2005; Keskin and Uster, 2007b,a; Chen, 2007; Silva and Cunha, 2009)	20
$(500, 1000]$	(Ghosh, 2003; Guner and Sevkil, 2008; Wang et al., 2008)	(Stanimirovic et al., 2007; Alp and Erkut, 2003; Diaz and Fernandez, 2006),	(Jaramillo et al., 2002)			7
$(1000, 2500]$	(Salhi, 2003; Kratica et al., 2001; Michel and Hentenryck, 2004; Resende and Werneck, 2006; Sun, 2006) ()	(Garcia-Lopez et al., 2003; Salhi, 2002)		(Aytug and Saydam, 2002; Krzanowski and Raper, 1999; Jia et al., 2007)		10
Total	17	10	3	6	9	45

Table 1: Distribution of problem size in the literature

passenger flow between 25 US cities as evaluated by Civil Aeronautics Board (CAB)) (O'Kelly, 1987) dominates other test problems.

3 The Metaheuristics

All the metaheuristics surveyed in this paper except tabu search are population based metaheuristics. All of these metaheuristics start with a set of initial solutions (a singleton in case of tabu search), generated either randomly or by exploiting some information about the problem at hand. This set of solutions is iteratively improved by applying a set of operations on the solutions in the set, such as combining two solutions or searching for a better neighbour of a solution in the set etc. for a pre-specified number of iterations.

A typical solution to a discrete facility location problem consists of the locations where facilities are to be opened and an assignment of demand points to facilities. For uncapacitated variants of the problem, such as the UFLP, PMP, PCP etc. given the locations of the facilities this assignment is trivial as each demand point is assigned to its nearest opened facility. Assignment of demand points to facilities is not as straightforward in case of capacitated problems such as the CFLP. For capacitated variants, the problem context can impose an additional constraint that each demand point must be assigned to exactly one facility. Problem variants with this additional constraint are referred to as single-source problems. When no such constraint is imposed the problems are referred to as multi-source problems. For single-source problems, a modified assignment problem is solved to find the optimal allocation while solving a transportation problem yields the optimal assignment of demand points to facilities in case of multi-source problems.

We now describe each of the four metaheuristics and their applications in solving discrete facility location problems. For each metaheuristic, we first explain its basic principle and describe its major components. We then categorize the papers reviewed based on the problem variant(s) they address and the maximum problem size on which they test their solution approach. We also summarize the details of the problem studied along six major aspects of the problem: problem type (the problem variant it belongs to), maximum problem size addressed in the paper, whether facility capacity is considered or not, for capacitated variants whether single-source or multi-source, whether single-commodity or multi-commodity, and whether single stage or multi stage. We then review these papers on each component of the metaheuristic, thus identifying possible areas for future research.

3.1 Tabu search

Tabu search is one of the widely used local search based improvement metaheuristics which continues to get superior results in discrete location problems. It starts with an initial solution to the problem, calls it the

Paper Reference	**MCLP**	**SPL**	**PMP**	**HL**
(Aytug and Saydam, 2002)	(Daskin, 1983)			
(Chaudhry et al., 2003)			(Toregas et al., 1971; Choi and Chaudhry, 1993)	
(Cunha and Silva, 2007)				(O'Kelly, 1987)
(Jaramillo et al., 2002)	(Daskin, 1995)	(Beasley, 1993, 1996)		
(Kratica et al., 2001)			(Beasley, 1996)	
(Lai et al., 2010)		(Beasley, 1988)		
(Resende and Werneck, 2006)		(Hoefer, 2003, 2002)		
(Stanimirovic et al., 2007)			(Beasley, 1996)	
(Topcuoglu et al., 2005)				(O'Kelly, 1987; Ernst and Krishnamoorthy, 1996)
(Alp and Erkut, 2003)			(Beasley, 1990; Koerkel, 1989)	
(Abdinnour-Helm, 1998)				(O'Kelly, 1987)
(Al-Sultan and Al-Fawzan, 1999)		(Beasley, 1990; Karg and Thompson, 1964)		
(Calik et al., 2009)				(O'Kelly, 1987; Tan and Kara, 2007)
(Ghosh, 2003)		(Koerkel, 1989)		
(Michel and Hentenryck, 2004)		(Kratica et al., 2001)		
(Silva and Cunha, 2009)				(O'Kelly, 1987; Ernst and Krishnamoorthy, 1996, 1998)
(Sun, 2006)		(Beasley, 1990; Kratica et al., 2001; Kochetov, 2003; Hoefer, 2004)		
(Garcia-Lopez et al., 2003)			(Reinelt, 1991)	
(Contreras and Diaz, 2008)		(Ahuja et al., 2004; Holmberg et al., 1999; Delmaire et al., 1999)		
(Diaz and Fernandez, 2006)			(Beasley, 1990; Beasley and Chu, 1996; Lorena and Senne, 2003, 2004)	
(Ohlemuller, 1997)		(Liu et al., 1982)		
(Chen, 2007)				(O'Kelly, 1987; Ernst and Krishnamoorthy, 1996)
(Mladenovic et al., 2007)			(Koerkel, 1989; Beasley, 1990; Rolland et al., 1996; Resende and Werneck, 2004; Kochetov, 2003)	

Table 2: List of benchmark problems

current solution, and searches for the best solution in a suitably defined *neighborhood* of the solution. It then designates the best solution in the neighborhood as the current solution and starts the search process again. Tabu search terminates when certain terminating conditions, either involving execution time, or solution quality objectives, or both, have been met. Throughout the search process, it keeps track of the current best solution and the best solution found so far. In order to prevent tabu search from considering solutions that it has visited in recent iterations, tabu search maintains a list of neighbor generation moves it considers forbidden, or *tabu* (hence the name, tabu search) and removes solutions that can be reached only through tabu moves from the neighborhood. Once a move enters the list of tabu moves, it stays there for some number of successive iterations (called the tabu tenure of the move). The list of tabu moves changes continuously during the execution of the search, making tabu search an adaptive memory search algorithm. Several researchers have added features to enrich the basic tabu search algorithm described here, implementing features like *intermediate term memory function* for intensification, *long term memory function* for search diversification and *aspiration criteria* to overrule tabu status. Other features that have been proposed, but not commonly implemented for tabu search are *strategic oscillation*, *path relinking*, *candidate list strategies* etc. (Glover, 1989). More details about tabu search and its components can be found in initial paper by Glover (Glover, 1989).

We present a list of tabu search applications on discrete facility location problems categorized by the problem variant and problem size in Table 3. As is evident from the table, more researchers have focused on simple plant location and hub location problems, followed by the p-median problem. Out of the 20 papers reviewed,

Problem Size	SPL	PMP	PCP	MCLP	HLP
≤ 50	(Vaithyanathan et al., 1996)				(Calik et al., 2009; Abdinnour-Helm, 1998)
$(50, 500]$	(Al-Sultan and Al-Fawzan, 1999; Delmaire et al., 1999; Ohlemuller, 1997; Arostegui et al., 2006)	(Rolland et al., 1996)	(Mozos and Mesa, 2000)	(Lee and Lee, 2010; Adenso-Diaz and Rodriguez, 1997)	(Chen, 2007; Silva and Cunha, 2009)
$(500, 1000]$	(Ghosh, 2003)				
$(1000, 2500]$	(Michel and Hentenryck, 2004; Resende and Werneck, 2006; Sun, 2006)	(Salhi, 2002)			

Table 3: Problem sizes addressed in papers with TS implementation

six papers consider a problem size of less than 100 demand points with majority of them addressing the hub location problem (Abdinnour-Helm, 1998; Calik et al., 2009; Topcuoglu et al., 2005). We could find only four papers (with no paper in MCLP) addressing a problem size of more than 1000 demand points.

3.1.1 Tabu search components

We now provide details of the reviewed applications of tabu search categorized by the tabu search components. **Initial solution:** As with any local search based heuristic, the initial solution plays a vital role in setting up the initial search space for tabu search as well. Some of the most commonly used strategies for generating initial solution for tabu search include randomly generating initial solution, greedy approach and initial solution based on some problem characteristics. In our case, we found that most papers use randomly generated initial solutions (Al-Sultan and Al-Fawzan, 1999; Prez et al., 2003; Rolland et al., 1996; Sun, 2006; Ohlemuller, 1997; Arostegui et al., 2006; Mozos and Mesa, 2000; Michel and Hentenryck, 2004; Resende and Werneck, 2006). Papers like (Silva and Cunha, 2009; Delmaire et al., 1999; Lee and Lee, 2010; Chen, 2007; Ghosh, 2003) discuss strategies based on a greedy approach to generate initial solutions while three papers (Calik et al., 2009; O'Kelly, 1987; Salhi, 2002), used different heuristic techniques to generate initial solutions for the HLP. The paper by Abdinnour-Helm (Abdinnour-Helm, 1998) elaborated the usage of other meta-heuristics, in particular genetic algorithm, to generate initial solution in the context of UFLP. To address a real-life MCLP, authors in a paper (Adenso-Diaz and Rodriguez, 1997) developed problem specific methods to assign ambulances for the counties as an initial solution for tabu search.

