ROUTING FOR POST-DISASTER NEEDS ASSESSMENT TO IMPROVE INFORMATION ACCURACY AND PRECISION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

DUYGU PAMUKÇU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

JULY 2018

Approval of the thesis:

ROUTING FOR POST-DISASTER NEEDS ASSESSMENT TO IMPROVE INFORMATION ACCURACY AND PRECISION

submitted by **Duygu Pamukçu** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** ______________

Prof. Dr. Yasemin Serin
Head of Department, **Industrial Engineering** ______________

Assist. Prof. Dr. Melih Çelik
Supervisor, **Industrial Engineering Department, METU** ______________

Assoc. Prof. Dr. Burcu Balçık Koyuncu
Co-supervisor, **Industrial Engineering Dept., Ozyegin Uni.** ______________

Examining Committee Members:

Prof. Dr. Nur Evin Özdemirel
Industrial Engineering Department, METU ______________

Assist. Prof. Dr. Melih Çelik
Industrial Engineering Department, METU ______________

Assoc. Prof. Dr. Burcu Balçık Koyuncu
Industrial Engineering Department, Ozyegin University ______________

Prof. Dr. Haldun Süral
Industrial Engineering Department, METU ______________

Prof. Dr. Bahar Yetiş Kara
Industrial Engineering Department, Bilkent University ______________

Date: ______________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Duygu Pamukçu

Signature :

ABSTRACT

ROUTING FOR POST-DISASTER NEEDS ASSESSMENT TO IMPROVE INFORMATION ACCURACY AND PRECISION

Pamukçu, Duygu
M.S., Department of Industrial Engineering
Supervisor : Assist. Prof. Dr. Melih Çelik
Co-Supervisor : Assoc. Prof. Dr. Burcu Balçık Koyuncu

July 2018, 115 pages

Obtaining reliable information in post-disaster needs assessment depends on how much time is spent for sampling to collect information and how many different beneficiary groups are visited. Estimated information on the needs is an input for subsequent relief distribution, so accuracy and precision of the estimation directly effects the efficiency of relief operations. Since the total time to visit affected sites and to collect information from each site is limited, an efficient routing scheme is important to perform effective needs assessment. Motivated by this, we define the Post-Disaster Needs Assessment Routing Problem, where the decisions of which sites to visit, in what sequence to perform these visits, and how much time to spend to collect information in each site are made, subject to a total needs assessment time constraint. We formulate a mixed integer program and propose a tabu search heuristic to obtain near-optimal solutions. Our solution approaches are tested on randomly generated instances and a case study based on the 2011 Van Earthquake.

Keywords: Humanitarian Logistics, Needs Assessment, Selective and Generalized Vehicle Routing Problem, Accuracy, Precision, Tabu Search

ÖZ

AFET SONRASI İHTİYAÇ DEĞERLENDİRME İÇİN DOĞRULUK VE DUYARLIĞI GELİŞTİREN ROTALAMA PROBLEMİ

Pamukçu, Duygu
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi : Dr. Öğr. Üyesi Melih Çelik
Ortak Tez Yöneticisi : Doç. Dr. Burcu Balçık Koyuncu

Temmuz 2018 , 115 sayfa

Afet sonrası ihtiyaç değerlendirme sürecinde elde edilen bilginin güvenilirliği, bilgi toplamak için örneklemeye ne kadar zaman ayrıldığına ve afetten etkilenen bölgelerin ne kadarının ziyaret edildiğine bağlıdır. Tahmin edilen ihtiyaç bilgisi değerlendirmeden hemen sonra gelen yardım dağıtımı süreci için girdi olarak kullanılır. Bu nedenle, tahmin edilen bilginin doğruluğu ve duyarlığı yardım operasyonlarının etkililiğini doğrudan etkiler. Afetten etkilenen bölgeleri ziyaret etmek ve bu bölgelerden bilgi toplamak için harcanacak zaman kısıtlı olduğu için verimli bir rotalama planı oluşturmak, etkili bir ihtiyaç değerlendirme için önemlidir. Bu çalışmada, afet sonrası ihtiyaç değerlendirme için bir rotalama problemi geliştirilmiştir. Problemin içerdiği kararlar hangi bölgelerin ziyaret edileceği, bu bölgelerin hangi sıra ile ziyaret edileceği ve bilgi toplamak için her bölgede ne kadar zaman geçirileceğidir. Bu kararlar, ihtiyaç değerlendirme için ayrılmış toplam zaman kısıtı içerisinde verilir. Öncelikle karışık tamsayılı programlama ile modellenen Afet Sonrası İhtiyaç Değerlendirme için Rotalama Problemi'ne çözüm yöntemi olarak bir tabu arama algoritması sunulmuştur. Önerilen çözüm yöntemini test etmek için rassal olarak oluşturulmuş örnek problem

seti ve vaka çalışması kapsamında 2011 Van Depremi verileri kullanılarak oluşturulan bir örnek problem seti kullanılmıştır.

Anahtar Kelimeler: İnsani Yardım Lojistiği, İhtiyaç Değerlendirme, Seçici ve Genellenmiş Araç Rotalama Problemi, Doğruluk, Duyarlık, Tabu Arama

To all victims of disasters who face death due to scarcity...

ACKNOWLEDGMENTS

First of all, I owe my deepest gratitude to my supervisor Assist. Prof. Dr. Melih Çelik as he raised my awareness on humanitarian issues in IE/OR research area. I wish to thank him for his endless support, patience and enthusiasm during this thesis and my education in METU-IE Department. Additionally, it gives me great pleasure in acknowledging the support and help of my co-supervisor Assoc. Prof. Dr. Burcu Balçık Koyuncu. Although the distance we have in between, I always feel her support close by me. Her guidance helped and motivated me in all the time of my study. I could not have imagined having a better supervisor pair for my thesis. I experienced moments of pleasure during the time we worked together.

I would also like to thank the rest of my thesis commitee members Prof. Dr. Nur Evin Özdemirel, Prof. Dr. Haldun Süral and Prof. Dr. Bahar Yetiş Kara for their insightful comments and contributions.

Moreover, I acknowledge my colleagues and my friends for their patience and support. I really thank my roommate Ipek Eroğlu for her continuous help and support through my master education. I would like to thank my labmates Nurbanu Demir and Kadir Biçe for not leaving me alone on the sleepness nights. Additionally, I thank my neighbor collegues Selin Akifoğlu and Derya Dinler for their help and enthusiasm.

Last but not the least, I would like to present my special thanks to my family for supporting me spiritually thoughout my life and my besty Sıla Aygün for finding her right beside all the time when I need.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

CARP	Capacitated Arc Routing Problem
CI	Confidence Interval
CIH	Cheapest Insertion Heuristic
CluVRP	Clustered Vehicle Routing Problem
DOM	Disaster Operations Management
EPI	Expended Programme on Immunization
GTSP	Generalized Traveling Salesperson Problem
GVRP	Generalized Vehicle Routing Problem
IFRC	International Federation of Red Cross and Red Crescent Societies
IP	Integer Programming
MSTS	Multi-Start Tabu Search Heuristic
NGO	Non-Governmental Organization
OP	Orienteering Problem
PCTSP	Prize Collecting Traveling Salesperson Problem
PDNARP	Post-Disaster Needs Assessment Routing Problem
PTP	Profitable Tour Problem
SRS	Simple Random Sampling
TOP	Team Orienteering Problem
TPP	Traveling Purchaser Problem
TS	Tabu Search
TSP	Traveling Salesperson Problem
VRP	Vehicle Routing Problem
WHO	World Health Organization

CHAPTER 1

INTRODUCTION

Crises requiring humanitarian relief, such as natural or human-inflicted disasters, may have profound impact on the communities affected by them. This impact may involve fatalities, casualties, damages on the infrastructure, effects on economic activities, and an increase in the need to sustain the quality of basic daily activities. To provide effective response to a humanitarian crisis, these impacts must be forecasted accurately a priori and and/or they should be efficiently and effectively assessed in the aftermath.

In the case of disasters, particularly sudden-onset ones, the uniqueness of the event generally deems accurate forecasting of post-disaster needs (such as the need for food, water, medical aid, shelters, etc.). In such cases, learnings and analyses on previous disasters do not help to predict the impacts of and the needs arising from a new one, since the latter differs by some aspects (such as type, magnitude, scale, population characteristics, etc.) from the previous ones. For this reason, predictions according to past observations may not be close to reality and may lead to fallacies for the post-disaster management process. As it is not possible to make strong assumptions by referencing the past observations, reliable information can only be obtained by making specific surveys in the affected region. Decisions on from where and how to collect the information are critical, since the management of post-disaster operations is planned according to the information collected during these needs assessment activities. Consequently, estimation of the beneficiaries' needs should not only be reliable, but also as close enough to reality as possible.

Each disaster has unique dynamics and requires a different needs assessment plan.

For this reason, agencies and governments should be well-prepared for any potential emergent situation to be able to start assessment activities immediately. Learnings from the past events and the shared experience may accelerate the assessment planning process and improve its quality.

Assessment starts with the evaluation of secondary information (information gathered from external sources such as population data, risk maps, socio-economic indicators, etc.) related with the affected region. This can be obtained from a variety of sources such as government, national statistical organizations, and the media. Generally, collection of secondary information is completed within three days. This data is used for clarifying the situation and preparing the assessment plan.

In the disaster management cycle, needs assessment phase is composed of three stages which differ by the way of information collection techniques according to availability of secondary information. Rapid assessment is performed in the immediate aftermath of the crisis to estimate its scale and scope quickly. Then, the detailed assessment process takes place to verify estimations made in rapid assessment phase with increased volume of information and to catch up changing situations. Finally, during continual assessment, information is continually collected and updated during the response phase. In this study, we focus on a rapid assessment setting, in which agencies evaluate people's needs quickly. In this stage, reliable primary information collection activities are performed by making visits in the affected region. These visits start after secondary information becomes available, and they need to be completed in at most two weeks [4]. The primary data is collected directly from the potential sources such as key informants, community groups or individuals, and is mainly aimed at estimating key information such as the proportion of people who need food, water, medical aid, shelter, etc.

After the collection of primary data in the field, these findings are compiled, validated and conclusions are reported. Compilation and validation process is also a challenging part where cross-check of all secondary and primary data obtained from different sources of information are made. Hence, even in the same disaster, it is possible to make conflicting conclusions regarding the findings. In order to reduce these conflicts, agencies should coordinate and share their resources and experience.

During the rapid assessment phase, there is limited time for assessment activities, so that response operations can start as quickly as possible. This requirement for quick assessment of the situation makes it impossible to survey the whole affected region in order to collect information. Consequently, sampling methods are used to estimate beneficiaries' needs such as the proportion of people who need shelter, number of people who do not have access to water, proportion of females with infants who need human breast milk, etc. In practice, there are some basic systematic techniques, which do not necessarily rely on analytical methods, applied for the sampling in emergent assessment situations such as the Expanded Programme on Immunization (EPI) method developed by the World Food Organization [6]. Additionally, organizations that take part in needs assessment publish some reports and guidelines about their methodology, where practitioners can find process details and tools, especially for rapid assessment activities [2, 4, 6, 8, 7, 9, 20, 50]. These reports are useful guides for the steps needed to apply for future assessment activities.

Since needs assessment involves a sampling process, its effectiveness can be evaluated by two main criteria: (1) how well the sampled data reflects the actual situation (accuracy) and (2) how certain the obtained information is (precision). In the context of needs assessment, the former can be achieved by visiting as many sites as possible, whereas the latter mainly depends on how much time is spent to collect information in each site. When multiple sites are to be visited for needs assessment, there exists an inherent trade-off between the time spent to travel between sites and the time to survey the beneficiaries in each visited site. In other words, if the assessment team spends more time to survey more beneficiaries in each site (thereby improving precision), fewer sites may need to be visited (thereby sacrificing from accuracy), and vice versa. Hence, the trade-off between travel time and sampling time lead to a trade-off between accuracy and precision of the sampling process. This points to the need to develop quantitative models and solution approaches that will result in efficient routing and sampling schemes to balance this trade-off. To the best of our knowledge, despite the abundance of reports and guidelines on the sampling methodologies by the organizations, there exists no study that combines sampling decisions with the routing of the needs assessment teams.

In this study, we define the Post-Disaster Needs Assessment Routing Problem (PDNARP), which is a variation on the selective version of the Generalized Traveling Salesman Problem (GTSP) for the post-disaster needs assessment operations. The main decisions in this problem involve (1) which communities to visit, (2) in which sequence to visit these communities, and (3) how much time to spend to survey each community. The objective function makes use of probability sampling to quantify the accuracy and precision of the estimations. In our problem, time, which is a scarce resource, is consumed by logistics and sampling processes. We also use a clustered network structure, where clusters are determined according to prior knowledge or secondary information about emergency zone. We assume that each cluster involves a set of homogeneous community groups. In the objective function, accuracy is represented by how many clusters are visited, whereas precision depends on how many beneficiaries are surveyed in each cluster. All assessment activities are expected to be finished within a pre-determined time limit. Selective characteristic of the problem prevents the assessment team from visiting all beneficiary groups, as is the case in real assessment operations. Consequently, our aim is to obtain reliable information from as many diverse groups as possible.

We present a mixed integer programming formulation of the PDNARP. Furthermore, for large-scale networks, which mimic real-life cases, we develop a tabu search metaheuristic which is able to solve large scale instances very quickly and which provides near-optimal solutions in most instances. In the computational study, results and the performance analysis for some variants of the proposed algorithm is provided. Finally, we illustrate our metaheuristic on a realistic case study based on the 2011 Van Earthquake.

The remainder of this thesis is structured as follows. In Chapter 2, relevant literature on problem structure and solution methods is reviewed. Problem definition and mathematical model formulation are presented in Chapter 3. Our solution approach is described in Chapter 4. Computational analysis on experimental data set and results on a case study are provided in Chapter 5. Finally, in Chapter 6, we conclude this thesis and discuss future research directions.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a literature review of the related studies is presented. This chapter is divided into three subsections where previous studies on post-disaster needs assessment operations, the selective and generalized versions of the Traveling Salesperson Problem, and tabu-search-based meta-heuristics are reviewed in detail.

2.1 Literature on Post-Disaster Needs Assessment Operations

Although there are many guidelines and technical reports on needs assessment published by various organizations, in the IE/OR literature, this area is not studied as widely as other phases of the disaster management cycle.

Reviews on disaster operations management (DOM) point out that there is an increasing interest on this topic in the last decades. Together with the accelerated increase in the world population and damage on nature, frequency and the consequences of disasters have also risen. Galindo and Batta [33] provide an extensive literature review on the developments in OR/MS research on DOM between 2005 and 2010. This review shows the rising trend for research in humanitarian operations. It is claimed that the successive large-scale catastrophes within a short span of time, namely the World Trade Center attacks (2001), tsunami in the Indian Ocean (2004), Hurricane Katrina (2005) and the Haiti earthquake (2010) have highly increased the interest of researchers in this area. Noticing the potential diminishing impact of disaster management operations in consequences of disasters, studies have sped up for each part of the disaster management cycle.

When IE/OR literature on the response phase is further analyzed, it can be seen that most of the studies are centered upon the relief distribution problems [1, 15, 55, 83]. However, it should be recognized that for successful relief distribution, resource allocation should be made in an efficient way. This is only possible with the reliable information that comes from the needs assessment phase. Despite this fact, for this critical part of the response phase, there are only a few studies in the IE/OR literature.