Moves: Tabu search being an improvement heuristic moves from one solution to the next in search of an optimal solution. The method of moving from one solution to another is described by a set of rules and is called a move. The set of all solutions that can be reached from a given solution using a pre-specified move is called the neighborhood of the solution. Mostly two types of moves are frequently used in literature: 1-interchange and 2-swap. The ways by which these two moves are executed in papers depend on the problem context. For example, in the context of the CFLP 1-interchange move is defined as changing the allocation of one demand point from one facility to another, while in the context of HLP, it is defined as changing the status of one non-hub node to a hub node. 12 papers (Mozos and Mesa, 2000; Ohlemuller, 1997; Prez et al., 2003; Rolland et al., 1996; Silva and Cunha, 2009; Sun, 2006; Al-Sultan and Al-Fawzan, 1999; Arostegui et al., 2006; Calik et al., 2009; Michel and Hentenryck, 2004; Delmaire et al., 1999; Chen, 2007) out of the 18 papers reviewed used 1-interchange move. Similarly, 2-swap move can be defined as closing an existing facility and opening up a new facility or swapping the assigned facilities for two demand points. 8 papers (Abdinnour-Helm, 1998; Ghosh, 2003; Lee and Lee, 2010; Rolland et al., 1996; Salhi, 2002; Prez et al., 2003; Delmaire et al., 1999; Adenso-Diaz and Rodriguez, 1997) used some variant of the 2-swap move. Our observations are summarized in Table 4 for quick reference.

Tabu tenure: Tabu tenures are used as short term memory functions in tabu search to prevent the search process from re-visiting solutions that it has visited in the past few iterations. Tabu search implementations either keep the tenure value as constant through the process or assign it a value that changes deterministically as a function of algorithm parameters (such as the number of iterations already executed), or problem parameters (such as the problem size), or assign it a value generated randomly within a pre-specified range. We refer to the first two kinds of tabu tenures as deterministic tabu tenure, and the third one as random tabu tenure. A detailed investigation on the impact of tabu tenure to the solution quality obtained can be found in a paper by

Move	SPL	PMP	PCP	MCLP	HLP
1-*interchange*	(Ohlemuller, 1997; Sun, 2006; Al-Sultan and Al-Fawzan, 1999; Arostegui et al., 2006; Delmaire et al., 1999; Michel and Hentenryck, 2004; Resende and Werneck, 2006)	(Rolland et al., 1996; Salhi, 2002)	(Mozos and Mesa, 2000)		(Silva and Cunha, 2009; Calik et al., 2009; Chen, 2007)
2-*swap*	(Ghosh, 2003; Delmaire et al., 1999; Abdinnour-Helm, 1998)	(Rolland et al., 1996; Salhi, 2002)		(Lee and Lee, 2010; Adenso-Diaz and Rodriguez, 1997)	(Abdinnour-Helm, 1998)

Table 4: Moves used in papers with TS implementation

Salhi (Salhi, 2002).

Out of the papers reviewed, we found 10 papers implementing fixed tabu tenure and 5 papers implementing random tabu tenure. Table 3.1.1 summarizes our findings with regard to the types of tenures used in surveyed papers.

Tenure Type	SPL	PMP	PCP	MCLP	HLP
Deterministic Tabu Tenure					
Constant	(Sun, 2006)		(Mozos and Mesa, 2000)		(Abdinnour-Helm, 1998; Calik et al., 2009; Chen, 2007)
Deterministic Variable	(Michel and Hentenryck, 2004; Ohlemuller, 1997)	(Salhi, 2002)		(Lee and Lee, 2010)	(Silva and Cunha, 2009)
Random Tabu Tenure					
Fixed Range	(Al-Sultan and Al-Fawzan, 1999; Arostegui et al., 2006; Delmaire et al., 1999)			(Adenso-Diaz and Rodriguez, 1997)	
Dependent Range		(Rolland et al., 1996)			

Table 5: Usage of Tabu Tenure in papers with TS implementation

Intermediate term memory function: Intermediate term memory functions are used in tabu search to intensify the search process by restricting it to promising regions of the solution space. Apart from the paper by Sun (Sun, 2006), we could not find any evidence of any other paper explicitly using intermediate term memory function.

Long term memory function: Long term memory functions are used to diversify the search to new regions in the solution space. In the survey, we found three distinct strategies being used to diversify the search process: restarting the search process from initial solutions different from current solution (Arostegui et al., 2006; Prez et al., 2003; Silva and Cunha, 2009; Delmaire et al., 1999), frequency based penalty to discourage locations involved in recent moves (Ghosh, 2003; Rolland et al., 1996; Sun, 2006) and change of status in some randomly selected demand points (Michel and Hentenryck, 2004; Ohlemuller, 1997; Delmaire et al., 1999) for perturbation in current solution. Most of the papers with diversification strategy consider SPL as their problem contexts.

Aspiration criteria: Aspiration criteria are a set of conditions which if satisfied, permits tabu search to make use of tabu moves to reach neighboring solutions. All the papers in which this component of tabu search was implemented, tabu status of a particular solution was withdrawn if the solution quality found is the best so far. In our survey, seven of the papers (e.g. (Ghosh, 2003; Mozos and Mesa, 2000; Rolland et al., 1996; Silva and Cunha, 2009; Sun, 2006; Delmaire et al., 1999; Salhi, 2002)) used aspiration criteria. Among these papers, one paper by Salhi (Salhi, 2002) elaborates on different aspiration mechanisms and their implications on solution quality.

Strategic oscillation: This component of tabu search allows the search process to go to infeasible solution

Strategies	SPL	PMP	PCP	MCLP	HLP
Restart with different initial solution	(Arostegui et al., 2006; Delmaire et al., 1999; Resende and Werneck, 2006)				(Silva and Cunha, 2009)
Frequency based penalty	(Sun, 2006; Ghosh, 2003)	(Rolland et al., 1996)			
Change in status	(Michel and Hentenryck, 2004; Ohlemuller, 1997; Delmaire et al., 1999)				

Table 6: Long term memory functions in papers with TS implementation

space for some iterations in the hope of reaching a better feasible solution. It is primarily used to move out from a search region and can be an alternative of diversification strategy in tabu search. We identified four papers implementing strategic oscillation as mentioned in Table 7. In the paper by Calik et al. (Calik et al., 2009), if no feasible solution is found in 100 neighboring solutions, an infeasible neighbor is chosen for the next iteration. In (Rolland et al., 1996), authors allowed infeasible moves with a certain tolerance. As in (Rolland et al., 1996) for a p-median problem, a slack number is introduced to change the number of facilities from p to $p \pm slack$. In case of capacitated plant location problem (Delmaire et al., 1999), capacities for some opened facilities were relaxed to reach to a new solution region.

Termination criteria: Typical termination criteria used in tabu search are as follows: maximum number of iterations (Abdinnour-Helm, 1998; Al-Sultan and Al-Fawzan, 1999; Calik et al., 2009; Rolland et al., 1996; Salhi, 2002; Adenso-Diaz and Rodriguez, 1997; Lee and Lee, 2010; Silva and Cunha, 2009), maximum number of iterations without any improvement in solution value (Michel and Hentenryck, 2004; Salhi, 2002; Delmaire et al., 1999; Arostegui et al., 2006; Resende and Werneck, 2006) or maximum time limit (Arostegui et al., 2006; Calik et al., 2009; Ghosh, 2003). In some papers the termination criteria are dependent on the problem size, e.g. maximum iteration limit is a function of the problem size in papers like (Al-Sultan and Al-Fawzan, 1999; Mozos and Mesa, 2000; Rolland et al., 1996). In (Sun, 2006), author stopped the search process based on number of time the diversification process had been invoked.