The study of Malilay et al. [56] proposes a cluster-sampling method for rapid assessment, which is the modification of the Expanded Programme on Immunization (EPI) method proposed by World Health Organization (WHO) [6]. This study differs from the previous methods with the inclusion of the estimation on many parameters such as remaining population, scope of damage, people in specific requirements, estimation of destructed/remaining house units, instead of considering only a single estimation. However, the survey design in this study does not consider the sequencing of the vulnerable beneficiary groups.

To the best of our knowledge, Balcik [14] is the only study on the rapid needs assessment problem that includes both site selection and routing decisions. Here the objective is to maximize the coverage of the community groups' distinct attributes (such as distance to epicenter, elevation, disabled population, female population with children, etc.). This study focuses on purposive sampling in order to satisfy a predetermined level of coverage for each attribute. In the context of this study, coverage is the collected proportion of each attribute. Attributes are observed in more than one site, and each site also has multiple attributes. The coverage of an attribute increases by one in the case that visited site includes this attribute. There exists a necessity of collecting information from each distinct attribute, i.e., it does not involve characteristic selection. Moreover, Balcik [14] assumes that once a site is visited, required information about all attributes of the corresponding site is collected immediately. Thus, this study does not consider the effect of sampling times in each visited site. However, it provides a basis for this thesis with the idea of covering different beneficiary groups in a limited time to make reliable estimations on disaster impacts.

Although there are only a few studies about needs assessment in the IE/OR literature, many booklets and guidelines of the organizations are available [2, 3, 4, 5, 6, 7, 9, 51].

However, these do not make use of mathematical solution approaches to determine the optimal assessment strategies. To the best of our knowledge, there is no such study among these that is statistically supported and that includes routing and selection decisions to assess the trade-off between information accuracy and precision.

2.2 Literature on the Selective and Generalized versions of the Traveling Salesperson Problem

Since the total time for travel between sites and survey of the population is limited in the rapid assessment phase, it may not be possible to visit all sites for assessment. This aspect of the PDNARP makes it similar to the Selective Traveling Salesperson Problem, where only a subset of the cites/nodes can be visited. Furthermore, when the disaster-affected area can be clustered into multiple regions (based on the similarity of sites in terms of their geographical, demographic or economic properties), it may be possible to assess the needs of a cluster by visiting a single site that belongs to it. In this sense, the PDNARP resembles the Generalized Traveling Salesperson Problem. In this section we review the IE/OR literature on the Selective and Generalized versions of the TSP.

Under the general TSP setting, when it is not possible to visit all nodes, visiting each node brings a certain amount of "profit". This version is generally called as Selective TSP/VRP in the IE/OR literature. Laporte and Martello [52] describe the Selective TSP as constructing a route with maximum profit whose length does not exceed a predetermined limit. Therefore, only a subset of all nodes could be visited within predetermined limit. In the study of [52], for the exact solution of this problem, an integer linear program (IP) is proposed. In another study that proposes an alternative IP approach for Selective TSP, an upper bounding scheme is used in order to decrease the size of the problem [57].

The Selective TSP is also referred to as the Orienteering Problem (OP) in the IE/OR literature. Its name comes from the outdoor sport "orienteering". In this game, start and end points are determined, and the aim is to collect the maximum score within fixed amount of time by visiting certain points of interest. The OP arises in variety of applications such as home heating oil delivery [40], athlete recruiting from high schools [25], and routing technicians to service customers [78]. There are many studies which propose optimal solution approaches for the OP. As an exact solution method, Ramesh et al. [69] use Lagrangean relaxation within a branch-and-bound framework, Fischetti and Toth [32] describe a branch-and-bound algorithm, and [23] propose a branch-and-price algorithm. On the other hand, there are also many studies where heuristic/metaheuristic approaches are proposed. For example, Chao et al. [26] propose an effective heuristic, Wang et al. [82] use artificial neural networks, Gendreau et al. [36] present tabu search and [79] use a genetic algorithm as a solution approach. Moreover, time windows [48] and team orienteering problem (TOP) [49, 78] are the most common variations of this problem.

The Prize Collecting Traveling Salesperson Problem (PCTSP) is also a widely-studied version of the Selective TSP in the IE/OR literature. In PCTSP, each node has a prize and penalty, and the objective is finding the shortest tour length while maximizing the difference between the total profit of the visited nodes and total penalty due to unvisited nodes. This problem has a threshold for the total prize, so the solution is feasible only if total prize reaches this threshold. The PCVRP is the variant of this problem with multiple tours. Both problems are NP-hard, and due to the complexity of the PCTSP there are only a few studies where exact solution methods are proposed [12, 32]. On the other hand, heuristic methods are widely used to solve the PCTSP such as a Christofides' algorithm based heuristic [21], a hybrid Lagrangian genetic algorithm [42], and a Lagrangian heuristic [30]. In the literature, directed versions of PCTSP also arise in scheduling problems. For example, Balas [13] presents PCTSP for scheduling the daily operations of a steel rolling mill.

The Profitable Tour Problem (PTP) is the simplified version of the PCTSP which has an objective of minimizing the length of tour plus the sum of penalties of unvisited nodes. The PTP is initially proposed by [31]. Studies on symmetric [21], asymmetric [31, 61] and capacitated [11] versions of the PTP are available in the literature .

The OP, PCTSP and PTP are the variations of the Selective TSP, where the main difference in between is the objective function selection. The PDNARP resembles these problems, if the information collected from visited sites is considered as "profit".

However, in these problems, the profit collected from a node is fixed and time spent in visited nodes is not taken into account. On the other hand, in the PDNARP, "profit" obtained in visited nodes increases if more time is spent in these nodes. Thus, our problem differs from these Selective TSP variants due to the increasing "profit" (information precision) by spending more time for sampling (see Figure 3.8 in Section 3.4.1).

For the PDNARP, if it is possible to assess a cluster by visiting one of its sites (i.e., when clusters are "homogeneous"), the problem is analogous to the Generalized Vehicle Routing/Traveling Salesperson Problem (GVRP/GTSP), which are extensions of their classical counterparts (VRP and TSP), where the aim is to find the shortest route that visits exactly one node (or at least one node) from each cluster. The GVRP/GTSP is a useful framework for a variety of applications such as the TSP with profits, VRP extensions and arc routing problems in the literature [16]. However, this problem is NP-hard. For this reason, heuristic methods are widely studied besides exact methods.

The study of [37] provides an efficient transformation of the GVRP into a Capacitated Arc Routing Problem (CARP), which is also NP-hard, and summarizes the solution approaches available in the literature. Pop et al. [64], Pop and Pop-Sitar [66] and Quttineh [68] propose mixed integer programming models for the GVRP extensions in their studies. Bektaş et al. [19] introduce four different integer programming formulations and propose a branch-and-cut algorithm. Furthermore, a simple metaheuristic called Large Neighborhood Search and a preprocessing algorithm are also presented in this study. Other heuristic methods studied in the literature can be listed as incremental tabu search [59], parallel universes' algorithm and tabu search [60], hybrid ant-based heuristic [62], genetic algorithm [67], nearest neighbor [63], a Clarke-Wright based and local-global heuristic [63] and an iterated-tour partitioning heuristic [74]. The GTSP is analogous to the PDNARP with its clustered structure and the need for selecting one node from each cluster. However, the latter is a selective version of the GTSP due to the time limitation.

Another related problem from the literature is the Clustered VRP (CluVRP) where the aim is to maximize profits from visiting a subset of the predetermined clusters. If a cluster is to be included in the tour of a vehicle, all of its nodes must be visited by the same vehicle [17]. The idea of dividing customers into zones and making routes using that zones in order to decrease complexity is initially presented in the study of [75]. Battara et al. [18] present two exact algorithms for the CluVRP, which are branch-and-cut and branch-and-cut-and-prize. Barthelemy et al. [17] suggest a three-step metaheuristic approach which combines the Clark and Wright heuristic with a Simulated Annealing procedure. Pop and Chira [65] present a hybrid metaheuristic based on a genetic algorithm. Additionally, Vidal et al. [81] describe alternative hybrid metaheuristics that are combinations of iterated local search and genetic algorithm. The CluVRP differs from the PDNARP in terms of its node selection policy.

In all of the Selective TSP, GVRP and CluVRP, time spent in the visited nodes is not considered. In other words, total "prize" of a node is obtained directly when that node is visited. Different from these, the Traveling Purchaser Problem (TPP) aims to collect a predetermined quantity of "items" from a subset of nodes (in each of which the "items" are "priced" differently), and the objective is to minimize total travel and purchasing costs. Laporte et al. [53] describe a branch-and-cut algorithm for the undirected version of the TPP. Riera-Ledesma and Salazar-Gonzales [73] adapt the TPP into a school bus routing problem where a column generation approach is used as the solution method. Ravi and Salman [70] study approximation algorithms for the TPP variants. Angelelli et al. [10] present a greedy heuristic approach for the dynamic variant of the TPP.

Besides the similarities with the problems mentioned above, (i) using time as a resource for both information collection and routing decisions and (ii) usage of the accuracy and precision concepts to evaluate the performance of needs assessment in the PDNARP bring about new perspectives and, to the best of our knowledge, no similar study has been reported in the literature considering both of these aspects.

In the light of these, we can define the PDNARP as a selective version of the GTSP with shared resource used by both information collection and routing decisions. Our problem takes total travel and needs assessment time as a constraint and tries to collect valuable information within the allowable time limit. The PDNARP has two dependent decisions; how much information to collect from the visited areas and in what sequence to visit them. The proposed model for this problem aims to provide both accurate and precise information about the effected region. According to findings from the previous needs assessment reports, our primary objective is to visit as many clusters as possible. Then, secondarily, the model tries to maximize the overall precision level of the information from each visited cluster. This purpose directs us to a lexicographic approach for these two objectives.

2.3 Literature on Tabu Search-Based Heuristics

Tabu Search (TS) is one of the most popular metaheuristics studied in the IE/OR literature. In order to solve the large sized instances, TS was adapted to the Post-Disaster Needs Assessment Routing Problem. TS was initially proposed by [38] and later improved by [41]. This metaheuristic includes basic local search moves and uses these moves in a systematic manner. TS has the purpose of performing an efficient search in the solution space by using adaptive memory and avoiding cycling, which causes being stuck into local optimal solutions.

Glover [39] states the main reason of the popularity of TS as the impressive practical success of this method in applications. Scheduling, telecommunication, design, production, inventory management, routing and graph optimization are some of these application areas. In the study of [44], TS is used to solve the graph coloring problem. This method is also adapted to the economic dispatching optimization problem by [54]. Additionally, there are studies in the literature where TS is used for solving the capacitor placement problem in a radial distribution system [46] and where it is used for the political districting problem [24].

Even though TS has a wide range of application areas, routing is where it is most highly used. There are many studies of TSP/VRP variants where TS is adapted such as the TSP/VRP with capacity restrictions [34], soft time windows [77], periodic and multi-depot versions [28], stochastic demands and customers [35]. Moccia et al. [59] and Navidadham et al. [60] apply TS for the Generalized Vehicle Routing Problem (GVRP) and point out the effectiveness of the method in their studies.

The original version of the TS method is deterministic, i.e, it does not contain randomness. This is the main feature which makes this metaheuristic different from simulated annealing and genetic algorithm, which are other well-known metaheuristics used in optimization problems. Even though many traditional versions are fully deterministic, there are some other studies where randomness may be included in procedure, such as random restart [29], random tabu tenure selection [24], and random moves [45].

An important characteristic of TS method is the systematic use of *memory*. The role of this memory is to restrict the choice within the neighborhood of the current solution. *Neighborhood* is the candidate list of solutions that are reachable in one move. Neighborhood in TS is dynamic and changes according to defined rules. Two types of memory is recorded in TS. *Recency-based (short term) memory* keeps the recently visited solutions as inactive to avoid cycling, whereas *frequency-based (long term) memory* records the number of iterations that a solution attribute used during the search process.

In the tabu search framework, *solution attributes* are the elements that change in moving from current solution to another solution. If a solution attribute belongs in recently selected solution, it becomes *tabu-active*. Solutions that includes tabu-active attributes in the neighborhood of a solution are called as *tabu*, and they are removed from neighborhood. Tabu-active solution attributes are kept in a *tabu list* until a predetermined number of iterations, called *tabu tenure*. *Tabu status* indicates whether an attribute is active or inactive. When it turns into *tabu-inactive*, this attribute is excluded from tabu list and included in the neighborhood of the solution again. *Aspiration criteria* are used to override the tabu status of a tabu solution in the case of becoming the new incumbent solution.

Use of both *intensification* and *diversification* techniques in this framework increases the possibility of finding the global optimum. *Intensification* is a form of exploitation which involves searching an attractive region in depth to find a good solution. By changing rules directing solution to better moves is desired. On the other hand, *diversification* is a form of exploration which involves moving into farther regions of the search space better to possibly attain new promising solutions.

CHAPTER 3

SYSTEM DESCRIPTION AND PROBLEM DEFINITION

In this chapter, the Post-Disaster Needs Assessment Routing Problem is formally defined and a mathematical model for the problem is presented, along with a preprocessing method.

3.1 Needs Assessment

A disaster is an immediate catastrophic event that causes fatalities, vulnerabilities, economic and material losses where society is not able to cope with the impacts of that event. The International Federation of Red Cross and Red Crescent Societies (IFRC) defines a disaster as the combination of a hazard with vulnerability and the inability to decrease or decelerate the destructive effects of it [4]. Statistics point out the devastating effects of disasters clearly. According to statistics provided by Munich Reinsurance Company, the deadliest disaster in between 1980 and 2017 is recorded as tsunami struck in Thailand in 2004 where approximately 220,000 people died [72]. The largest economic damage in between 1980 and 2017 was recorded as 210 billion dollars [71], which was incurred after the Japan Earthquake and Tsunami in 2011 (see Figure 3.1 and Figure 3.2).