Combination with other metaheuristic: There have been a number of attempts to create hybrid metaheuristics in combination with tabu search. In (Abdinnour-Helm, 1998; Topcuoglu et al., 2005), authors used genetic algorithm on a hub location problem to generate initial solutions. In the paper by Chen (Chen, 2007), the author proposed a hybrid heuristic by combining tabu search with simulated annealing. In (Ghosh, 2003), the author used a complete local search along with tabu search on UFLP. Vaithyanathan et al. (Vaithyanathan et al., 1996) used a combination of tabu search and neural networks to solve the simple plant location problem. A hybrid multi-start heuristic was developed by Resende and Werneck (2006) by combining tabu search with local search and path relinking on UFLP.

3.1.2 Summary of tabu search implementation

Table 7 summarizes the findings from the 18 tabu search papers in our survey. Our initial observations from these papers indicate a lack of papers that address capacity constraints and few papers on multi-commodity problems. Also, we are unable to find any paper on multi-objective location problem. On the tabu search implementation side, scarcity of papers implementing intensification techniques, long term memory functions and strategic oscillation shows further scope of improvement. We feel that further experimental studies are required to see the impact of including these techniques.

Table 7: Summary of papers implementing TS on location problems

Paper Reference	Problem Type	Size	Capacitated(Y/N)	Source	Multi-commodity(Y/N)	Multi-stage (Y/N)	Benchmark (Y/N)	Short MF (Y/N)	Medium MF(Y/N)	Long MF(Y/N)	Aspiration (Y/N)	Strategic Oscillation (Y/N)
(Abdinnour-Helm, 1998)	HLP	25	N	NA	N	Y	Y	Y	N	N	Y	N
(Adenso-Diaz and Rodriguez, 1997)	MCLP	213	N	NA	N	N	N	Y	N	N	N	N
(Al-Sultan and Al-Fawzan, 1999)	UFLP	57	N	NA	N	N	Y	Y	N	N	Y	N
(Arostegui et al., 2006)	CFLP	250	Y	Multi	Y	N	N	Y	N	Y	N	N
(Calik et al., 2009)	HLP	81	N	NA	N	Y	Y	Y	N	N	N	Y
(Chen, 2007)	HLP	200	N	NA	N	Y	Y	Y	N	N	N	N
(Delmaire et al., 1999)	CFLP	90	Y	Single	N	N	Y	Y	N	Y	Y	Y
(Ghosh, 2003)	UFLP	400	N	NA	N	N	Y	N	N	Y	Y	N
(Lee and Lee, 2010)	MCLP	150	Y	Single	Y	Y	N	Y	N	N	N	N
(Michel and Hentenryck, 2004)	UFLP	2000	N	NA	N	N	Y	Y	N	Y	N	N
(Mozos and Mesa, 2000)	p-variance	30	N	NA	N	N	N	Y	N	N	Y	N
(Resende and Werneck, 2006)	UFLP	3000	N	NA	N	N	Y	Y	N	Y	N	N
(Rolland et al., 1996)	PMP	500	N	NA	N	N	N	Y	N	Y	Y	Y
(Salhi, 2002)	PMP	500	N	NA	N	N	Y	Y	N	Y	Y	N
(Silva and Cunha, 2009)	HLP	81	N	NA	N	Y	Y	Y	N	Y	Y	N
(Sun, 2006)	UFLP	1000	N	NA	N	N	Y	Y	Y	Y	Y	N
(Adenso-Diaz and Rodriguez, 1997)	MCLP	213	N	NA	N	N	N	Y	N	N	N	N
(Ohlemuller, 1997)	UFLP	150	N	NA	N	N	Y	Y	N	Y	N	N

* Short MF: Short term memory function
* Medium MF: Intermediate term memory function
* Long MF: Long term memory function

* The 'source' column denotes whether the problem is single-source or multi-source for capacitated problems. Since this issue becomes irrelevant for uncapacitated problems, we have used 'NA' to denote 'Not Applicable'.

3.2 Genetic algorithm (GA)

Genetic algorithm is one of the most widely used population based metaheuristics applied to location problems. Genetic algorithm, proposed by Holland (Holland, 1975), is a stochastic search technique based on the mechanism used by natural selection and genetics in evolution theory. One of the earliest applications of GA to location problems was in a paper by Hosage and Goodchild (Hosage and Goodchild, 1986) where they implemented GA on the p-median problem. GA works with a set of feasible solutions called the *population*. Each individual solution in the population is called a *chromosome* and has an assigned fitness value according to the quality of the solution. Chromosomes evolve through successive iterations, which are called *generations*. The population evolves toward better solutions mostly by using three operators: crossover, mutation, and selection in every generation. In crossover, two chromosomes from the current generation are merged to create one or two new chromosomes called *offsprings* for the next generation. In mutation an existing chromosome is modified to create a new one for the next generation. While crossover and mutation produce new chromosomes or solutions using the existing one, given a pair of existing solutions, selection operator simply chooses the better solution for inclusion in the next generation. Selection operator is therefore used to ensure good quality solutions in future generations. Note that these operators are not applied to each chromosome in the population, instead the number of chromosomes on which these operators are applied as well as the specific chromosomes on which they are applied are usually randomly chosen. Typical termination criteria for GA include a maximum number of generations, a maximum number of generations without any improvement in solution value or a limit on the maximum computational time.

We present a list of genetic algorithm applications on discrete facility location problems categorized by the problem variant and problem size in Table 8. As is evident from the table, more researchers have focused on simple plant location, PMP and MCLP. Within SPL, researchers have mainly focused on the UFLP. A probable reason for limited focus on CFLP could be that ensuring feasibility of each solution or chromosome is not as straight forward as in case of uncapacitated problems such as UFLP, PMP, and MCLP. Out of the 17 papers reviewed, we found that eight papers consider a problem size of less than 500 demand points and six papers (with no paper in PMP and PCP) address a problem size of more than 1000 demand points.

Problem Size	SPL	PMP	PCP	MCLP	HLP
≤ 50		(Correa et al., 2004)			(Cunha and Silva, 2007)
$(50, 500]$	(Lai et al., 2010)	(Chaudhry et al., 2003; Chi et al., 2011)	(Chi et al., 2011)	(Jia et al., 2007)	(Topcuoglu et al., 2005)
$(500, 1000]$		(Stanimirovic et al., 2007; Alp and Erkut, 2003; Jaramillo et al., 2002)			
$(1000, 2500]$	(Jaramillo et al., 2002; Salhi, 2003; Kratica et al., 2001)			(Aytug and Saydam, 2002; Krzanowski and Raper, 1999; Jia et al., 2007)	

Table 8: Problem sizes addressed in papers with GA implementation

3.2.1 Genetic algorithm components:

We now consider the components of genetic algorithm, and present our observations from the survey on each of these components. We also identify some unexplored areas within genetic algorithm and specify scope for further research.

Population size: In the published literature, we found two distinct ways by which authors determined the size of the population in their solution approach. The more often used way is to fix the size of the population to a constant for the entire execution of the algorithm irrespective of the size of the problem instance. The other method used in the literature sets the size of the population as a function of the problem size (in which case it remains the same throughout the execution of the algorithm) or sometimes a function of the solution quality obtained so far (wherein the population size changes with the solution quality). The references of the papers that use of either one of these criteria are given in Table 9.

As is evident from the Table 9 papers with specific comments on population size, 13 papers (Aytug and Saydam, 2002; Chaudhry et al., 2003; Correa et al., 2001; Cunha and Silva, 2007; Jourdan and De Weck, 2004; Kratica et al., 2001; Krzanowski and Raper, 1999; Salhi, 2003; Stanimirovic et al., 2007; Topcuoglu et al., 2005; Lai et al., 2010) kept the population size as constant irrespective of the problem parameters. The value of this constant population size varied from 20 (see (Topcuoglu et al., 2005)) to 500 (see (Chaudhry et al., 2003)). 5

Population Type	SPL	PMP	PCP	MCLP	HLP
Constant	(Kratica et al., 2001; Lai et al., 2010; Salhi, 2003)	(Chaudhry et al., 2003; Correa et al., 2001; Stanimirovic et al., 2007; Chi et al., 2011)	(Chi et al., 2011)	(Aytug and Saydam, 2002; Jourdan and De Weck, 2004; Krzanowski and Raper, 1999)	(Cunha and Silva, 2007; Topcuoglu et al., 2005)
Function of problem parameters		(Alp and Erkut, 2003; Correa et al., 2004; Jaramillo et al., 2002)		(Jia et al., 2007; Aytug and Saydam, 2002)	

Table 9: Population type used in papers with GA implementation

papers (Aytug and Saydam, 2002; Jaramillo et al., 2002; Jia et al., 2007; Alp and Erkut, 2003) varied population size as a function of some problem parameter. In the paper by Aytug and Saydam (Aytug and Saydam, 2002), authors specified population size as 75% of problem size with population size not exceeding 100. In (Jia et al., 2007), authors made population size dependent on number eligible facilities and maximum number of facilities to be opened. A similar approach is followed in (Alp and Erkut, 2003), where authors chose population size to be proportional to the value of the ratio between number of demand points and number of facilities to be opened.