The world has suffered through irrepressible catastrophic disasters since its formation, and the increase in human population and improper use of natural resources increase the frequency of these events. Classification of these disasters can be made according to various aspects: according to their cause (natural or man-made/technological), according to disaster timing (sudden-onset or slow-onset), or according to predictability

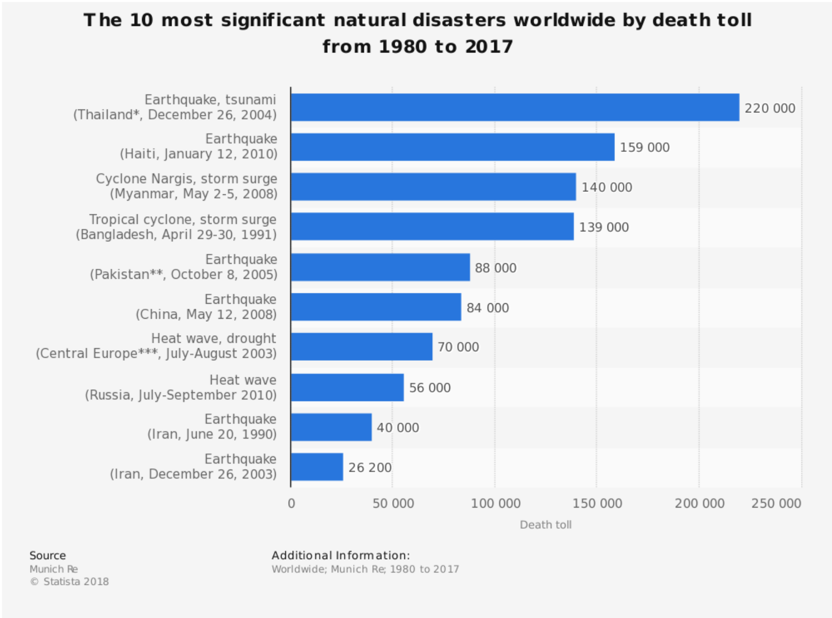

Figure 3.1: Most significant disasters according to mortality rate in between 1980 and 2017

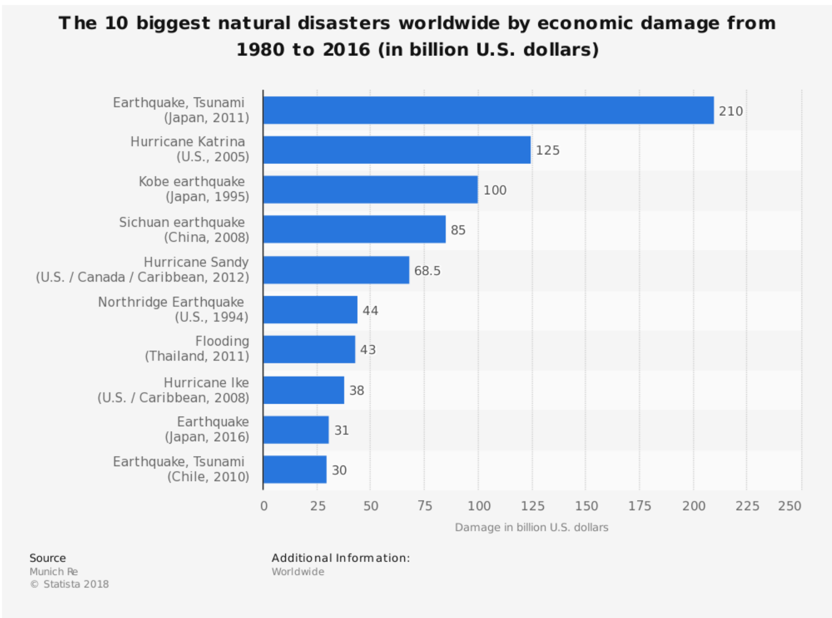

Figure 3.2: Most significant disasters according to economic damage in between 1980 and 2017

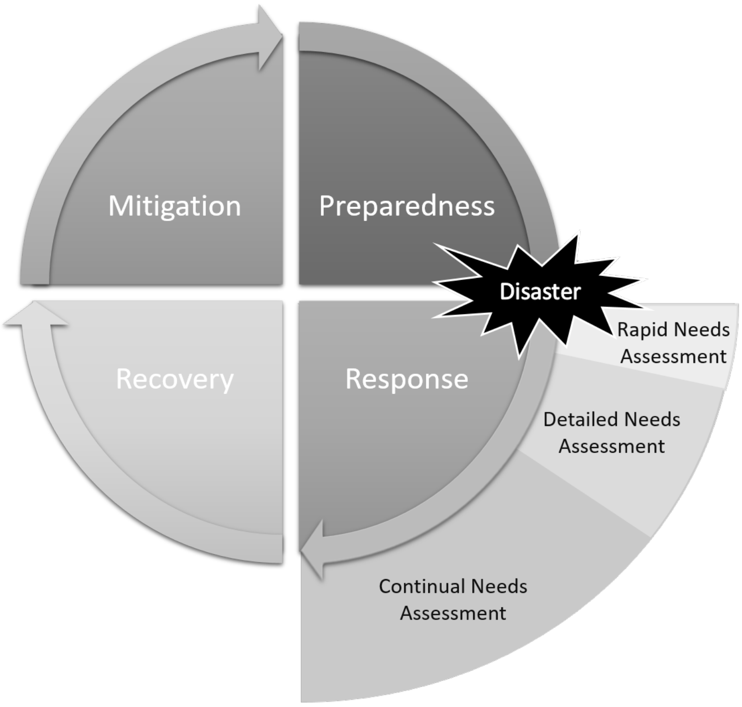

Figure 3.3: Disaster Management Cycle

due to their locations and timing.

Disaster management is considered as a cyclic timeline composed of four main phases, which are mitigation, preparedness, response and recovery (see Figure 3.3).

Mitigation activities aim to reduce the effects of disasters by eliminating or diminishing the catastrophic outcomes. Effectiveness of mitigation processes depends on integrating proper measures in planning. Preparedness activities are performed in order to achieve the desired level of readiness to respond to sudden crises successfully. In this phase, the aim is to prepare the resources and to improve capacities for an efficient response rather than to prevent the occurrence of the emergency. Response activities begin in the immediate aftermath of a disaster. Humanitarian organizations actively take place in this phase in order to assess needs, distribute relief, maintain life, and improve the health conditions and morale of beneficiaries. Primary focus of response activities is to supply the basic needs of victims and to quickly stabilize the emergency situation. Success of the mitigation and preparedness phases improves the effectiveness of response. Recovery is the longest phase of this cycle, which aims at the restoration of the society and environment. This phase starts when the crisis is brought under control and continues until the affected region returns to (at least some form of) the pre-disaster state.

In the disaster management cycle, needs assessment activities take place immediately after a crisis hits a region and before response activities begin. Many primary characteristics of the humanitarian emergencies are identified via assessment such as magnitude, influenced areas and needs of beneficiaries. Common needs required after a disaster may include food, shelter, medical care, essential items (blankets, heaters, water containers etc.), safe drink water, sanitation and waste disposal, income, protection, and psychological support. Needs assessment is performed in order to provide an efficient relief distribution by identifying the requirement of these needs. It is an inevitable process due to unpredictable nature of the disasters.

Most of the time, experience from previous disasters may not be used to predict the impact of a new event, since each disaster is unique in terms of certain attributes such as location, magnitude, population characteristics, etc. For this reason, estimations made based on the previous disasters may misdirect the subsequent relief distribution operations. Thus, organizations should apply separate assessment plans in the aftermath of each disaster in order to investigate the needs of beneficiaries in the affected area. For an efficient relief distribution, collected information should represent reality as much as possible.

In practice, there are three types of needs assessment, which are rapid, detailed and continual [4]. For all of these, the main principle of the assessment is identification of vulnerabilities and capacities. However, they differ in terms of how and when information is collected. Needs assessment timeline is provided in Figure 3.4.

In the immediate aftermath of a disaster, *rapid needs assessment* is performed in order to gather quick information from the emergency zone. This phase takes place before preparation of the response plan. Effective rapid assessment is essential to determine the scale and scope of the response and prioritize resources properly before relief distribution. This phase takes approximately one or two weeks. There is limited access to the area, thus, it may not be possible to visit all sites and individuals in the region. Hence, within this short time period, there is limited access to information resources. Data from the government, NGOs, health services, as well as information collected

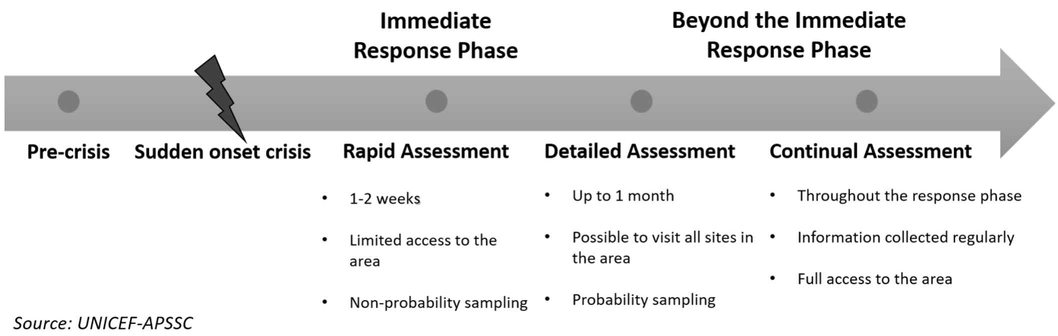

Figure 3.4: Needs Assessment Timeline

from a limited sample of beneficiaries may be used. In this stage, assumptions made according to past events are very important and making use of experienced teams improves the quality of the operation. Non-probability sampling methods are more appropriate due to insufficient data availability.

Detailed needs assessment phase takes place after rapid needs assessment is completed. Information volume increases and inferences are closer to reality as opposed to the previous phase. A detailed assessment reveals if more information is required to take action and/or if there is a possibility of change in the situation. An approximate time frame for detailed assessment is one month; however, this time interval may be shorter or longer according to complexity of the event and resource availability. In this phase it may be possible to visit all sites in the affected area and survey the required number of beneficiaries to apply probability sampling methods.

After completion of detailed assessment phase, response teams start distribution operations in the emergency zone. At this point, *continual assessment* is performed simultaneously and during the whole operational period, assessment teams provide updates in information through regular collection of data. In this stage, there is a full access to beneficiaries and other sources of information.

3.2 Information Collection for Needs Assessment

Needs assessment is conducted in chaotic emergency situations where there are resource restrictions, accessibility and security problems. Against all odds, collected information in assessment phase should reflect reality.

In the Assessment Toolkit presented by [50], key points of data collection are summarized. Accordingly collected data should be useful for decision making, feasible to collect, reliable, complete, worth the cost, timely, and triangulated.

During the assessment, both qualitative and quantitative data can be gathered. The questions prepared for the assessment process and the method of information collection may change as the assessment unfolds. Using quantitative data allows for more accurate and precise estimations on the impact of the catastrophic event and requirements of beneficiaries. Scientific measurement is stated as a key aspect of quantitative research since it provides more reliable findings from accurate and precise analyses [7].

Before starting the assessment, secondary information is collected from sources such as the government, national statistical organizations, media and so on. Risk maps, population statistics, socio-economic indicators are used as secondary information used to clarify the situation before planning the assessment. In general, secondary information collection is completed in three days. Then the assessor agencies prepare an assessment plan according to their capacities and resources. After these, the agencies become ready to collect primary data in the emergency zone.

Building an assessment plan for each disaster requires some setup time, which curtails the time for survey in the zone. For this reason, agencies and governments should be well-prepared for any potential emergency situation to be able to start assessment activities immediately.

Experiences and learnings from the past events may accelerate the assessment process and improve the quality of the plan. After collection of primary data in the field, these findings are compiled and validated in order to cross-check all secondary and primary data obtained from different sources, which also requires remarkable effort.

In a report by [6], it is mentioned that after the 2007 earthquake and tsunami in the Solomon Islands, a variety of organizations performed their own assessment plans and the evaluation of these points out the high inconsistency among information collected by these organizations. This example points out the possibility of making conflict-

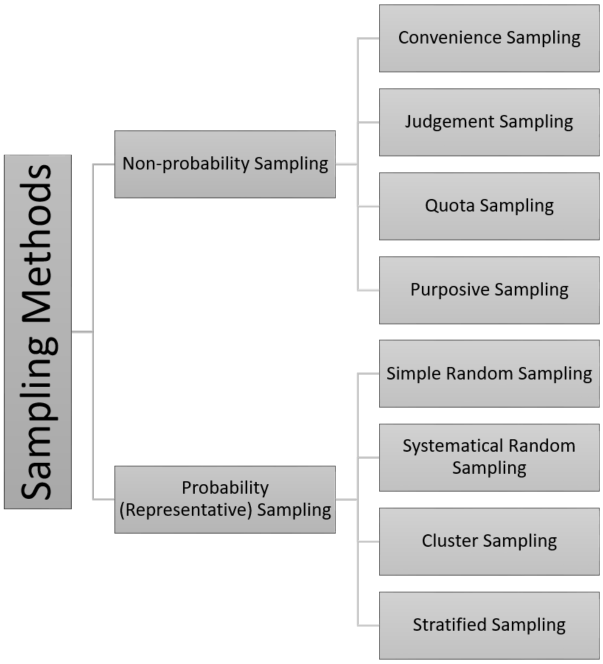

Figure 3.5: Sampling Methods

ing conclusions regarding the findings. In order to reduce these conflicts, agencies can coordinate and share their resources and experiences. Additionally, practitioners can find the details of assessment methodology in the reports and guidelines of the organizations [2, 6, 9, 20].

3.2.1 Sampling Techniques

Rapid needs assessment activities are required to be completed within a limited time interval. At this critical phase, since visiting the whole region is not possible, applying a proper sampling method for data collection is an inevitable result.

In the technical reports of humanitarian aid organizations, suitable sampling methods are defined for each phase of needs assessment. These are summarized in Figure 3.5. The main distinction between these methods is whether statistical methods are used for sampling (probability sampling) or not (non-probability sampling).

In the case where information and accessibility are not adequate to perform sampling according to statistical methods, non-probability sampling methods are applied. These methods are mainly based on the strategies of experienced professionals according to available information, which is very limited and uncertain. These methods can provide useful results under the right conditions. However, representativeness of estimates is not guaranteed as it is in probability sampling. Some of the well-known non-probability sampling types are convenience, judgement, quota and purposive sampling.

In *convenience sampling*, easily reachable individuals are selected directly. Unfortunately, this selection results in a biased estimation most of the time and it is not recommended for assessment operations if other methods are available.

Judgement sampling is also a biased method where professionals decide on the individual selection based on their knowledge and professional judgment. This method provides good estimates only if an expert is the only person who can decide on the most representative sample.

Quota sampling method requires the fulfillment of a predetermined quota level for mutually exclusive subgroups obtained according to population characteristics such as gender, education level, income level and so on. Sample selection is not random, and hence the sample may not be representative, which may result in sampling bias.

Purposive sampling is the most preferred non-probability sampling method used in rapid needs assessment [6]. Sample selection is performed according to any accessible data which can be proximity to epicenter, accessibility of sites or population characteristics such as size, gender or income level.

Probability sampling methods ensure a statistical level of confidence if the same method is applied repeatedly to the same population. This provides more reliable estimate of needs compared to non-probability sampling. Simple random sampling, systematic random sampling, cluster sampling and stratified sampling are well-known types of probability sampling used in needs assessment.

Simple random sampling (SRS) is the most basic probability sampling method. This is an easy method since samples are directly selected from total population randomly without dividing it into smaller groups. Besides its practicality and simplicity, it has the advantage of being easy to evaluate in terms of the accuracy and precision of estimations from the sample. However, the common drawback in SRS applications the representativeness of data for whole population (possibility of sampling bias).

Systematic random sampling is an alternative strategy where sample selection is performed according to a predetermined method after first random selection. An example method could be listing all individuals in the population and selecting the ones whose index numbers are divisible by five. This method gives better estimations than SRS only if individuals in the population are listed properly considering the population and region characteristics. However, in this method, sample selection is not fully random, so accuracy and precision of estimates can not be calculated as in SRS.

In *stratified sampling*, the population is divided into strata according to determined characteristics (such as population size, proximity to epicenter, gender, age, education level, etc.). Then, random sampling is applied in each stratum separately. For the final selection in a stratum, SRS or systematic sampling can be used. This method provides better use of scarce assessment resources.

Cluster sampling is the most commonly used probability sampling method in practice. It is composed of two stages. In the first stage, the region is divided into smaller units. In each unit, random clusters are formed according to a predetermined size of each unit. As an example, if the small units represent villages, clusters are the districts of each village. In the second stage, individuals are selected randomly in each cluster. SRS or systematic random sampling may be applied in this final stage.