Crossover and mutation probability: All the papers surveyed implemented both crossover and mutation operator to improve on solution qualities over generations. While the basic idea behind crossover is to retain good characteristics of parent chromosomes while generating offsprings, mutation is used as a diversification mechanism to explore the less explored regions of the solution space. In a typical crossover, two chromosome are chosen randomly from the current population to generate offsprings which are then included in the next generation. There are many different mechanisms of performing the crossover operation. The most commonly used mechanism splits each chromosome into two parts, where the position of the split is determined randomly. The two split parts from one chromosome are then merged with the two split parts from the other chromosome to generate new chromosomes which are included in the next generation. Another popular implementation of crossover produces offspring by randomly choosing each of the bit value from one of the two parents. Minor variations exist between papers in ways of implementation. The number of pairs of chromosomes that undergo crossover in a generation depends on crossover probability. While crossover is a binary operator requiring two parents, mutation is a unary operator. In mutation, each bit of a chromosome is switched to other possible values than the existing one based on a probability. As mutation is a tool to induce diversification, mutation probability is generally quite small. Again, there are various ways authors implemented mutation in published papers keeping the basic concept same. The crossover probability is much higher (in some cases, as high as 0.99 (Cunha and Silva, 2007)) than mutation probability. In the paper by Jia et al. (Jia et al., 2007), authors investigated the dependency of crossover rate and population size. From the experimental results, they inferred that 35% crossover rate is good with population size of 100 or less whereas with population size of more than 100, $15-25\%$ crossover rate is more appropriate. Summary of crossover probabilities used in various papers is reported in Table 10. Last row in the table lists the papers which mention usage of crossover techniques but corresponding probability values are not given explicitly. Most of the papers implement standard crossover mechanism with exceptions like (Alp and Erkut, 2003; Correa et al., 2004). Impact of different crossover techniques were experimented in the paper by Aytug and Saydam (Aytug and Saydam, 2002).

Most of the published papers include mutation as part of their GA implementation. Typically the mutation probability value changes between 0.005 and 0.8 with most of the papers taking very low values (0.01 to 0.1). Table 11 summarizes our findings by listing papers using different mutation probability values. Last row in the table lists the papers which mention usage of crossover techniques but corresponding probability values are not given explicitly. By *variable* mutation, we refer to a paper by Stanimirovic et al. (Stanimirovic et al., 2007) where authors specified mutation probability as a function of problem size. In the paper by Jaramillo et al. (Jaramillo et al., 2002), authors suggested usage of dynamic mutation rate over generations depending on the rate at which GA converges to the optimal solution. Similarly in (Krzanowski and Raper, 1999), authors reported different mutation probabilities for different problem sizes.

Termination criteria: Three types of termination criteria are used in the literature surveyed: (TC1) maximum number of generations, (TC2) maximum number of generations without any improvement in the best solution value and (TC3) the execution time. Table 12 provides a summary of different termination criteria used in surveyed papers with most papers using maximum number of generations to terminate GA. Some Papers such as (Jia et al., 2007; Jaramillo et al., 2002; Chaudhry et al., 2003; Stanimirovic et al., 2007; Topcuoglu et al., 2005) combine more than one termination criteria. Within the papers in which the maximum number of generations was used as termination criteria, some authors fixed it to a constant independent of

Crossover Probability	SPL	PMP	PCP	MCLP	HLP
$(0, 0.2]$		(Correa et al., 2004)			
$(0.2, 0.4]$	(Jaramillo et al., 2002)	(Jaramillo et al., 2002)		(Jia et al., 2007)	
$(0.4 - 0.6]$		(Stanimirovic et al., 2007)		(Krzanowski and Raper, 1999)	(Topcuoglu et al., 2005)
$(0.6 - 0.8]$	(Lai et al., 2010)			(Aytug and Saydam, 2002)	
$(0.8 - 1.0]$	(Kratica et al., 2001; Salhi, 2003)	(Chaudhry et al., 2003; Stanimirovic et al., 2007)			(Cunha and Silva, 2007)
Not mentioned		(Alp and Erkut, 2003; Chi et al., 2011)	(Chi et al., 2011)		

Table 10: Crossover probabilities in papers with GA implementation

Mutation Probability	SPL	PMP	PCP	MCLP	HLP
$(0, 0.1]$	(Kratica et al., 2001; Lai et al., 2010)	(Correa et al., 2004; Chaudhry et al., 2003)		(Aytug and Saydam, 2002; Jia et al., 2007; Jourdan and De Weck, 2004)	(Cunha and Silva, 2007)
$(0.1, 0.2]$		(Jaramillo et al., 2002)			
$(0.2, 1.0]$		(Jaramillo et al., 2002; Salhi, 2003)		(Krzanowski and Raper, 1999)	(Topcuoglu et al., 2005)
$Variable$		(Stanimirovic et al., 2007)			
Not mentioned		(Chi et al., 2011)			

Table 11: Mutation probabilities in papers with GA implementation

problem size whereas some other authors (Salhi, 2003; Alp and Erkut, 2003) made it dependent on problem size.

Termination Criteria	SPL	PMP	PCP	MCLP	HLP
$TC1$	(Salhi, 2003; Jaramillo et al., 2002)	(Alp and Erkut, 2003; Chaudhry et al., 2003; Correa et al., 2004; Jaramillo et al., 2002; Stanimirovic et al., 2007)		(Aytug and Saydam, 2002; Jia et al., 2007; Jourdan and De Weck, 2004)	(Cunha and Silva, 2007; Topcuoglu et al., 2005)
$TC2$	(Kratica et al., 2001)	(Jaramillo et al., 2002; Stanimirovic et al., 2007)		(Jia et al., 2007)	(Topcuoglu et al., 2005)
$TC3$		(Chaudhry et al., 2003)			

Table 12: Termination criteria used in papers with GA implementation

3.2.2 Summary of implementation:

Table 13 summarizes our observations from the 11 papers reviewed. As mentioned earlier, we report maximum number of demand points considered in the paper under the problem size column. Under population size column, we denote *Fixed* when a fixed number is considered as population size or *Variable* when the number varies with problem parameters.

Table 13: Summary of papers implementing GA on location problems

Paper Reference	Problem Type	Size	Capacitated (Y/N)	Source	Multi-commodity (Y/N)	Multi-stage (Y/N)	Benchmark (Y/N)	Population Size	Crossover (Y/N)	Mutation **(Y/N)**
(Alp and Erkut, 2003)	PMP	1000	N	NA	N	N	Y	Variable	Y	N
(Aytug and Saydam, 2002)	MCLP	1600	N	NA	N	N	N	Variable	Y	Y
(Chaudhry et al., 2003)	PMP	150	N	NA	N	N	Y	Fixed	Y	Y
(Chi et al., 2011)	PMP PCP MCLP	390	Y	Multi	Y	Y	Y	Fixed	Y	Y
(Correa et al., 2004)	PMP	43	Y	Multi	N	N	N	Fixed	Y	Y
(Jaramillo et al., 2002)	PMP UFLP	1000	Y	Multi	N	N	Y	Variable	Y	Y
(Jia et al., 2007)	MCLP	2054	N	NA	N	N	N	Variable	Y	Y
(Kratica et al., 2001)	UFLP	2000	N	NA	N	N	Y	Fixed	Y	Y
(Lai et al., 2010)	CFLP	500	Y	Multi	N	N	Y	Fixed	Y	Y
(Stanimirovic et al., 2007)	PMP	900	N	NA	N	N	Y	Fixed	Y	Y
(Topcuoglu et al., 2005)	HLP	200	N	NA	N	Y	Y	Fixed	Y	Y

As is evident from the table, there is a lack of GA applications in the area of single-source CFLP and multi-commodity location problems. For multi-objective location problems, we could find only one paper (Chi et al., 2011). We also observed that a mix of different benchmark problems and real life problems have been used for testing the effectiveness of the proposed algorithms. We also observed that most of the papers do not give rationale for choosing any particular implementation of the three operators, also the impact of variable population size, crossover and mutation rates is not extensively researched.

3.3 Particle swarm optimization (PSO)

Particle Swarm Optimization is a population based evolutionary computational technique developed by Kennedy and Eberhart (Kennedy and Eberhart, 1995) in 1995. The concept emerged while simulating the unpredictable choreography of bird flock. Similar to genetic algorithm, PSO starts with a set of random feasible solutions (called *particles*) and moves through the problem space in search of a better solution (Eberhart et al., 1996). In the class of location problems, the initial applications of PSO were on continuous location problem. In these problems, each particle keeps track of coordinates of opened facilities in the problem space. Each particle is also assigned a random component, called *velocity*, to change the coordinate to move toward a better solution. Each particle also keeps track of its best solution achieved so far (called *pbest*). The best solution achieved across all solutions (particles) is stored in *gbest*. In every iteration (generation) of PSO, solutions are changed using *velocity* component. The velocity is updated in every iteration by taking certain characteristics of pbest and gbest solutions weighted by a component called *acceleration*. There is also a local version of PSO in which, in addition to pbest, each particle keeps track of the best solution obtained in the local neighborhood called *lbest*. Typical terminating conditions for PSO include maximum number of iterations, maximum number of iterations without any improvement in solution value or maximum computational time.

In this paper, we provide a brief summary of applications of PSO to discrete facility location problems. Lack of papers in this area has restricted us to do a limited survey in comparison to the other metaheuristics. Application of PSO to continuous facility location problems is straightforward and hence is evident by the larger number of papers published (Guner and Sevkil, 2008; Sevkil and Gurner, 2006). When implementing PSO on discrete facility location problems, the implementation process is not straightforward. In the revised framework, instead of using velocity to change coordinates of the location, operators like 1-change crossover are used (Guner and Sevkil, 2008) along with pbest and gbest solutions to develop new solutions from the existing set of solutions.

Within the papers surveyed on PSO implementation for discrete location problems, most of the papers use uncapacitated facility location as the problem context, e.g. (Guner and Sevkil, 2008; Sevkil and Gurner, 2006; Wang et al., 2008). Some other variants such as multiple facility location problem with competition (Yano et al., 2008), multi-objective facility location problems (Yapicioglu et al., 2007; Uno et al., 2007), capacitated facility location problems (Wang and Watada, 2012), p-median problems (Garica and Perez, 2008) are also addressed in the published literature.

Due to the limited number of implementations for each problem variant, we do not provide details of PSO implementation for each paper as we did for other metaheuristics. Given the number of papers published in reputed journals, we think PSO has not matured to the point that it can outdo other metaheuristics applied to discrete location problems. This may be due to internal limitation in the heuristic or it may indicate an unexplored area - it is yet to be decided.

3.4 Scatter Search

Originally proposed by Fred Glover (Glover, 1977, 1998), scatter Search is a population based evolutionary method which has been successfully applied to many hard combinatorial optimization problems (Marti et al., 2006). It maintains a set of solutions called the *reference set*, and iteratively combines solutions in the reference set to generate new solutions, which are in turn used to update the reference set. The method terminates when either the set of solutions does not change or after a pre-specified time limit or after a pre-specified number of iterations.

Although the framework of scatter search appears similar to that of genetic algorithm, the basic principles on which the two methods work are significantly different. Unlike genetic algorithm, scatter search uses a much smaller reference set and applies more systematic procedures to ensure diversity and coverage of the solution space. Similarly, while in genetic algorithm solutions are chosen randomly for crossover, scatter search uses a more systematic and exhaustive procedure to choose the solutions for combination. It also uses intensification strategies such as local search, tabu search etc. to improve upon each solution generated by the combination method.

The method starts with a seed solution which can either be generated randomly or using any method that exploits the problem structure (Keskin and Uster, 2007b). The seed solution is then used to generate a population of diverse solutions by iteratively using a *diversification generation* method (Glover, 1977, 1998) and an *improvement* method, which generally employs local search or tabu search with problem specific neighborhood structure, although other strategies also exist (see (Glover, 1977, 1998)). The improved solution, thus obtained, is added to the population if it does not already exist.

The reference set is then built by first choosing a pre-specified number of good quality solutions from the population. A set of diverse solutions from the population is then chosen by the following procedure. Given a metric to measure the distance between two solutions, the solution in the population which maximizes the minimum distance between this solution and any solution in the reference set is added to the reference set.

Once the initial reference set is generated, a *subset generation* method is used to generate subsets of the reference sets. The solutions in these subsets are combined to generate new solutions. With these new solutions, the reference set is updated using the reference set update method. This method replaces any worse quality solutions in the current reference set by a newly generated solution. Similarly the reference set also replaces diversified solutions, if the newly generated solutions are more diversified than those in the current reference set.

The subset generation method, improvement method and the reference set update methods are applied iteratively for a pre-specified time limit or a pre-specified number of iterations or until the reference set does not change. The method then outputs the best solution from the reference set.

If compared with number of papers that implemented GA or tabu search on discrete location problems, number of papers that report scatter search implementation is significantly less. Hence we follow a different representation schema to describe the literature on scatter search by presenting an aggregate view and paper-wise details together while explaining problem characteristics or metaheuristic components.

Table 14 presents the different variants of discrete location problems along with the problem size on which scatter search has been applied. From the Table, it is evident that the p-median problem and simple plant location problems are two primarily used problem categories in papers implemented scatter search on location problems.

Problem Size	SPL	PMP	PCP	MCLP	HLP
≤ 50	(Du and Evans, 2008)				
$(50, 500]$	(Contreras and Diaz, 2008)	(Scheuerer and Wendolsky, 2006)	(Pacheco and Casado, 2005)		(Keskin and Uster, 2007b,a)
$(500, 1000]$		(Diaz and Fernandez, 2006)			
$(1000, 2500]$		(Garcia-Lopez et al., 2003)			

Table 14: Problem sizes in papers with SS implementation

As can be noted from the table only two papers address problems of size of more than 500 nodes. Also only two papers (Du and Evans, 2008; Contreras and Diaz, 2008) attempted scatter search on simple facility location, which is not so commonly seen in other metaheuristics. Unlike the papers surveyed in previous sections, most of the papers address capacitated location problems. There is only one paper dealing with multiple products and no paper with multiple objectives.

3.4.1 Scatter Search Components

In this section, we describe the characteristics of the scatter search implementation. Table 18 presents the implementation details for each of the five components of scatter search along with the type of solution encoding used.

Initial solution generation: Seed solution is used for generating the population of diverse and good quality initial solutions which are then used for combination. While Scheuerer and Wendolsky (Scheuerer and Wendolsky, 2006) used a construction heuristic to ensure diversity, Keskin and Uster used randomly generated seed (Keskin and Uster, 2007a) and a combination of random and greedy heuristic (Keskin and Uster, 2007b) for initial solution. Many papers (Garcia-Lopez et al., 2003; Pacheco and Casado, 2005; Diaz and Fernandez, 2006; Contreras and Diaz, 2008; Du and Evans, 2008) did not use a seed solution instead generated the population directly.

Diversification generation method: In a paper by Garcia-Lopez et al. (Garcia-Lopez et al., 2003), authors used a diversity based multi-start construction heuristic to generate the initial population. While the heuristic used by them (Garcia-Lopez et al., 2003) emphasizes both the solution quality and diversity, the GRASP based heuristics used in papers like (Pacheco and Casado, 2005; Diaz and Fernandez, 2006; Contreras and Diaz, 2008) concentrate only on the solution quality and the adaptive memory based heuristic used by Scheuerer and Wendolsky (Scheuerer and Wendolsky, 2006) concentrate only on solution diversity. In (Keskin and Uster, 2007b) and (Keskin and Uster, 2007a), a construction heuristic proposed by (Glover, 1998) is used to generate the initial population. While all the above papers used systematic methods to generate the initial population, (Du and Evans, 2008) randomly generated the initial population.

Strategies	**SPL**	**PMP**	**PCP**	**MCLP**	**HLP**
Construction (single/multi-start)	(Garcia-Lopez et al., 2003)				(Keskin and Uster, 2007b,a)
GRASP	(Contreras and Diaz, 2008)	(Diaz and Fernandez, 2006)	(Pacheco and Casado, 2005)		
Random		(Du and Evans, 2008)			
Adaptive memory		(Scheuerer and Wendolsky, 2006)			

Table 15: Diversification generation methods in papers with SS implementation

Improvement method: Improvement methods are used for ensuring feasibility of a solution and to intensify the search around this solution. They can be applied both to the solutions generated for inclusion in the initial population and to the solutions generated by combining two or more solutions. Most of the scatter search implementations for facility location problems except (Pacheco and Casado, 2005; Du and Evans, 2008) use local search with different neighborhood structures as improvement methods. While (Du and Evans, 2008) uses two heuristics only to balance the capacities allocated to facilities, and does not use any specific improvement method, (Pacheco and Casado, 2005) uses alternate and exchange heuristics to improve the solutions. Local search based improvement methods use shift and swap neighborhoods. For instance, (Garcia-Lopez et al., 2003; Keskin and Uster, 2007b) used local search with swap neighborhood, (Diaz and Fernandez, 2006; Scheuerer and Wendolsky, 2006; Contreras and Diaz, 2008) used local search with shift and swap neighborhood. Some implementations (Keskin and Uster, 2007b,a; Contreras and Diaz, 2008) further improve the quality of the best solution output by the local search based improvement method by using path relinking (Keskin and Uster, 2007b,a) and tabu search (Contreras and Diaz, 2008).