Applying a probability sampling method can be conveniently applied to make estimations on the impact level of disaster in the affected region. Although in technical reports it is stated that non-probability sampling methods are more applicable in the rapid assessment phase, assessment organizations should make an effort on the collection of secondary information as quickly as possible in order to apply probability sampling techniques in assessment operations.

3.2.2 Measuring the Quality of Sampling: Accuracy and Precision

In the context of needs assessment, we define "prevalence" as the proportion of a particular population affected by the disaster or the proportion of beneficiaries in requirement of the common types of needs described in §3.1. Throughout this thesis, needs estimation aims to end up with estimates of the prevalence in different sites of the disaster region.

The quality of statistical sample is generally determined by its *accuracy* and *precision*. Here, accuracy refers to how close an estimated prevalence is to the true value, measured by the difference of the sample average and the actual mean of the population. Precision is the level of certainty of the estimations, generally measured by the width of the confidence interval for the estimated parameter.

A target board example can be used in order to clarify the terms accuracy and precision. One example figure can be seen in Figure 3.6. The desired outcome is represented in board (a) where the observations are close to center and also each other.

From the perspective of post-disaster needs assessment, accurate information can be obtained by visiting as many affected sites with distinct characteristics as possible. However, an accurate estimation is not enough for a reliable estimation. In each site, as many beneficiaries should be surveyed as possible to reach the desired level of precision. In practice, making the prevalence estimation quickly is critical for the quick start for the relief distribution operations, since time is also a limited resource for the relief distribution phase. Due to this time limitation, relief assessment may not be able to reach desired levels of accuracy and precision at the same time. Hence, the decisions of which sites to visit (for accuracy) and how many beneficiaries to survey in each site (for precision) are the important decisions in needs assessment.

3.2.3 Clustering of Sites

The most popular and effective probability sampling method for needs assessment is cluster sampling. Furthermore, [6] suggests stratifying the affected region into clusters according to socio-economic or demographical characteristics and visiting diverse sites in order to capture the dissimilarities on the impact of the disaster. Clustering is done based on the concept of *homogeneity*, which can be defined as the level of similarity among a group of sites according to prevalence levels. In contrast, the term *heterogeneity (non-homogeneity)* is the distinctness of the sites.

There may be two extreme cases about homogeneity. If all sites are homogeneous,

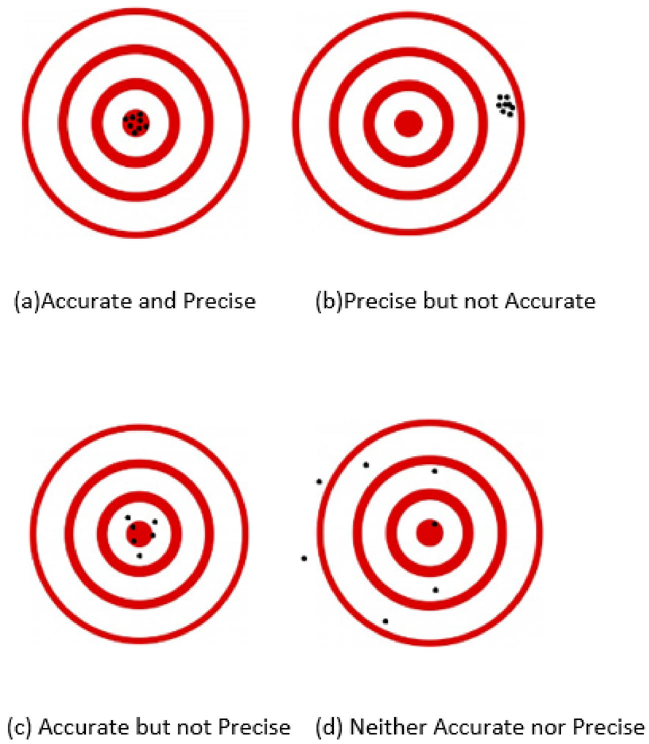

Figure 3.6: Target board that represents accuracy and precision

it is enough to visit only one site to collect information. On the other hand, if all nodes are heterogeneous, all sites should be visited to collect information under the assumption that time is enough to visit all sites. In the real case it is not possible to divide whole region into exactly homogeneous or heterogeneous beneficiary groups. The importance of the pre-knowledge on homogeneity takes place at this point since this information can be used to prevent visiting unnecessary/similar sites.

Without loss of generality, clusters are considered as mutually exclusive and collectively exhaustive. In this setting, each homogeneous subset of nodes constitutes a cluster and clusters are heterogeneous beneficiary groups between themselves; in other words, a cluster includes similar beneficiary groups in terms of predicted prevalence rates.

3.3 Sampling in Post-Disaster Needs Assessment Practice

Technical reports of organizations point out the use of non-probability sampling approaches in the practice of rapid needs assessment. In these guidelines, certain basic rules are proposed for sampling. As an example, Binkin et al. [22] suggests a design of 30x30 (which means 30 clusters to visit and 30 households to survey in each cluster). It is stated that this amount provides a reliable estimate on the prevalence of malnutrition. On the other hand, another study on sampling design points out that by increasing the number of clusters and decreasing the number of observations in each, it is possible to reach the same precision level [5]. In addition to these studies, WHO makes use of a tool named the "Expanded Programme on Immunization" (EPI), which is one of the best-known rapid assessment survey tools developed for immunization surveys [43]. In this method, 30 clusters are determined with a probability proportional to their population sizes and seven households are visited from each. Household selection is performed randomly by selecting a random direction and visiting the nearest seven households in that direction [8].

Sampling approaches in needs assessment practice have some drawbacks. Firstly, they do not consider the locations and population sizes of the clusters and sites. Furthermore, the time restriction in the rapid assessment phase may not allow making the decided amount of observations. Visiting a fixed and equal number of beneficiaries in each cluster does not provide same level of precision for all. Moreover, the question of which cluster can be ignored if there is not enough time should be evaluated according to the locations of the clusters. Such drawbacks indicate the need for quantitative decision making mechanisms to tackle these operational issues during needs assessment, which provides the main motivation for defining the Post-Disaster Needs Assessment Routing Problem.

3.4 The Post-Disaster Needs Assessment Routing Problem

The need for quantitative decision making tools for probability sampling during the rapid assessment process is underlined in the reports of relevant organizations because of two valuable properties of statistical methods:

1. It is possible to obtain reliable information from a representative sample of the target beneficiary group,
2. It is possible to calculate the sampling error and assess reliability of the survey estimates.

Rapid needs assessment activities need to be performed efficiently within a limited period of time. Due to time required for traveling among sites, less time is available for sampling. Hence, these activities compete for the same scarce resource. Furthermore, designing an inefficient route not only prevents the visiting of more sites (thus decreasing accuracy), it also leads to less time to survey the beneficiaries (thereby reducing the precision). Motivated by the importance of the relationship between routing and sampling, the Post-Disaster Needs Assessment Routing Problem (PDNARP) aims to find a balance between these decisions to maximize the effectiveness of sampling.

In the PDNARP, the goal is to provide a statistically supported assessment strategy which includes both sampling and routing in order to meet the requirements of a reliable sampling strategy. There are a number of questions that need to be answered in order to establish systematic strategies in emergency cases. Such critical questions are as follows;

- From which clusters and sites should information be collected?
- How many beneficiaries should be visited in each site?
- In what sequence should the sites be visited?
- Which factors affect the selection of which sites to visit?

3.4.1 Measuring Accuracy and Precision in the Needs Assessment Routing Problem

In order to gather a high-quality sample, the collected data must lead to an accurate and precise sample. Since assessment teams are not able to collect data from all individuals in the emergency zone, they need to allocate time effectively.

In the PDNARP, accuracy refers to the closeness of the estimate to true value of prevalence. Clustered structure of the network ensures similarity of beneficiary groups within the cluster in terms of needs. Therefore, diverse beneficiary groups can be discovered by visiting as many clusters as possible. Hence, we define the accuracy measure as the number of clusters visited. In other words, as the number of clusters visited increases, the accuracy of the estimate also increases. However, accurate information requires spending more time in traveling in order to visit more places. As a consequence of the need to visit as many sites as possible, remaining time for sampling in each site decreases.

Sample size is the required number of individuals that ensures a desired precision level for the corresponding population. For the calculation of sample size, it is assumed that assessment will be performed in a finite population (without replacement). In PDNARP, sites are regions (such as districts, counties, villages etc.) whose population size is more than 100 in general. Thus, we assumed that population size of each site is at least 100. Therefore, due to the Central Limit Theorem, normality assumption holds for prevalence estimates. A sample with size n will be selected from each region with population size N. $\hat{p}$ is the initial estimate on prevalence. α level is used in the sample size calculations for the desired confidence level of $(1-\alpha)$. Furthermore, ϵ represents the margin of error which is the desired precision level (the smaller margin of error is, the better the precision level). Sample size calculation can be made from [27]:

$$n = \frac{z^2_{\alpha/2}\frac{\hat{p}(1-\hat{p})}{\epsilon^2}}{\frac{N-1}{N} + z^2_{\alpha/2}\frac{\hat{p}(1-\hat{p})}{N\epsilon^2}} \tag{3.1}$$

where $z^2_{\alpha/2}\frac{\hat{p}(1-\hat{p})}{\epsilon^2}$ is sample size required for infinite population and n is the sample size for estimating population proportion of a finite population.

If assessment teams are not able to give an initial estimate on prevalence, taking $\hat{p}$ as estimated prevalence value of 0.5 is the most risk-averse approach, since this estimation gives the upper bound for the sample size $\hat{p}(1-\hat{p})$ is maximum when $\hat{p} = 0.5$). According to the relation between population and sample size in Figure 3.7, it can be concluded that after some point population size does not affect the required sample size, population can be considered as infinite.

In Figure 3.8, the relation between sample size and margin of error can be seen. This calculation is made by taking constant population size of 1,000 and 95% confidence

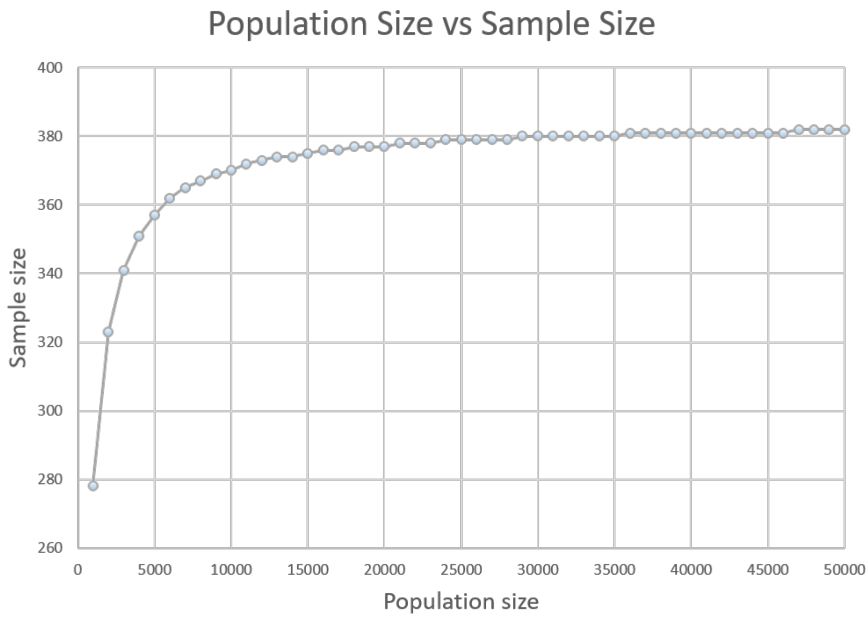

Figure 3.7: Change in sample size with respect to population size ($\hat{p} = 0.5$, $\alpha = 0.05$, $\epsilon = 0.05$)

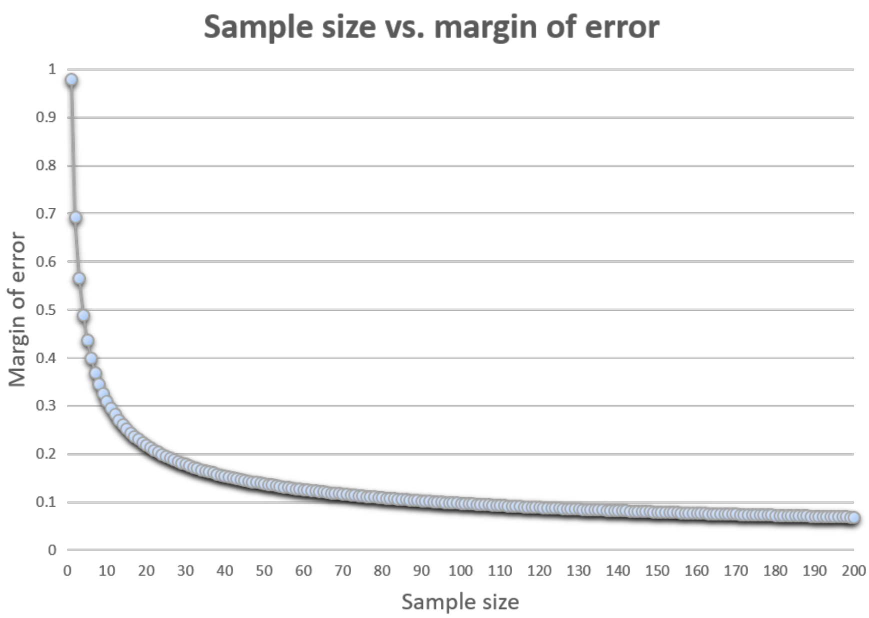

Figure 3.8: Change in margin of error with respect to sample size ($\hat{p} = 0.5$, $\alpha = 0.05$, $\epsilon = 0.05$)

interval (CI). There is an inverse relationship between the margin of error and precision. Therefore, it can be interpreted that information precision increases by the sample size, and the relation between the precision of information and number of individuals visited is concave (that is, diminishing returns exist).

Sample size selection is critical for estimating the prevalence value because a survey performed without calculating the required sample size may lead to biased results or waste of resources due to insufficient information.

Depending on the influence area of the disaster, available time may not be enough to survey the calculated number of individuals. At this point, a tradeoff occurs between number of beneficiaries to be surveyed and number of sites to be visited. This can be interpreted as a tradeoff between more precise or more accurate estimation. In order to survey a lot of individuals in one site, assessment teams may prefer to visit a far-away site with any individuals to survey or more than one site with only few individuals to survey. So, knowing the sufficient number of individuals provides a better decision how much time to spend for survey.