Strategies	**SPL**	**PMP**	**PCP**	**MCLP**	**HLP**
Shift	(Contreras and Diaz, 2008)	(Diaz and Fernandez, 2006; Scheuerer and Wendolsky, 2006)			
Swap	(Contreras and Diaz, 2008)	(Garcia-Lopez et al., 2003; Diaz and Fernandez, 2006; Scheuerer and Wendolsky, 2006)			(Keskin and Uster, 2007b)
Path relinking					(Keskin and Uster, 2007b,a)

Table 16: Improvement methods in papers with SS implementation

Reference set generation and update method: Generally the reference set is built by selecting a pre-specified number b_1 of good quality solutions and a pre-specified number b_2 of diversified solutions. For instance, to build the reference set (Garcia-Lopez et al., 2003; Pacheco and Casado, 2005; Scheuerer and Wendolsky, 2006; Keskin and Uster, 2007b,a) choose b_1 best solutions in terms of their cost and and b_2 most diversified solutions with respect to the solutions already in the reference set. While the previously mentioned papers use a combination of good quality and diversified solutions, (Du and Evans, 2008; Contreras and Diaz, 2008) use only good quality solutions to build the reference set.

Many implementations (Scheuerer and Wendolsky, 2006; Keskin and Uster, 2007b,a; Du and Evans, 2008) strive for the balance between the quality of solutions and diversity of solutions during the reference set update as well. In these implementations, if the solution obtained by the combination method is either better than an existing solution in the b_1 best quality solutions in the reference set, or is more diversified than a solution in the b_2 most diversified solutions in reference set, then this solution replaces an existing worse solution in the reference set. Some papers like (Pacheco and Casado, 2005; Du and Evans, 2008; Contreras and Diaz, 2008) use the quality of the solution as the only criterion for updating the reference set.

Subset generation method: To generate the subsets of solutions that are combined together to generate new solutions, (Pacheco and Casado, 2005; Du and Evans, 2008) consider all possible subsets of size two, (Diaz and Fernandez, 2006; Contreras and Diaz, 2008) consider all possible subsets of size two and three of the reference set. (Scheuerer and Wendolsky, 2006) considers all possible subsets of size two and from these subsets generated subsets of size three by adding the best solution not present in the subset to the subset. (Keskin and Uster, 2007b,a) use subsets of sizes three and four.

Subset Size	SPL	PMP	PCP	MCLP	HLP
2	(Du and Evans, 2008; Contreras and Diaz, 2008)	(Diaz and Fernandez, 2006)	(Pacheco and Casado, 2005)		
3	(Contreras and Diaz, 2008)	(Diaz and Fernandez, 2006)			(Keskin and Uster, 2007b,a)
4					(Keskin and Uster, 2007b,a)

Table 17: Strategies for subset generation in papers with SS implementation

Solution combination method: To combine the solutions in the subsets generated by the subset generation method, (Keskin and Uster, 2007b,a) use a weighted linear combination wherein the reciprocals of the objective function values are used as weights. (Garcia-Lopez et al., 2003) constructs a partial solution by opening the facilities common to all the solutions in the subset. They then identify a location farthest to this partial solution and randomly choose a location that is within a given radius of this farthest location. This procedure is repeated until the solution is complete. While (Garcia-Lopez et al., 2003) uses combination by construction, (Diaz and Fernandez, 2006; Contreras and Diaz, 2008) use combination by destruction. To generate a new solution through combination, (Diaz and Fernandez, 2006; Contreras and Diaz, 2008) initially assign clients to facilities that are open in at least one solution in the subset and then systematically reduce the number of open facilities. (Du and Evans, 2008) uses a random solution combination method wherein the probability of turning a bit to one depends on the fixed costs of these two solutions. Combination by voting (Scheuerer and Wendolsky, 2006) and combination by path relinking (Pacheco and Casado, 2005) are also used to generate new solutions.

Advanced strategies: The template for scatter search (Marti et al., 2006) proposes many advanced strategies for better exploration of the solution space including dynamic reference set updating, reference set rebuilding, reference set tiers, diversity control, use of memory etc. While some of these have been used in context of facility location problems, others have not yet been explored. For instance, (Garcia-Lopez et al., 2003; Pacheco and Casado, 2005; Scheuerer and Wendolsky, 2006) use reference set rebuilding for preventing premature convergence of scatter search. In scatter search, if it is not possible to generate new subsets before the termination criteria are satisfied, diversification method is again called upon to bring in new solutions and the process restarts.

Dynamic reference set updating is used for better exploration of the solution space by allowing more solutions to be combined (Keskin and Uster, 2007b). Another advanced strategy that has been used in context of facility location problems, is usage of adaptive memory. For instance, (Pacheco and Casado, 2005; Scheuerer and Wendolsky, 2006) use adaptive memory to keep track of the number of solutions in the reference set in which a particular facility has been opened.

3.4.2 Summary of scatter search implementation

We observed scatter search application as one of the emerging research interests in metaheuristic applications on location problems. It has its limitations though in terms of problem characteristics addressed and methodologies implemented to tackle location problems. Like in the papers on scatter search implementation, very few papers are available to address the following problem characteristics: large problem size, multiple commodities, multiple objectives etc. Similarly in the metaheuristic implementation part, various issues are yet to be tested. Improvement method in scatter search consists of mostly local search in published papers, with scope of introducing advanced local search, e.g. tabu search etc., still remains. For diversification generation, GRASP and random search are cited in most of the papers. Other diversification methodologies (see (Glover, 1998)) are not explored properly apart from papers by Keskin and Ustar (Keskin and Uster, 2007b,a). While updating reference sets, it is suggested to include solutions based on quality and diversity. In some papers (Contreras and Diaz, 2008; Du and Evans, 2008), authors considered only quality to include solutions in the reference set.

Table 18: Summary of papers implementing SS on location problems

Paper Reference	Problem Variant	Size	Capacitated (Y/N)	Source	Multi-commodity (Y/N)	Multi-stage (Y/N)	Seed Generation Method	Diversification Generation Method	Improvement Method	Reference Set Generation Method	Subset Generation Method
(Garcia-Lopez et al., 2003)	PMP		N	NA	N	N	None	Multi-start Construction	LS	QD	
(Pacheco and Casado, 2005)	PCP		N	NA	N	N	None	GRASP	EAH	QD	A-2
(Diaz and Fernandez, 2006)	PMP		Y	Single	N	N	None	GRASP	LS	QD	A-2, A-3
(Scheuerer and Wendolsky, 2006)	PMP		Y	Single	N	N	Construction	Adaptive memory based Construction	LS	QD	A-2, S-3
(Keskin and Uster, 2007b)	HLP		Y	Multi	Y	N	Random	Construction	LS	QD	S-3, S-4
(Keskin and Uster, 2007a)	HLP		Y	Single	Y	Y	None	GRASP	LS	Q	S-3, S-4
(Contreras and Diaz, 2008)	SPL		Y	Single	N	N	None	GRASP	LS	Q	A-2, A-3
(Du and Evans, 2008)	SPL		Y	Multi	Y	N	None	Random	None	Q	A-2

* LS: Local Search
* QD: Combination of good quality and diversified solutions
* Q: Only good quality solutions
* EAH: Exchange and alternate heuristic
* A-i: All subsets of size i
* S-j: All subsets of size j

* The 'source' column denotes whether the problem is single-source or multi-source for capacitated problems. Since this issue becomes irrelevant for uncapacitated problems, we have used 'NA' to denote 'Not Applicable'.

4 Conclusion

This paper highlights some of untouched areas in location problems along with some implementation gaps of four metaheuristics in recent practice: genetic algorithm, tabu search, particle swarm optimization and scatter search. We reported some observations while summarizing each metaheuristic in its respective section. In this section, we highlight some common issues found across sections which may be of interest to the research community. In the problem context area, the problem size is still restricted to a range of 2000 to 2500 nodes with only 9 papers addressing problem size of more than 1000 nodes. Apart from scatter search literature, capacitated facility location problems are scarcely handled in the literature. Also multi-commodity, multi-objective facility locations are rarely taken as testbed problems which creates another interesting area for researchers to work. Among the problem categories, p-center is used as problem context in only 3 papers.

Regarding changes in metaheuristic implementation, we found some gaps specific to each metaheuristic which can be addressed by systematic experimental study. In genetic algorithm, it is interesting to see the effectiveness of variable population size, crossover and mutation rate on the quality of final solution obtained. In tabu search, we did not find papers to investigate the effect of intermediate term memory function etc. Similarly scatter search also requires some experimental study for better evaluation of other improvement methods, diversification mechanisms etc. In our view, these aspects contain the research prospects to guide future work on this area.

References

Abdinnour-Helm, S. (1998). A hybrid heuristic for the uncapacitated hub location problem. *European Journal of Operational Research*, 106:489–499.

Adenso-Diaz, B. and Rodriguez, F. (1997). A simple search heuristic for the MCLP: Application to the location of ambulance bases in a rural region. *Omega*, 25: 2:181–187.

Ahuja, R., Orlin, J., Pallottino, S., Scaparra, M., and Scutella, M. (2004). A multi-exchange heuristic for the single-source capacitated facility location problem. *Management Science*, 50(6):749–760.

Al-Sultan, K. S. and Al-Fawzan, M. A. (1999). A tabu search approach to the uncapacitated facility location problem. *Annals of Operations Research*, 86:91–103.

Alp, O. and Erkut, E. (2003). An efficient genetic algorithm for the p-median problem. *Annals of Operations Research*, 122:21–42.

Arostegui, M., Kadipasaoglu, S., and Khumawala, B. (2006). An empirical comparison of tabu search, simulated annealing, and genetic algorithms for facilities location problems. *International Journal of Production Economics*, 103(b):742–754.

Aytug, H. and Saydam, C. (2002). Solving large-scale maximum expected covering location problems by genetic algorithms: A comparative study. *European Journal of Operational Research*, 141:480494.

Beasley, J. (1988). An algorithm for solving large capacitated warehouse location problems. *European Journal of Operational Research*, 33:314–325.

Beasley, J. (1990). OR-library - distributing test problems by electronic mail. *Journal of the Operational Research Society*, 41 (11):1069–1072.

Beasley, J. (1993). Lagrangean heuristics for location problems. *European Journal of Operational Research*, 65:383–399.

Beasley, J. (1996). Obtaining test problems via internet. *Journal of Global Optimization*, 8:429–433.

Beasley, J. and Chu, P. (1996). A genetic algorithm for the set covering problem. *European Journal of Operational Research*, 94:392–404.

Calik, H., Alumur, S., Kara, B., and Karasan, O. (2009). A tabu-search based heuristic for the hub covering problem over incomplete hub networks. *Computers & Operations Research*, 36:12:3088–3096.

Campos, V., Laguna, M., and Mart, R. (2005). Context-independent scatter and tabu search for permutation problems. *INFORMS Journal on Computing*, 17: 1:111–122.

Chaudhry, S. S., He, S., and Chaudhry, P. E. (2003). Solving a class of facility location problems using genetic algorithms. *Expert Systems*, 20, No. 2:86–92.

Chen, J.-F. (2007). A hybrid heuristic for the uncapacitated single allocation hub location problem. *Omega*, 35:2:211–220.

Chi, T., Yang, H., and Hsiao, H. (2011). A new hierarchical facility location model and genetic algorithm for humanitarian relief. In *Proceedings of the Information Science and Service Science (NISS), 2011 5th International Conference on New Trends, Macao, China*, pages 367–374.

Choi, I. and Chaudhry, S. (1993). The p-median problem with maximum distance constraints: a direct approach. *Location Science*, 1:235–243.

Chung, C. (1986). Recent applications of the maximal covering location planning model. *Journal of the Operational Research Society*, 37(8):735–746.

Church, R. and Revelle, C. (1974). The maximal covering problem. *Papers of the Regional Science Association*, 32:101–118.

Contreras, I., Cordeau, J. F., and Laporte, G. (2011). The dynamic uncapacitated hub location problem. *Transportation Science*, 45: 1:18–32.

Contreras, I. A. and Diaz, J. (2008). Scatter search for the single source capacitated facility location problem. *Annals of Operations Research*, 157:73–89.

Correa, E., Steiner, M., Freitas, A., and Carnieri, C. (2001). A genetic algorithm for the p-median problem. In *Procedings of 2001 Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 1268–1275.

Correa, E., Steiner, M., Freitas, A., and Carnieri, C. (2004). A genetic algorithm for solving a capacitated p-median problem. *Numerical Algorithms*, 34:373–388.

Cunha, C. B. and Silva, M. R. (2007). A genetic algorithm for the problem of configuring a hub-and-spoke network for a ltl trucking company in brazil. *European Journal of Operational Research*, 179:747758.

Daskin, M. (1983). A maximal expected covering location model: Formulation properties and heuristic solution. *Transportation Science*, 17:48–69.

Daskin, M. (1995). *Network and discrete location: models, algorithms, and applications*. Wiley, New York.

Delmaire, H., Daz, J., Fernndez, E., and Ortega, M. (1999). Reactive GRASP and tabu search based heuristics for the single source capacitated plant location problem. *Information Systems and Operational Research*, 37(3):194–225.

Diaz, J. and Fernandez, E. (2006). Hybrid scatter search and path relinking for the capacitated p-median problem. *European Journal of Operational Research*, 169:570–585.

Drezner, Z. (1995). *Facility location: a survey of applications and methods*. Springer, New York.

Drezner, Z. and Hamacher, H. W., editors (2004). *Facility location: applications and theory*. Springer.

Du, F. and Evans, G. (2008). A bi-objective reverse logistics network analysis for post-sale service. *Computers & Operations Research*, 35:2617 2634.

Eberhart, R., Simpson, P., and Dobbins, R. (1996). *Computational Intelligence PC Tools*. Academic Press Professional.

Ernst, A. and Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science*, 4:3:139–154.

Ernst, A. and Krishnamoorthy, M. (1998). Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, 104:100–112.

Garcia-Lopez, F., Melian-Batista, B., Moreno-Perez, J., and Moreno-Vega, J. (2003). Parallelization of the scatter search for the p-median problem. *Parallel Computing*, 79:575–589.

Garica, F. J. M. and Perez, J. A. M. (2008). Jumping frogs optimization: A new swarm method for discrete optimization. *JFO for Discrete Optimization*, DEIOC 3/2008:1–10.

Ghosh, D. (2003). Neighborhood search heuristics for the uncapacitated facility location problem. *European Journal of Operational Research*, 150:150–162.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences*, 8(1):156–166.

Glover, F. (1989). Tabu search– part I. *ORSA Journal on Computing*, 1:0190–0206.

Glover, F. (1998). *Artificial Evolution*, chapter A template for scatter search and path relinking, pages 1–51. Springer Berlin Heidelberg.

Guner, A. R. and Sevkil, A. (2008). A discrete particle swarm optimization algorithm for uncapacitated facility location problem. *Journal of Artificial Evolution and Applications*, Article ID 861512,:9.

Hoefer, M. (2002). Ufllib.

Hoefer, M. (2003). *Experimental and Efficient Algorithms*, volume 2647 of *Lecture Notes in Computer Science*, chapter Experimental Comparison of Heuristic and Approximation Algorithms for Uncapacitated Facility Location, pages 165–178. Springer-Verlag Berlin Heidelberg.

Hoefer, M. (2004). Ufllib.

Holland, J. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press.

Holmberg, K., Rnnqvist, M., and Yuan, D. (1999). An exact algorithm for the capacitated facility location problems with single sourcing. *European Journal of Operational Research*, 113(3):544–559.

Hosage, C. and Goodchild, M. (1986). Discrete space location-allocation solutions from genetic algorithms. *Annals of Operational Research*, 6:35–46.

Huang, R., Kim, S., and Menezes, M. B. (2010). Facility location for large-scale emergencies. *Annals of Operations Research*, 181: 1:271–286.

Jaramillo, J. H., Bhadury, J., and Batta, R. (2002). On the use of genetic algorithms to solve location problems. *Computers & Operations Research*, 29:761–779.

Jia, H., Ordonez, F., and Dessouky, M. M. (2007). Solution approaches for facility location of medical supplies for large-scale emergencies. *Computers & Industrial Engineering*, 52:257276.