3.4.2 Network Characteristics

When a disaster hits some region, besides attributes of the devastating event such as magnitude, scope and duration, site characteristics also have a significant impact on outcomes. These characteristics can be listed as follows;

- Proximity to epicenter
- Demographical features
- Geographical features
- Education level
- Income level

These features are obtained by the secondary information which can be collected in a short time period easily from the sources of information such as government, national statistical centers, NGOs and so on. If this collected data can be combined with the preliminary information (type, predicted magnitude, duration, etc.) about the disaster,

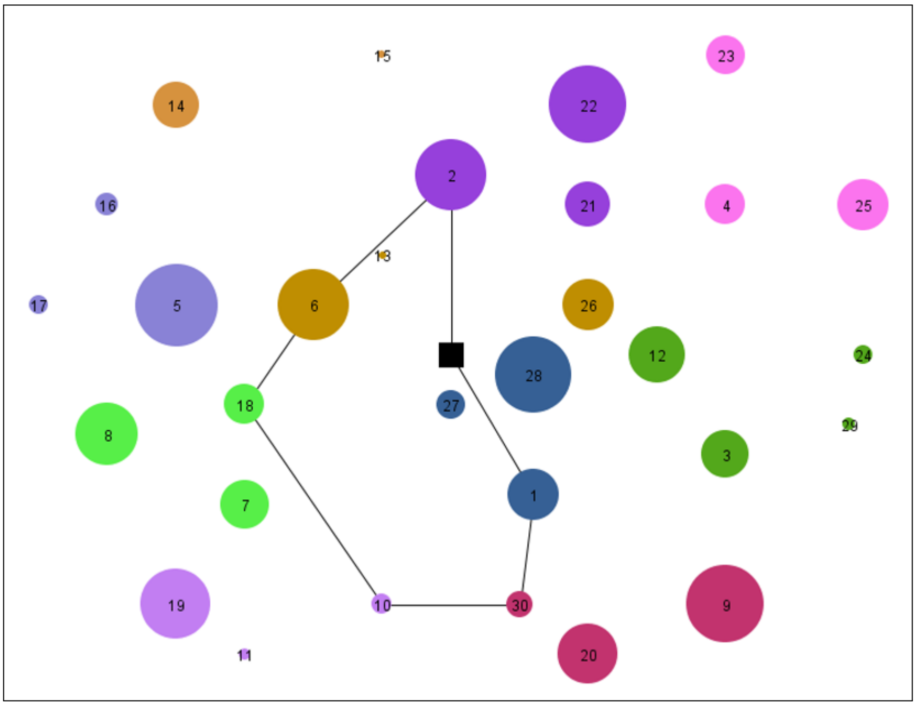

Figure 3.9: Network representation and a feasible PDNARP solution for a geographically clustered 30-node 10-cluster network

proximity of impact levels for the subsets of whole region can be estimated. Making good use of this information provides a good start for the assessment process.

A feasible representative solution of the Post-Disaster Needs Assessment Problem for the network with 30 nodes and 10 clusters is provided in Figure 3.9. Nodes and edges are the members of the indicated network. Nodes represent the sites in the affected region. Population size of each site is demonstrated by the size of that node, and colors of nodes reflect the clusters where the members of the same cluster are indicated with same color. Nodes are connected by edges. Routing starts from the depot and finishes at the depot, which is indicated by a black square in the network.

3.4.3 Assumptions

In order to simplify the problem settings, reduce the complexity and specify the boundaries, the following assumptions are made about the problem:

- Clusters are mutually exclusive and collectively exhaustive.

- Nodes in the same cluster are identical in terms of needs (homogeneous clusters). Variability within clusters is assumed to be insignificant.
- Each site in a cluster is homogeneous within itself.
- Estimated prevalence value is taken as 0.5 since we consider no pre-information.
- After site selection and sample size decisions, we do not consider how sampling within sites is performed. It is assumed that simple or systematic random sampling techniques may be applied.
- For precision, a discrete set of how many beneficiaries to survey is used, leading to a discrete set of possible confidence interval widths. The members of this set are called *options*.
- Each option is admissible for the decision maker.

3.4.4 Mathematical Model

3.4.4.1 Notation

Sets

$\mathcal{N}$: set of nodes (1,..., *n*)

$\mathcal{N}_0$: set of nodes including the depot (0, 1,..., *n*)

$\mathcal{K}$: set of discrete precision options (1, 2,..., *k*)

$\mathcal{C}$: set of clusters ($\mathcal{C}_0$, $\mathcal{C}_1$,..., $\mathcal{C}_m$)

$\alpha(i)$: index of the cluster that node $i \in \mathcal{N}$ belongs to

Parameters

t_{ij}: travel time from node $i \in \mathcal{N}_0$ to node $j \in \mathcal{N}_0$ where $\alpha(i) \neq \alpha(j)$

h_{ik}: precision level (margin of error) for node i with finite population if option $k \in \mathcal{K}$ is selected

n_{ik}: sample size requirement of node $i \in \mathcal{N}_0$ for option $k \in \mathcal{K}$

β: unit sampling (information collection) time

$\mathcal{T}_{max}$: maximum time to complete assessment activities

w : weight of the precision objective ($\mathcal{Z}_2$)

Decision variables

x_{ik} : 1, if option $k \in \mathcal{K}$ is selected for node $i \in \mathcal{N}_0$; 0, otherwise

y_{ij} : 1, if assessment team goes from node $i \in \mathcal{N}_0$ to node $j \in \mathcal{N}_0$; 0, otherwise

p_i: precision option selected for node $i \in \mathcal{N}_0$

$\gamma_{\mathcal{C}_{\alpha(i)}}$: precision level associated with cluster $C_{\alpha(i)} \in \mathcal{C}$ where $i \in \mathcal{N}_0$

$\mathcal{Z}_1$: accuracy objective which denotes the proportion of selected clusters

$\mathcal{Z}_2$: precision objective which denotes the average half width of all clusters $\in \mathcal{C}$

Auxiliary variables

u_i: auxiliary variable for $i \in \mathcal{N}_0$ to define subtour elimination constraint

3.4.4.2 Formulation

$$Maximize \quad \mathcal{Z} = \mathcal{Z}_1 - w\mathcal{Z}_2 \tag{3.2}$$

s.t.

$$\mathcal{Z}_1 = (\sum_{i\in\mathcal{N}_0}\sum_{k\in\mathcal{K}} x_{ik})/|\mathcal{C}|, \tag{3.3}$$

$$\mathcal{Z}_2 = (\sum_{\mathcal{C}_{\alpha(i)}\in\mathcal{C}} \gamma_{\mathcal{C}_{\alpha(i)}})/|\mathcal{C}|, \tag{3.4}$$

$$p_i = \sum_{k\in\mathcal{K}}(1 - x_{ik})0.5 + \sum_{k\in\mathcal{K}} h_{ik}x_{ik} \qquad \forall i \in \mathcal{N}_0, \tag{3.5}$$

$$\gamma_{\mathcal{C}_{\alpha(i)}} = \sum_{i\in\mathcal{C}_{\alpha(i)}} p_i \qquad \forall \mathcal{C}_{\alpha(i)} \in \mathcal{C}, \tag{3.6}$$

$$\sum_{k\in\mathcal{K}} x_{ik} \leq 1 \qquad \forall i \in \mathcal{N}_0, \tag{3.7}$$

$$\sum_{k\in\mathcal{K}} x_{ik} = \sum_{j\in\mathcal{N}_0,\ \alpha(i)\neq\alpha(j)} y_{ij} \qquad \forall i \in \mathcal{N}_0, \tag{3.8}$$

$$\sum_{k\in\mathcal{K}} x_{ik} = \sum_{j\in\mathcal{N}_0,\ \alpha(i)\neq\alpha(j)} y_{ji} \qquad \forall i \in \mathcal{N}_0, \tag{3.9}$$

$$\sum_{i\in\mathcal{N}_0} y_{0i} = 1, \tag{3.10}$$

$$\sum_{i\in\mathcal{N}_0} y_{i0} = 1, \tag{3.11}$$

$$\sum_{i\in\mathcal{N}_0}\sum_{k\in\mathcal{K}} \beta n_{ik}x_{ik} + \sum_{i\in\mathcal{N}_0}\sum_{j\in\mathcal{N}_0} y_{ij}t_{ij} \leq \mathcal{T}_{max}, \tag{3.12}$$

$$u_i - u_j + |\mathcal{N}+1|y_{ij} \leq |\mathcal{N}| \qquad \forall i \in \mathcal{N}_0, j \in \mathcal{N}_0, \alpha(i) \neq \alpha(j), \tag{3.13}$$

$$2 \leq u_i \leq |\mathcal{N}+1| \qquad \forall i \in \mathcal{N}_0, \tag{3.14}$$

$$\sum_{j\in\mathcal{C}_{\alpha(i)}}\sum_{k\in\mathcal{K}} x_{jk} \leq 1 \qquad \forall i \in \mathcal{N}_0, \mathcal{C}_{\alpha(i)} \in \mathcal{C}, \tag{3.15}$$

$$u_i, p_i \geq 0 \qquad \forall i \in \mathcal{N}, \quad (3.16)$$

$$\gamma_{\mathcal{C}_{\alpha(i)}} \geq 0 \qquad \forall i \in \mathcal{N}_0, \mathcal{C}_{\alpha(i)} \in \mathcal{C}, \quad (3.17)$$

$$x_{ik} \in \{0, 1\} \qquad \forall i \in \mathcal{N}_0, k \in \mathcal{K}, \quad (3.18)$$

$$y_{ij} \in \{0, 1\} \qquad \forall i \in \mathcal{N}_0, j \in \mathcal{N}_0, \quad (3.19)$$

(3.2)-(3.19) is a mixed integer program with the weighted combination of precision and accuracy objectives. We give priority to the accuracy objective. If accuracy is the single objective of our problem, model selects maximum number of nodes within allowable time. However, in this case, the model would not try to maximize average precision level. On the other hand, if precision is the single objective, although nodes are selected with the highest precision levels, number of selected nodes is smaller than the maximum possible selection. This is also an undesirable solution, since it may be possible to visit one more node by decreasing the precision levels on each selected node. All precision options are already acceptable for the agencies and selecting more nodes is more desirable solution for the PDNARP. Thus, maximizing the number of selected nodes becomes the primary objective of this problem, since this selection already ensures the acceptable worst precision level for each selected node. Secondary objective (precision), is added to the objective function with a small coefficient (w) in order to allocate remaining time to the selected nodes efficiently (i.e., minimizing the average half width value to ensure best precision level).

Constraint set (3.5) calculates the half width value for the confidence interval of each node. Constraint set (3.6) assigns the half width value of selected node to the corresponding cluster. Constraint set (3.7) prevents assigning one node to multiple sample size options. Constraint sets (3.8), (3.9) are the flow constraints which ensure that an arc enters and leaves each selected node. Constraints (3.10) and (3.11) limit the number of routes as one. Time restriction is provided in constraint (3.12). Constraint sets (3.13) and (3.14) are the subtour elimination constraints proposed in [58]. Selecting only one node per cluster is imposed by (3.15). Constraints (3.16), (3.17), (3.18) and (3.19) are the sign constraints.

CHAPTER 4

A MULTI-START TABU SEARCH HEURISTIC FOR THE POST-DISASTER NEEDS ASSESSMENT ROUTING PROBLEM

The PDNARP is NP-hard, as it includes the orienteering problem as a special case (when there is a single option in each node, and when all clusters have a single node). Hence, realistic instances of this problem are not solvable within a reasonable time interval. In order to solve larger instances, we propose a heuristic method based on tabu search, which is an effective and popular metaheuristic, particularly for routing applications in the IE/OR literature.

The approach developed to solve the PDNARP is a Multi-Start Tabu Search Heuristic (MSTS). It is composed of preprocessing, construction and improvement phases, and is able to find near-optimal solutions for real size instances very quickly. Initially, in the preprocessing part, reduction procedure of [19] is adapted in order to decrease the complexity of the problem. In the construction part of the heuristic, a promising initial solution is found by taking problem characteristics into consideration. Then, local search moves are adapted in a tabu search framework in order to improve the initial solution.

4.1 Preprocessing: Reduction Procedure

A reduction procedure is proposed in order to narrow down the solution space and to decrease the complexity of the problem. This is an adaptation of the preprocessing method presented for the GVRP in the study of [19] to our problem structure. In the procedure for each cluster dominated nodes are found and removed from the solution space. Proof of reduction procedure is adapted from [19].

Node i is *dominated if*
(1) $\exists$ a node $j \in \mathcal{C}_{\alpha(i)}, j \neq i$ such that $t_{pi}+t_{iq} \geq t_{pj}+t_{jq}$ for any $p, q \in \mathcal{V}/\mathcal{C}_{\alpha(i)}, \alpha(p) \neq \alpha(q)$ and,
(2) $\exists$ a node $j \in \mathcal{C}_{\alpha(i)}, j \neq i$ such that $t_{0i} \leq t_{j0}$ and,
(3) $\exists$ a node $j \in \mathcal{C}_{\alpha(i)}, j \neq i$ such that $s_{jk} \leq s_{ik}$ for any $k \in \mathcal{K}$, where $s_{ik} = \beta n_{ik}$ is sampling (information collection) time of node $i \in \mathcal{N}_0$ for option $k \in \mathcal{K}$.

Proposition 1. *The optimal solution to a Post-Disaster Needs Assessment Routing Problem instance does not change if a dominated node is removed.*

Proof. Let $i \in \mathcal{V}/0$ be a dominated node. If node i is not visited in the optimal solution, then removing node i from the instance obviously does not change the value of the optimal solution. Assume now that node i is visited in the optimal solution. If node i is visited by a route visiting exactly one site, then it is possible to exchange i with another node from $\mathcal{C}_{\alpha(i)}$ without worsening the objective function value. This follows from (2) in the definition of dominated nodes. If node i is visited on a longer route, then it is surrounded by nodes $p \in \mathcal{V}$ and $q \in \mathcal{V}$ where $\alpha(p) \neq \alpha(q)$ and either $p \neq 0$ or $q \neq 0$. Then, p and q satisfy the requirement of condition (1) in the definition. If the dominated node i is visited in the optimal solution with precision option k, where s_{ik} is the sampling time requirement for option k, condition (3) holds and it is possible to exchange i with another node from $\mathcal{C}_{\alpha(i)}$ without worsening the objective function value. $\square$

4.2 Construction of an Initial Solution

In the first part of MSTS, our aim is to find a promising initial feasible solution before starting systematic local search moves. The notation used in the construction algorithm is provided in Table 4.1 and the pseudocode is provided in Algorithm 1.

In the first step of the algorithm, we adapt the Cheapest Insertion Heuristic (CIH), which is a basic route construction method. Our motivation for using the CIH arises from the selective nature of the PDNARP. Starting from the most attractive nodes and leaving the further away (or less attractive) ones to the end, we aim to arrive at a promising initial feasible solution in the classical CIH methodology. However, in this approach, in order to lead the direction of the route to the most promising nodes in terms of both route duration and sampling time requirement, we assign a desirability score to each node.

In the PDNARP, besides routing considerations, node selection also depends on the sampling time requirements of nodes (i.e., population sizes of the nodes affects sampling times, and consequently node selection). For this reason, a promising node may not necessarily be the one with smaller travel times on the edges incident to it, but rather the one whose sampling and travel times is smaller. To take this into account, we assign a desirability score (denoted by $Score_1^i$) which quantifies the attractiveness of a node as a starting point. In order to select most promising node, we define the *impact region* for each node as a node with its two closest nodes, which together represent the potential neighbors of the corresponding node in the route (i.e., previous and following nodes in the route). Impact region of a node is demonstrated in Figure 4.1, where the impact region of node 30 consists of nodes 1 and 10.

The desirability score includes travel times between the node and its closest nodes, sampling times of its closest nodes, and the travel time from it to the depot. Reciprocal of this total time is proposed as $Score_1^i$, where the node with highest score is the most promising one:

$$Score_1^i = \frac{1}{t_{i0} + t_{ij} + t_{il} + s_{ik} + s_{jk} + s_{lk}} \tag{4.1}$$

where node $j \notin N$ and $l \notin N$ are the two closest nodes to node $i \in N$ in terms of travel time.

This score is used for selecting the first node in CIH. In the classical CIH procedure, after the initial node selection, the closest nodes (in terms of travel time) are added to the sequence one by one. This procedure continues until the terminating condition is satisfied.

However, we modify this sequence, since it may be desirable to select the node set which may increase the precision level within smaller sampling time. In other words, nodes with small population size may be more attractive and we may direct our route not only to the closest nodes in terms of travel time, but also to the nodes with smaller populations (which may be farther away), to decrease time spent in selected node. For this purpose, Step 2 of the algorithm involves a mathematical model for node selection.

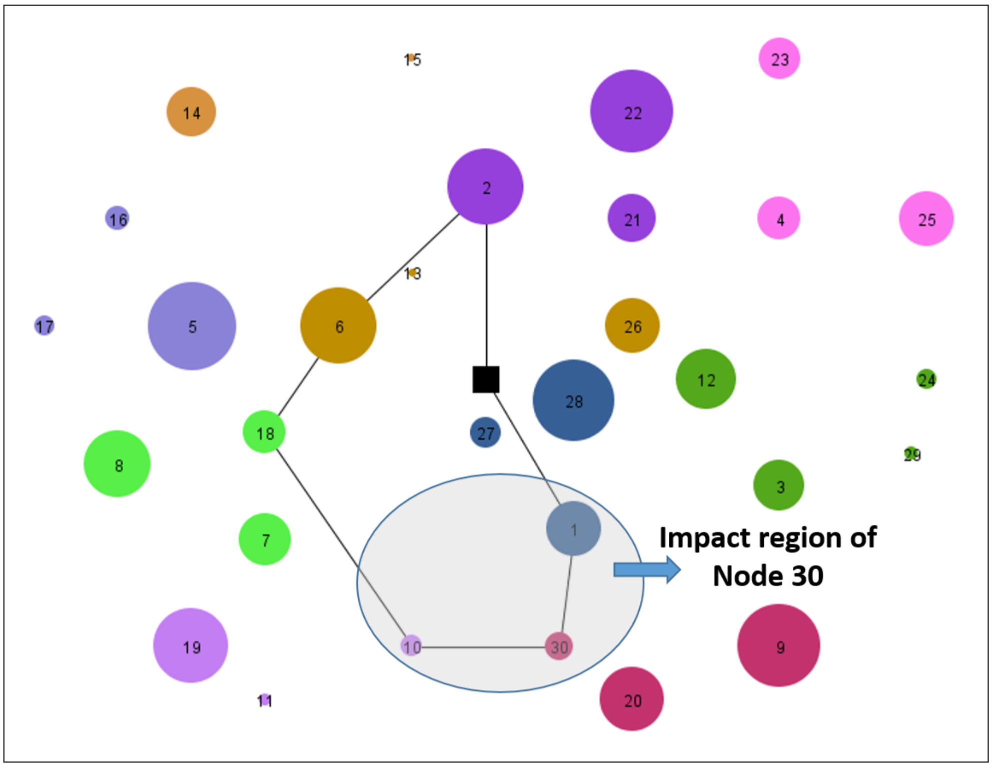

Figure 4.1: A feasible tour on a PDNARP instance with 30 nodes and 10 clusters, where the size of a node is directly proportional to its population

Table 4.1: Terminology of Iterative Construction Heuristic

Term	**Definition**
$\mathcal{S}(\mathcal{N})$	Set of all nodes
$\alpha(i)$	Set of all nodes that belong to cluster of node i
$\mathcal{S}(\mathcal{CIH})^{current}$	Current set of nodes selected by CIH
$\mathcal{S}(\mathcal{CIH})^{last}$	Last set of nodes selected by CIH
$\mathcal{S}(\mathcal{NS})$	Set of nodes selected by Node Selection Model
$\mathcal{F}$	Set of fixed nodes
$\mathcal{S}(\mathcal{A})$	Set of candidate nodes to be selected nodes in clusters other then those of $\mathcal{S}(\mathcal{CIH})^{current}$
$Score_1^i$	Desirability score for node $i \in \mathcal{N}$
$Score_2^i$	Total distance of node i to nodes in $\mathcal{F}$
t_{route}	Travel time of the current sequence
t_{total}	Travel and sampling (worst option case) times of the current sequence
$temp_{total}$	Temporary travel and sampling (worst option case) times of the current sequence
$dist(i, j)$	Distance between node i and node j
$samp(i)$	Sampling time of node i
$fixingDist$	Allowable distance between nodes to fix corresponding node
$termination$	Terminating condition
$same$	Counts unchanged solutions for termination
$tmax$	Maximum allowable time for the completion of the assessment
i'	predecessor of node i in current route
i''	successor of node i in current route
j'	predecessor of node j in current route
j''	successor of node j in current route

Algorithm 1 Iterative Construction Heuristic

Initialization:
$\mathcal{S}(\mathcal{CIH})^{last} = \emptyset$, $\mathcal{S}(\mathcal{CIH})^{current} = \emptyset$, $\mathcal{S}(\mathcal{NS}) = \emptyset$, $\mathcal{F} = \emptyset$
$\mathcal{S}(\mathcal{A}) = \mathcal{S}(\mathcal{N})$
t_{route}= 0, t_{total}= 0, $temp_{total}$= 0
$same$= 0
Step 1: Cheapest Insertion
if $\mathcal{F} = \emptyset$ **then**
    Select the starting node $i \in \mathcal{S}(\mathcal{N})$ with highest $Score_1^i$
    $\mathcal{S}(\mathcal{CIH})^{current} \leftarrow \mathcal{S}(\mathcal{CIH})^{current} \cup \{i\}$
    $\mathcal{S}(\mathcal{A}) \leftarrow \mathcal{S}(\mathcal{A})/\alpha(i)$
    $t_{route} \leftarrow t_{route} + dist(i', i) + dist(i, i'') - dist(i', i'')$
    $t_{total} \leftarrow t_{total} + samp(i) + dist(i', i) + dist(i, i'') - dist(i', i'')$
else
    for $\forall j \in \mathcal{F}$ **do**
        Find node $j \in \mathcal{S}(\mathcal{A})$ with minimum increase in travel time to add to $\mathcal{S}(\mathcal{CIH})^{current}$
        $temp_{total} \leftarrow temp_{total} + samp(j) + dist(j', j) + dist(j, j'') - dist(j', j'')$
        if $temp_{total} \leq tmax$ **then**
            $\mathcal{S}(\mathcal{CIH})^{current} \leftarrow \mathcal{S}(\mathcal{CIH})^{current} \cup \{j\}$
            $\mathcal{S}(\mathcal{A}) \leftarrow \mathcal{S}(\mathcal{A})/\alpha(j)$
            $t_{route} \leftarrow t_{route} + dist(j', j) + dist(j, j'') - dist(j', j'')$
            $t_{total} \leftarrow temp_{total}$
        end if
    end for
end if
if $t_{total} \leq tmax$ **then**
    Find node $i \in \mathcal{S}(\mathcal{A})$ with minimum increase in travel time to add to $\mathcal{S}(\mathcal{CIH})^{current}$
    $temp_{total} \leftarrow temp_{total} + samp(j) + dist(j', j) + dist(j, j'') - dist(j', j'')$
    if $temp_{total} \leq tmax$ **then**
        $\mathcal{S}(\mathcal{CIH})^{current} \leftarrow \mathcal{S}(\mathcal{CIH})^{current} \cup$ {i}
        $\mathcal{S}(\mathcal{A}) \leftarrow \mathcal{S}(\mathcal{A})/\alpha(i)$
        $t_{route} \leftarrow t_{route} + dist(i', i) + dist(i, i'') - dist(i', i'')$
        $t_{total} \leftarrow temp_{total}$
    else
        Go to Step 2.
    end if
end if
if $\mathcal{S}(\mathcal{CIH})^{current} = \mathcal{S}(\mathcal{CIH})^{last}$ **then**
    $same \leftarrow same + 1$
end if
if $same \geq termination$ **then**
    break and terminate Algorithm 1.
else
    $\mathcal{S}(\mathcal{CIH})^{last} = \mathcal{S}(\mathcal{CIH})^{current}$
end if

Algorithm 1 Iterative Construction Heuristic (cont'd)

Step 2: Node Selection
Initialize $\mathcal{S}(\mathcal{NS}) = \emptyset$
Apply node selection IP model
Update $\mathcal{S}(\mathcal{NS})$
Step 3:Node Fixing Procedure
Compare the nodes $\in$ sets $\mathcal{S}(\mathcal{CIH})$ and $\mathcal{S}(\mathcal{NS})$
for $\forall i \in \mathcal{S}(\mathcal{NS})$ **do**
  for $\forall j \in \mathcal{S}(\mathcal{CIH})$ **do**
    if $dist(i,j) \leq fixingDist$ **and** j has min $Score_2^j$ **then**
      Update $\mathcal{F} = \mathcal{F} \cup \{i\}$
      Go to Step 1.
    end if
  end for
end for
Go to Step 1.

The node selection model has the objective of maximizing the total precision level where the number of nodes is restricted with the number of the nodes found with the CIH heuristic in Step 1. The mathematical model of this selection problem is provided below. For the explanation of the previously defined variables and parameters, see Section 3.4.4.

Additional Parameters

L: # of nodes on the route found with the CIH heuristic in Step 1.

t: Travel time of the route found with CIH in Step 1.

Decision variables

g_{ik} = 1, if option $k \in \mathcal{K}$ is selected for node $i \in \mathcal{N}_0$; 0, otherwise

MIP Formulation

$$Minimize \quad \sum_{i \in \mathcal{N}_0} \sum_{k \in \mathcal{K}} g_{ik} h_{ik} \tag{4.2}$$

s.t.

$$\sum_{k \in \mathcal{K}} g_{ik} \leq 1 \qquad \forall i \in \mathcal{N}_0 \tag{4.3}$$

$$\sum_{j \in \mathcal{C}_{\alpha(i)}} \sum_{k \in \mathcal{K}} g_{jk} \leq 1 \qquad \forall i \in \mathcal{N}_0, \mathcal{C}_{\alpha(i)} \in \mathcal{C} \tag{4.4}$$

$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{N}_0} g_{ik} \leq L \tag{4.5}$$

$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{N}_0} \beta n_{ik} g_{ik} \leq (\mathcal{T}_{max} - t) \tag{4.6}$$

$$g_{ik} \in \{0,1\} \qquad \forall i \in \mathcal{N}_0, k \in \mathcal{K} \tag{4.7}$$

Objective function of this node selection model (4.2) minimizes the total half width of the confidence interval on the prevalence of the selected nodes. Selecting only one node per cluster is imposed by (4.3). Selecting only one option per node is imposed by (4.4). Upper limit on number of nodes to select is provided by (4.5). Time restriction is given in constraint (4.6) and (4.7) are the sign constraints.

In Step 3 of MSTS, the nodes $i \in \mathcal{N}$ found in Steps 1 and 2 are candidate for fixing if they satisfy at least one of the following conditions:

- (1) If there exists a node $i \in N$ which is common in solutions of Step 1 and Step 2.
- (2) If there exists a node $j \in N$ from Step 1 and node $i \in N$ from Step 2 whose travel time (t_{ij}) is smaller than a pre-determined threshold (represented as $fixingDist$)

After finding the candidate nodes for fixing, node $i \in F$ with minimum $Score_2^i$ is fixed. This score provides selection of the node which is closer to all nodes in F in terms of travel time. $Score_2^i$ is given by:

$$Score_2^i = \sum_{j\in\mathcal{F}, j\neq i} t_{ij} \tag{4.8}$$

Algorithm 1 is an iterative process where in each iteration one node is fixed and the procedure continues with Step 1. The construction phase terminates when the solution of the CIH heuristic does not change in a pre-specified number of iterations (represented as $termination$).

4.3 Improvement by Tabu Search

In the improvement step of developed method, basic local search moves are combined using a tabu search framework. Keeping in mind that the primary objective is accu-

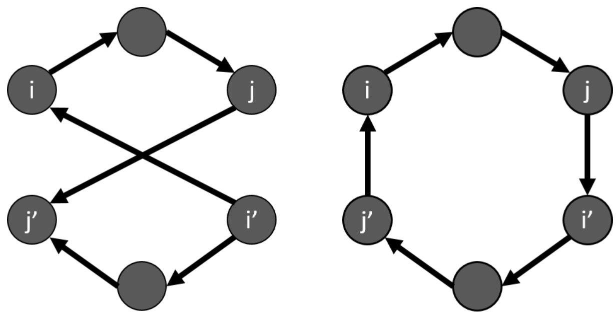

Figure 4.2: Change in route after *2-opt* move

racy within acceptable precision levels for each selected site, finding a shorter route is the primary consideration in this step, since a shorter route may allow visiting more sites, thereby increasing accuracy. In other words, in the improvement part, the aim is to find the best tour with the maximum number of sites selected. Both intensification and diversification strategies are applied for this purpose.

4.3.1 Intensification Strategies

For the intensification strategy, the purpose is to obtain the shortest route with the maximum possible number of selected nodes. In order to exploit the solution space in depth for shortening the route duration, we apply a number of simple TSP/VRP local search moves in the tabu search framework. *2-opt*, *swap*, $Replace_{1-1}$ and $Replace_{1-2}$ are the moves included in this algorithm.

The *2-opt* move basically reorders the current route by making a crossover. We apply a complete *2-opt* local search where all possible feasible combinations are attempted. A *2-opt* move is illustrated in Figure 4.2 (see Algorithm 3 for the pseudocode).

The *swap* move checks all possible exchanges among each cluster pairs separately. In this move, position of the exchanged node is not changed. According to the locations and population sizes of sites in a cluster, all possible exchanges are examined. *Swap* move is illustrated in Figure 4.3 (see Algorithm 4 for the pseudocode).

$Replace_{1-1}$ is a move represents the exchange of one node with another. All selected nodes are candidates to remove and all possible candidates among unselected nodes are considered for insertion to any position in the route. A $Replace_{1-1}$ move is illus-

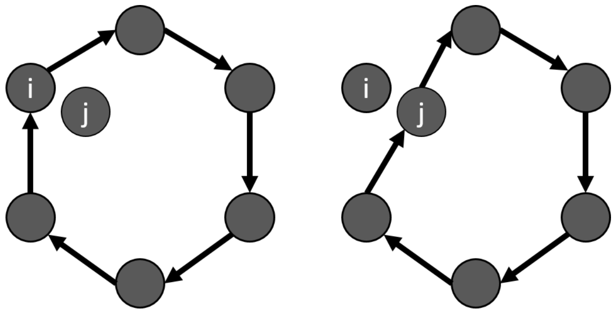

Figure 4.3: Change in route after *swap* move

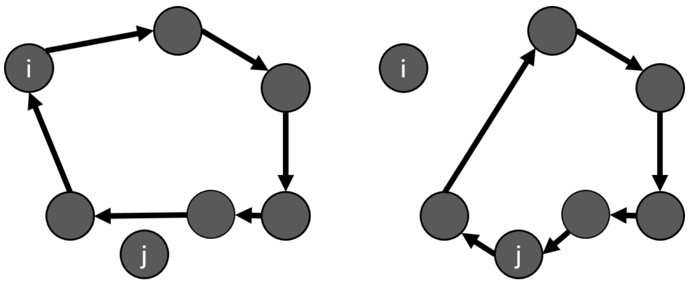

Figure 4.4: Change in route after $Replace_{1-1}$ move

trated in Figure 4.4 (see Algorithm 5 for the pseudocode)

$Replace_{1-2}$ is a move that represents the exchange of one node in the route with two nodes not included in it. This move is added in order to increase the number of nodes selected. This move examines all possible exchanges where a selected node is removed from the route and two feasible unselected nodes are inserted to any positions of the route at the same time. A $Replace_{1-2}$ move is illustrated in Figure 4.5 (see Algorithm 6 for the pseudocode)

The first two moves (*2-opt* and *swap*) are used to shorten the route with the current node set. $Replace_{1-1}$ move tries to replace one of the current node with a new node

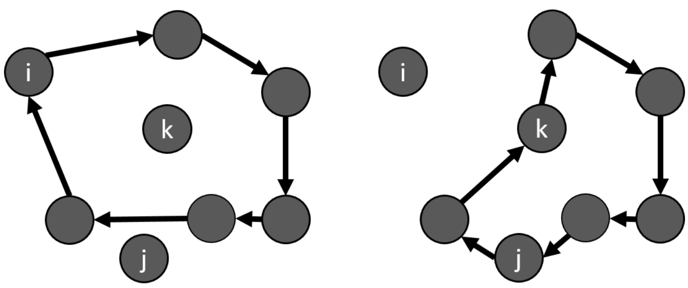

Figure 4.5: Change in route after $Replace_{1-2}$ move

from out of the current set. Finally, $Replace_{1-2}$ is used to increase number of selected nodes thanks to the shortened path after the implication of *2-opt*, *swap* and $Replace_{1-1}$ moves.