Jourdan, D. B. and De Weck, O. L. (2004). Multi-objective genetic algorithm for the automated planning of a wireless sensor network to monitor a critical facility. In *Proc. of SPIE Defense and Security Symposium*.

Karg, R. and Thompson, G. (1964). A heuristic approach to solving travelling salesman problems. *Management Science*, 10:225–248.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proc. IEEE International Conf. on Neural Networks, IEEE Service Center, Piscataway, NJ.*

Keskin, B. and Uster, H. (2007a). Meta-heuristic approaches with memory and evolution for a multi-product production/distribution system design problem. *European Journal of Operational Research*, 182:663–682.

Keskin, B. and Uster, H. (2007b). A scatter search-based heuristic to locate capacitated transshipment points. *Computers & Operations Research*, 34:3112–3125.

Kim, H. and O'Kelly, M. E. (2009). Reliable p-hub location problems in telecommunication networks. *Geographical Analysis*, 41: 3:283–306.

Kochetov, Y. (2003). Benchmarks library.

Koerkel, M. (1989). On the exact solution of large-scale simple plant location problems. *European Journal of Operational Research*, 39:157–173.

Kratica, J., Tosic, D., Filipovic, V., and Ljubic, I. (2001). Solving the simple plant location problem by genetic algorithm. *RAIRO Oper. Res*, 35:127–142.

Krzanowski, R. M. and Raper, J. (1999). Hybrid genetic algorithm for transmitter location in wireless networks. *Computers, Environment and Urban Systems*, 23:352–382.

Lai, M., Sohn, H., Tseng, T., and Chiang, C. (2010). A hybrid algorithm for capacitated plant location problem. *Expert Systems with Applications*, 37:12:8599–8605.

Lee, J. and Lee, Y. (2010). Tabu based heuristics for the generalized hierarchical covering location problem. *Computers & Industrial Engineering*, 58:4:638–645.

Liu, C. M., Kao, R., and Wang, A. (1982). Properties and solution methods for large location-allocation problem. *Journal of the Operational Research Society*, 33:443–452.

Lorena, L. and Senne, E. (2003). Local search heuristics for capacitated p-median problems. *Networks and Spatial Economics 3*, 3:409–419.

Lorena, L. and Senne, E. (2004). A column generation approach to capacitated p-median problem. *Computers and Operations Research*, 31:863–876.

Marti, R., Laguna, M., and Glover, F. (2006). Principles of scatter search. *European Journal of Operational Research*, 169:359–372.

Melo, M., Nickel, S., and Saldanha-da Gama, F. (2009). Facility location and supply chain management a review. *European Journal of Operational Research, Volume 196, Issue 2, 16 July 2009, Pages 401-412*, 196:2:401–412.

Michel, L. and Hentenryck, P. (2004). A simple tabu search for warehouse location. *European Journal of Operational Research*, 157:576–591.

Mirchandani, P. and R.L., F., editors (1990). *Discrete location theory*. Wiley.

Mladenovic, N., Brimberg, J., Hansen, P., and Moreno-Prez, J. (2007). The p-median problem: A survey of metaheuristic approaches. *European Journal of Operational Research, Volume 179, Issue 3, 16 June 2007, Pages 927-939*, 179:3:927–939.

Mozos, M. and Mesa, J. A. (2000). Heuristics for multiple facility location in a network using the variance criterion. *The Journal of the Operational Research Society*, 51:8:971–981.

Nagy, G. and Salhi, S. (2007). Location-routing: Issues, models and methods. *European Journal of Operational Research*, 177:2:649–672.

Ohlemuller, M. (1997). Tabu search for large location-allocation problems tabu search for large location-allocation problems. *The Journal of the Operational Research Society*, 48:7:745–750.

O'Kelly, M. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32:393–404.

Pacheco, J. and Casado, S. (2005). Solving two location models with few facilities by using a hybrid heuristic: a real health resources case. *Computers & Operations Research*, 32:3075–3091.

Prez, J., Moreno-Vega, J., and Martn, I. (2003). Variable neighborhood tabu search and its application to the median cycle problem. *European Journal of Operational Research*, 151:2:365–378.

Reinelt, G. (1991). TSP-Lib a travelling salesman library. *ORSA Journal of Computing*, 3:376–384.

Resende, M. and Werneck, R. (2004). A hybrid heuristic for the p-median problem. *Journal of Heuristics*, 10:59–88.

Resende, M. and Werneck, R. (2006). A hybrid multistart heuristic for the uncapacitated facility location problem. *European Journal of Operational Research*, 174:1:54–68.

Rolland, E., Schilling, D., and Current, J. (1996). An efficient tabu search procedure for the p-median problem. *European Journal of Operational Research*, 96:2:329–342.

Salhi, S. (2002). Defining tabu list size and aspiration criterion within tabu search methods. *Computers & Operations Research*, 29: 1:67–86.

Salhi, S. (2003). A genetic algorithm based approach for the uncapacitated continuous location - allocation problem. *Annals of Operations Research*, 123:203222.

Scheuerer, S. and Wendolsky, R. (2006). A scatter search heuristic for the capacitated clustering problem. *European Journal of Operational Research*, 169:533–547.

Sevkil, M. and Gurner, A. R. (2006). A continuous particle swarm optimization algorithm for uncapacitated facility location problem. *LNCS*, 4150:316–323.

Shen, Z. J. M., Coullard, C., and Daskin, M. S. (2003). A joint location-inventory model. *Transportation Science*, 37: 1:40–55.

Shen, Z. J. M., Z. R. L. . Z. J. (2011). The reliable facility location problem: Formulations, heuristics, and approximation algorithms. *INFORMS Journal on Computing*, 23: 3:470–482.

Silva, M. and Cunha, C. (2009). New simple and efficient heuristics for the uncapacitated single allocation hub location problem. *Computers & Operations Research*, 36:12:3152–3165.

Snyder, L. (2006). Facility location under uncertainty: A review. *IIE Transactions*, 38, No. 7:537–554.

Stanimirovic, Z., Kratica, J., and Dugosija, D. (2007). Genetic algorithms for solving the discrete ordered median problem. *European Journal of Operational Research*, 182:983–1001.

Sun, M. (2006). Solving the uncapacitated facility location problem using tabu search. *Computers & Operations Research*, 33:9:2563–2589.

Tan, P. and Kara, B. (2007). A hub covering model for cargo delivery systems. *Networks*, 49:2839.

Tansel, B., Francis, R., and Lowe, T. (1983). State of the art-location on networks: A survey. part i: The p-center and p-median problems. *Management Science*, 29, No. 4:482–497.

Topcuoglu, H., Corut, F., Ermis, M., and Yilmaz, G. (2005). Solving the uncapacitated hub location problem using genetic algorithms. *Computers & Operations Research*, 32:4:967–984.

Toregas, C., Swain, R., Revelle, C., and Bergman, L. (1971). The location of emergency service facilities. *Operations Research*, 19:1363–1373.

Tsai, C. F., Tsai, C. W., and Tseng, C. C. (2004). A new hybrid heuristic approach for solving large traveling salesman problem. *Information Sciences*, 166: 1:67–81.

Uno, T., Katagiri, H., and Kato, K. (2007). A fuzzy model for the multiobjective emergency facility location problem with A-distance. *The Open Cybernetics and Systemics Journal*, 1:21–27.

Vaithyanathan, S., Burke, L., and Magent, M. (1996). Massively parallel analog tabu search using neural networks applied to simple plant location problems. *European Journal of Operational Research*, 93:2:317–330.

Verter, V. (2011). *Uncapacitated and capacitated facility location problems in 'Foundations of Location Analysis'*, volume 155 of *International Series in Operations Research & Management Science*. Springer.

Wang, D., Wu, C., Ip, A., Wang, D., and Yan, Y. (2008). Parallel multi-population particle swarm optimization algorithm for uncapacitated facility location problem using OpenMP. In *In Evolutionary Computation, 2008. CEC 2008 (IEEE World Congress on Computational Intelligence)*, pages 1214–1218.

Wang, S. and Watada, J. (2012). A hybrid modified PSO approach to VaR-based facility location problems with variable capacity in fuzzy random uncertainty. *Information Sciences*, 192:3–18.

Yano, F., Shohdohji, T., and Toyoda, Y. (2008). Modification of hybridized particle swarm optimization algorithms applying to facility location problems. In *Proceedings of the 9th Asia Pasific Industrial Engineering & Management Systems Conference*, pages 2278–2287.

Yapicioglu, H., Smith, A. E., and Dozier, G. (2007). Solving the semi-desirable facility location problem using bi-objective particle swarm. *European Journal of Operational Research*, 177:733–749.