4.3.2 Diversification Strategies

In addition to the application of intensification strategies, we use two different diversification strategies, Div_1 and Div_2, in order to explore the solution space better and to avoid being stuck into local optima. For this end, the tabu search method involves multiple starts. Each restart point is obtained by the implementation of one of these two diversification strategies.

Div_1 is a restarting procedure where a predetermined proportion of selected nodes are removed from the current solution randomly. This random start procedure provides moving to a different portion of the solution space. For the implementation of Div_1, it is enforced to start from a different solution at each restart.

Div_2 is a restarting procedure where the nodes which stay longer in the current solution are removed. In this strategy we benefit from the *frequency based (long term) memory* of the tabu search metaheuristic. By counting the number of iterations that a node stays in the current solution, we determine the set of nodes which can be removed from the current set in order to move away from the current solution

4.4 Selection of Options in Visited Nodes

In order to allocate the remaining time from travel between nodes to selected nodes for sampling, *a 0-1 knapsack model* is used in the algorithm. Since the number of nodes and options are not too large even in real-sized problems, implementation of this model does not increase the solution time significantly. This model is applied in the final step of each restart.
For the explanation of the previously defined variables and parameters in this model, please see Section 3.4.4.

Additional Parameters

R: Number of nodes on final route found in the improvement part

t: Travel time found with the CIH heuristic in Step 1.

Decision variables

r_{ik} = 1, if option $k \in \mathcal{K}$ is selected for node $i \in \mathcal{R}$; 0, otherwise

0-1 Knapsack Model

$$Maximize \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{N}} r_{ik}(0.5 - h_{ik}) \tag{4.9}$$

s.t.

$$\sum_{k \in \mathcal{K}} r_{ik} \leq 1 \qquad \forall i \in \mathcal{R} \tag{4.10}$$

$$\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{R}} r_{ik} s_{ik} \leq (\mathcal{T}_{max} - t) \tag{4.11}$$

$$r_{ik} \in \{0, 1\} \qquad \forall i \in \mathcal{N}_0, k \in \mathcal{K} \tag{4.12}$$

Objective function of this 0-1 knapsack model (4.9) maximizes the total precision level (by minimizing total half width of the confidence interval for prevalence of the nodes in the route). Selecting only one node per cluster is imposed by (4.10). Time restriction is given in constraint (4.11). (4.12) are the sign constraints.

4.5 Overview of the Multi-Start Tabu Search Algorithm

The multi-start tabu search algorithm starts with the initialization of parameters in Step 0. At first, *2-opt* and *swap* moves are applied in Step 1.1, then $Replace_{1-1}$ and $Replace_{1-2}$ moves are applied in Step 1.2. This procedure is performed until reaching a predetermined number of iterations (represented by *maxMainIter*). At the end of the local search moves, in Step 2, selection of options is determined using the 0-1 knapsack model. This procedure is applied for a predetermined number of times (represented by *maxStart*) in order to find the set of final solutions. At each start, one of the diversification strategies (Div_1 or Div_2) is applied in order to create a new initial solution. Then, the improvement moves are repeated. Finally, best solution among the set of final solutions is selected.

In MSTS algorithm, solution attributes are *2-opt*, *swap*, $Replace_{1-1}$ and $Replace_{1-2}$ local search moves of all nodes in $\mathcal{N}$. When one of these solution attributes occurs in moving from current solution to another solution, its tabu status turns into tabu-active. A tabu-active solution attribute is kept in tabu list during predetermined number of iterations, called as tabu tenure. If the solution attribute is tabu-active, this move cannot be applied in the following iteration. For example, if in the current iteration a swap move is applied where node i is removed from and node j is inserted to the route, the same swap move (removing node i and inserting node j) cannot be applied during tabu tenure (denoted by $tabuTenure_{swap}$). Number of iterations that a solution attribute is tabu-active is counted in order to change tabu status of that solution.

Table 4.2: Terminology of Iterative Construction Heuristic

Term	**Definition**
$tabuTenure_{2opt}$	Predetermined tabu tenure for *2-opt* move
$tabuTenure_{swap}$	Predetermined tabu tenure for *swap* move
$tabuTenure_{remove}$	Predetermined tabu tenure for *remove* move
$tabuTenure_{insert}$	Predetermined tabu tenure for *insert* move
$tabuIter_{2opt}(i)$	Counts number of iterations that solution attribute i is tabu-active during *2-opt* moves
$tabuIter_{swap}(i)$	Counts number of iterations that solution attribute i is tabu-active during *swap* moves
$tabuIter_{remove}(i)$	Counts number of iterations that solution attribute i is tabu-active during $Replace_{1-1}$ and$Replace_{1-2}$ moves
$tabuIter_{insert}(i)$	Counts number of iterations that solution attribute i is tabu-active during $Replace_{1-1}$ and $Replace_{1-2}$ moves
$maxStart$	Number of starts to improvement algorithm
$maxMainIter$	Number of iterations where move set is repeated
$maxIter_1$	Number of iterations where *2-opt* and *swap* moves are repeated in each main iteration
$maxIter_2$	Number of iterations where $Replace_{1-1}$ and$Replace_{1-2}$ moves are repeated in each main iteration
$start$	Counts number of starts
$mainIter$	Counts number of main iterations in each start
$iter_1$	Counts number of iterations for *2-opt* and *swap* moves in each main iteration
$iter_2$	Counts number of iterations for $Replace_{1-1}$ and $Replace_{1-2}$ moves in each main iteration
$route_{const}$	Route constructed by ICH
$route_{current}$	Current route
$route_{next}$	Next (candidate) route
$route_{best}$	Best route found
rd_{const}	Route duration of route constructed by ICH
$rd_{current}$	Current route duration
rd_{next}	Next route duration
rd_{best}	Best route duration
$total_{const}$	Total time of route constructed by ICH (route duration and sampling time for worst precision option for all selected nodes)
$total_{current}$	Current total time
$total_{next}$	Next total time
$total_{best}$	Best total time
$length_{const}$	Length of route constructed by ICH
$lenght_{current}$	Length of current route
$lenght_{next}$	Length of next route
$lenght_{best}$	Length of best route
$z_{current}$	Objective value of current solution
z_{best}	Objective value of best solution
$countInRoute_i$	Counts number of iterations that node $i \in \mathcal{N}_0$ stays in current solution
F_{iter}	Set of iterations where $Diversification_1$ should be applied

Algorithm 2 Improvement- Tabu Search

Step 0:Initialization
Set $tabuTenure_{2opt}$, $tabuTenure_{swap}$, $tabuTenure_{remove}$, $tabuTenure_{insert}$
Set $maxStart$, $maxMainIter$, $maxIter_1$, $maxIter_2$
$tabuIter_{2opt} = 0$, $tabuIter_{swap} = 0$, $tabuIter_{remove} = 0$, $tabuIter_{insert} = 0$
$start = 0$, $mainIter = 0$, $iter_1 = 0$, $iter_2 = 0$
$route_{best} = 0$, $rd_{best} = 0$, $total_{best} = 0$, $length_{best} = 0$
$route_{current} = route_{const}$, $rd_{current} = rd_{const}$, $total_{current} = total_{const}$, $length_{current} = length_{const}$
$z_{best} = 0$, $z_{current} = 0$
$countInRoute_i = 0 \quad \forall i$
Step 1
while $start \leq maxStart$ **do**
Step 1.1
  while $mainIter \leq maxMainIter$ **do**
    $iter_1 = 0$, $iter_2 = 0$
    while $iter_1 \leq maxIter_1$ **do**
      Apply $2 - opt$ local search procedure
      Apply $swap$ local search procedure
    end while
Step 1.2
    while $iter_2 \leq maxIter_2$ **do**
      Apply $Replace_{1-1}$ local search procedure
      Apply $Replace_{1-2}$ local search procedure
    end while
Step 2
    Apply *0-1 knapsack model* to assign options to selected nodes
    Update z_{best}
    if $iter \geq 1$ **then**
      Update $route_{best} = route_{current}$, $rd_{best} = rd_{current}$, $total_{best} = total_{current}$, $length_{best} = length_{current}$
    end if
  end while
Step 3
  if $start \in F_{iter}$ **then**
    Apply Div_1
  else
    Apply Div_2
  end if
end while

Algorithm 3 Local Search - *2opt*

i' is the successor node of i in $route_{current}$
j' is the successor node of j in $route_{current}$
for $\forall\ i$ in $route_{current}$ **do**
  for $\forall\ j$ in $route_{current}$ **do**
    if $(tabuIter_{2opt}(i) \leq tabuTenure_{2opt})$ **and** $(tabuIter_{2opt}(j) \leq tabuTenure_{2opt})$ **then**
      Exchange arcs (i, i') and (j, j')
      Update $rd_{next} \leftarrow rd_{current} + dist(i, j') + dist(j, i') - dist(i, i') - dist(j, j')$
      Update $total_{next} \leftarrow total_{current} + dist(i, j') + dist(j, i') - dist(i, i') - dist(j, j')$
      if $total_{next} \leq Tmax$ **then**
        Update $total_{current} = total_{next}$, $rd_{current} = rd_{next}$, $route_{current} = route_{next}$, $length_{current} = length_{next}$
      end if
      if $rd_{current} \leq rd_{best}$ **then**
        Update $route_{best} = route_{current}$, $rd_{best} = rd_{current}$, $total_{best} = total_{current}$, $lenght_{best} = length_{current}$
        for $\forall k \in route_{current}$ **do**
          Update $countInRoute_k$=$countInRoute_k$+1
        end for
      end if
    end if
  end for
end for

Algorithm 4 Local Search - *Swap*

```
i' is the predecessor and i'' is the successor nodes of i in route_current
C_{s_i} is set of nodes in cluster of node i
for ∀ i in route_current do
    for ∀ j in N do
        if j ∈ Cs_i then
            if (tabuIter_swap(i) ≤ tabuTenure_swap) and (tabuIter_swap(j) ≤ tabuTenure_swap) then
                Remove node i from route_next
                Insert node j to same location of route_next
                Update rd_next ← rd_current + dist(i', j) + dist(j, i'') − dist(i', i) − dist(i, i'')
                Update total_next ← total_current + samp(j) − samp(i) + dist(i', j) + dist(j, i'') − dist(i', i) − dist(i, i'')
                if total_next ≤ Tmax then
                    Update total_current = total_next, rd_current = rd_next, route_current = route_next, length_current = length_next
                end if
                if rd_current ≤ rd_best then
                    Update route_best = route_current, rd_best = rd_current, total_best = total_current, lenght_best = length_current
                    for ∀k ∈ route_current do
                        Update countInRoute_k=countInRoute_k+1
                    end for
                end if
            end if
        end if
    end for
end for
```

Algorithm 5 Local Search - $Replace_{1-1}$

i' is the predecessor and i'' is the successor nodes of i in $route_{current}$
j' is the predecessor and j'' is the successor nodes of j in $route_{next}$
for $\forall\ i$ in $route_{current}$ **do**
  for $\forall\ j$ in $\mathcal{N}$ **do**
    if $(tabuIter_{remove}(i) \leq tabuTenure_{remove})$ **and** $(tabuIter_{insert}(j) \leq tabuTenure_{insert})$ **then**
      Remove node i from $route_{next}$
      Insert node j to best location of $route_{next}$
      Update $rd_{next} \leftarrow rd_{current} + dist(j', j) + dist(j, j'') - dist(i', i) - dist(i, i'')$
      Update $total_{next} \leftarrow total_{current} + samp(j) - samp(i) + dist(j', j) + dist(j, j'') - dist(i', i) - dist(i, i'')$
      if $total_{next} \leq Tmax$ **then**
        Update $total_{current} = total_{next}$, $rd_{current} = rd_{next}$, $route_{current} = route_{next}$, $length_{current} = length_{next}$
      end if
      if $rd_{current} \leq rd_{best}$ **then**
        Update $route_{best} = route_{current}$, $rd_{best} = rd_{current}$, $total_{best} = total_{current}$, $lenght_{best} = length_{current}$
        for $\forall k \in route_{current}$ **do**
          Update $countInRoute_k$=$countInRoute_k$+1
        end for
      end if
    end if
  end for
end for

Algorithm 6 Local Search - $Replace_{1-2}$

i' is the predecessor and i'' is the successor nodes of i in $route_{current}$
j' is the predecessor and j'' is the successor nodes of j in $route_{next}$
k' is the predecessor and k'' is the successor nodes of k in $route_{next}$
for $\forall$ i in $route_{current}$ **do**
  for $\forall$ j in $\mathcal{N}$ **do**
    for $\forall$ k in $\mathcal{N}$ **do**
      if $(tabuIter_{remove}(i) \leq tabuTenure_{remove})$ **and** $(tabuIter_{insert}(j) \leq tabuTenure_{insert})$ **and** $(tabuIter_{insert}(k) \leq tabuTenure_{insert})$ **then**
        Remove node i from $route_{next}$
        Insert node j to best location of $route_{next}$
        Insert node k to best location of $route_{next}$
        Update $rd_{next} \leftarrow rd_{current} + dist(j', j) + dist(j, j'') + dist(k', k) + dist(k, k'') - dist(i', i) - dist(i, i'')$
        Update $total_{next} \leftarrow total_{current} + samp(j) + samp(k) - samp(i) + dist(j', j) + dist(j, j'') + dist(k', k) + dist(k, k'') - dist(i', i) - dist(i, i'')$
        if $total_{next} \leq Tmax$ **then**
          Update $total_{current} = total_{next}$, $rd_{current} = rd_{next}$, $route_{current} = route_{next}$, $length_{current} = length_{next}$
        end if
        if $rd_{current} \leq rd_{best}$ **then**
          Update $route_{best} = route_{current}$, $rd_{best} = rd_{current}$, $total_{best} = total_{current}$, $lenght_{best} = length_{current}$
          for $\forall m \in route_{current}$ **do**
            Update $countInRoute_m$=$countInRoute_m$+1
          end for
        end if
      end if
    end for
  end for
end for

CHAPTER 5

COMPUTATIONAL STUDY

In this chapter, we conduct computational studies on (i) randomly generated instances and (ii) a case study based on a historical earthquake scenario to evaluate the performance of the proposed solution approach. Generation of test instances are described in detail in §5.1, whereas algorithm settings are given in §5.2. In §5.3, we present the results of the computational study.

5.1 Test Instances

The multi-start tabu search approach for the PDNARP is tested on two sets of instances. The first one consists of a randomly generated instance set where the coordinates of different sized networks are modified from the well-known *Solomon instances* [76]. The second instance set is based on a case study on the 2011 Van Earthquake for the analysis of the performance on a real-sized problem.

For both instance sets, precision level options are determined according to preferences of the assessor agencies. Then, in order to quantify precision level options, sample size requirements are calculated according to decided parameters.

Initial estimate on prevalence for each node ($\hat{p}$) is taken as 0.5, which is the most risk averse value in case of no initial information. Confidence level $(1 - \alpha)$ is selected as 95%, which can be increased or decreased according to preferences of the agencies or availability of resources. Corresponding *z score* ($z_{\frac{\alpha}{2}}$) under the two tailed normal distribution assumption is equal to 1.96, where α is 0.05 for 95% CI. Precision levels are discretized starting from 0.05, with increments of 0.025 up to 0.20. Based on these parameters and the population of the node, sample size requirement of each precision option is calculated. Then, multiplying the sample size with a constant unit survey time, total sampling times are obtained for precision level options of each node. At the end, sampling times of options and corresponding precision levels are used as parameters in the solution methods. Calculated sample size and sampling time values are provided in Tables A.1, A.2 and A.3 in Appendix A for the test instances used.

5.1.1 Random Instance Set Based on Solomon Instances

Random instances are generated by modifying Solomon's 100-node random instances [76]. Travel times between node pairs are calculated using the coordinates taken from these instances. Euclidean distance metric is used for the travel time calculations. Population size of each node is randomly generated between 100 and 20,000.

Since computational requirements for solution increase number of clusters, we use different size of clusters for each instance set. Moreover, three different clustering strategies are used. In *geographical* clustering, nodes are partitioned according to their geographical closeness. In *hierarchically* clustered instances, clusters are determined according to their distances to the epicenter of the disaster. In our setting, node 0 represents the epicenter of the disaster and assessment starts from that point. The nodes whose distances to the epicenter are similar belong to same cluster. In this setting, it is assumed that impact of the disaster is similar in nodes with similar distances to the epicenter. Finally, we generate a *random* clustered data set for each node and cluster size to represent a network structure where multiple attributes may lead to clustering nodes in various parts of the network in the same cluster.

A set of 78 instances with different sizes of nodes (30, 50, 75), clusters (10, 15, 20, 25), cluster types (geographic, hierarchical and random) and time limits (3, 5, 7, and 10 hours) are generated. Information about node size, cluster size, cluster type and time limit is involved in the names of the random test instances. For example, $30N\_10C\_tmax5\_geo$ represents a geographically clustered network with 30 node and 10 clusters and a time limitation of 5 hours. Representative networks for each type of clustering mechanism are provided for one of the random test instance (with 75 nodes and 15 clusters) in Figures 5.1, 5.2 and 5.3. In these figures, node size (radius of the circle) represents the population size of the corresponding node. The nodes with same color belong to the same cluster, and the depot (epicenter of the disaster) is represented by a black square in the middle of the network.

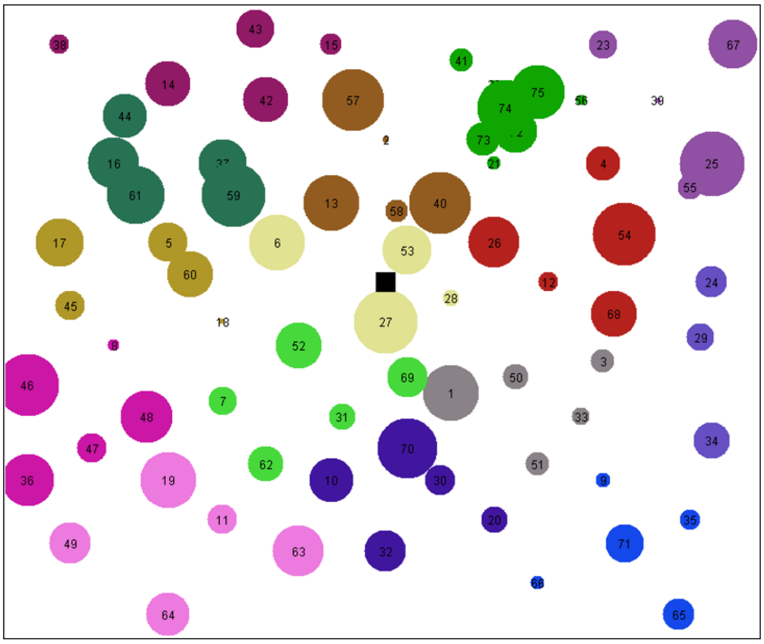

Figure 5.1: Network representation of geographic cluster on the instance with 75 nodes and 15 clusters

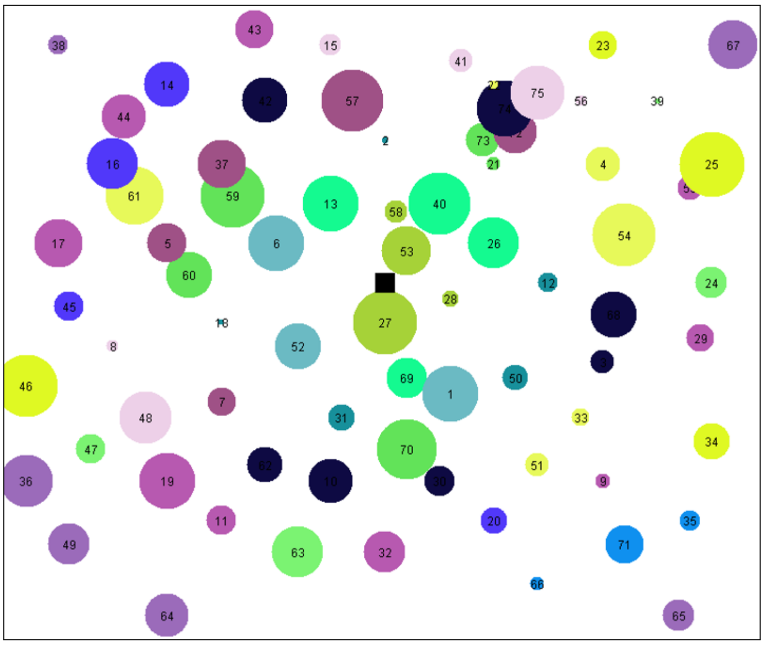

Figure 5.2: Network representation of hierarchic cluster on the instance with 75 nodes and 15 clusters

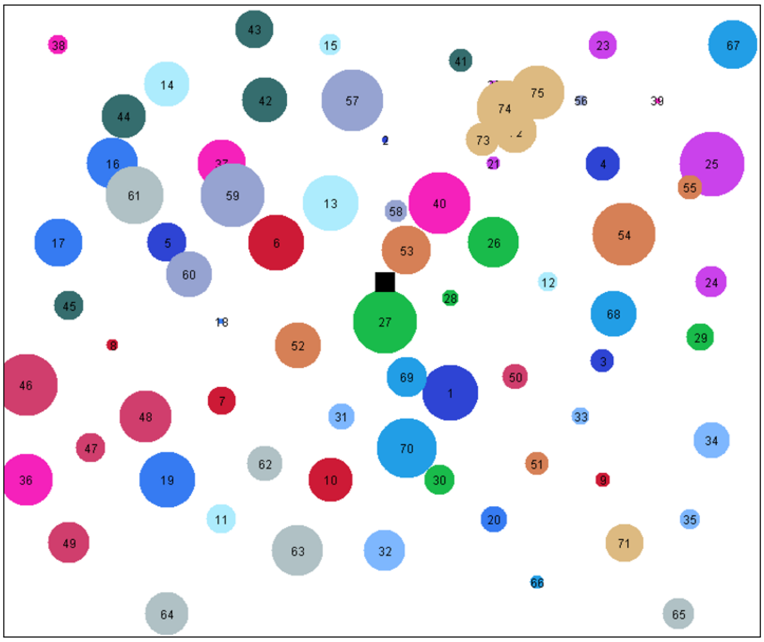

Figure 5.3: Network representation of random cluster on the instance with 75 nodes and 15 clusters

5.1.2 A Case Study Based on the 2011 Van Earthquake

In this study, our aim is to present a quick and efficient solution method that can be used in practice. In order to evaluate the performance of our algorithm in a real life disaster, the proposed solution method is applied to one of the most catastrophic disasters in Turkey, the Van Earthquake of 2011. In this disaster, more than 600 people died, 4,000 people injured and thousands of structures were destroyed. Due to damage in buildings around 60,000 beneficiaries were left homeless. Rescue and relief efforts are directed by Turkey's Ministry of Health and Turkish Red Crescent with the help of many other national and international humanitarian organizations. In the immediate aftermath, while rescue operations were performed, relief distribution operations also started. Some of the values on needs supplied within first week of the response phase are recorded as the following: 4,440 search and rescue personnel, 1,710 medical personnel, 146 ambulances, 42,711 tents, 54 collective shelter tent, 69 general purpose tents, 65 prefabricated houses, 160360 blankets, 37 mobile kitchens, 3,051 kitchen sets, 6,899 catalytic stoves, 5,792 sleeping bags and so on [80]. In such a crisis, efficient use of resources will provide true matches of beneficiaries with needs. This can only be possible with an efficient and effective assessment.

The real-world data from the 2011 Van Earthquake is used in the case study data, which was obtained from the study of [14]. Coordinates of the affected areas (districts, villages etc.), elevation of these places, and population characteristics (number of people younger than 14, older than 65, number of people disabled and females with children) are the available information before the assessment process (called as secondary information). Raw data used in the case study is available in Table A.4 in Appendix A

Secondary information is also used for the clustering of the sites affected in the Van Earthquake. In order to form clusters there are many different ways in the literature, including *k*-means clustering, density-based spatial clustering, mean shift clustering and agglomerative hierarchical clustering are some of these techniques. According to the characteristics of the secondary information, such as number of attributes and types of the data (continuous, binary, ordinary, etc.) any of these clustering methods can be applied.

In this case study, *k*-means clustering algorithm, which is a simple and well-known algorithm in the clustering literature, is implemented to our problem [47]. We use XLSTAT add-in of Excel software for the calculations of the *k*-means algorithm. In order to create an instance set, we use three different cluster sizes, determined according to the ratio of within variance to the between variance of clusters. In Table 5.1, within and between variance values are provided for the increasing number of clusters. Variance change is also presented in Figure 5.4, which also shows the breakpoints we decide to select as cluster sizes. We select three of these alternative clusters among the ones with smaller within/between variance ratio. For the Van Earthquake data set of 93 nodes, networks with 19, 23 and 29 clusters are selected. Generated instance set is given in Table 5.2. Names of the instances include number of nodes, number of clusters and time limit information. Network representation on a map is provided in Figure 5.5, where each color and each number represents a cluster.

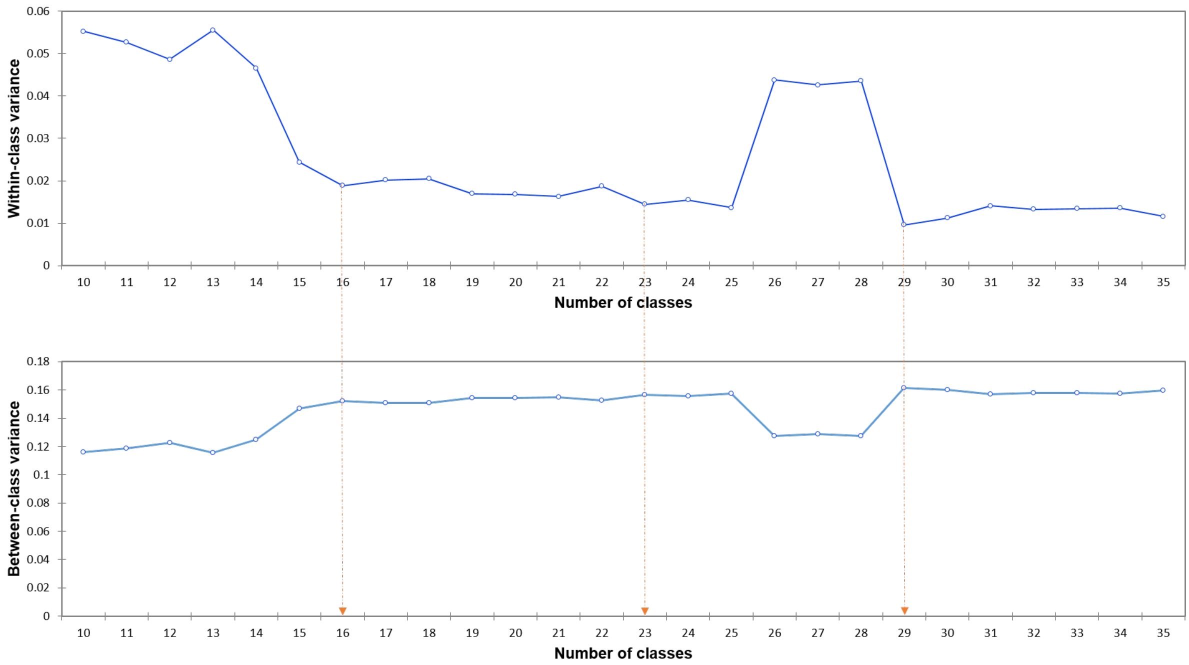

Figure 5.4: Within and between variance change according to increasing number of clusters

Table 5.1: Change in variance with increasing number of clusters

Variance\Classes	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Within-class	0.06	0.05	0.05	0.06	0.05	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.01	0.02	0.01	0.04	0.04	0.04	0.01	0.01
Between-classes	0.12	0.12	0.12	0.12	0.12	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.16	0.16	0.16	0.13	0.13	0.13	0.16	0.16
Within/Between	0.48	0.44	0.40	0.48	0.37	0.17	**0.12**	0.13	0.14	0.11	0.11	0.11	0.12	**0.09**	0.10	0.09	0.34	0.33	0.34	**0.06**	0.07

5.2 Algorithm Settings

In this section, analyses on the selection of heuristic parameters are summarized.

Threshold distance to fix nodes (fixingDist)

After applying Step 1 and Step 2 in construction part, Algorithm 1 decides on whether node should be fixed or not. If there is no common node selected in both steps of construction, node fixing is applied only if there exist two nodes (one node from CI and other node from Node Selection Model) which are close enough to each other. After preliminary trials on different alternatives, we fix this parameter as 0.20, since the minimum pairwise distance between nodes is 0.10 and maximum pairwise distance is

Table 5.2: Case study instances

#	Name	Nodes	Clusters	Tmax
1	93N_16C_tmax10	93	16	10
2	93N_16C_tmax15	93	16	15
3	93N_16C_tmax20	93	16	20
4	93N_23C_tmax10	93	23	10
5	93N_23C_tmax15	93	23	15
6	93N_23C_tmax20	93	23	20
7	93N_23C_tmax25	93	23	25
8	93N_29C_tmax10	93	29	10
9	93N_29C_tmax15	93	29	15
10	93N_29C_tmax20	93	29	20
11	93N_29C_tmax25	93	29	25

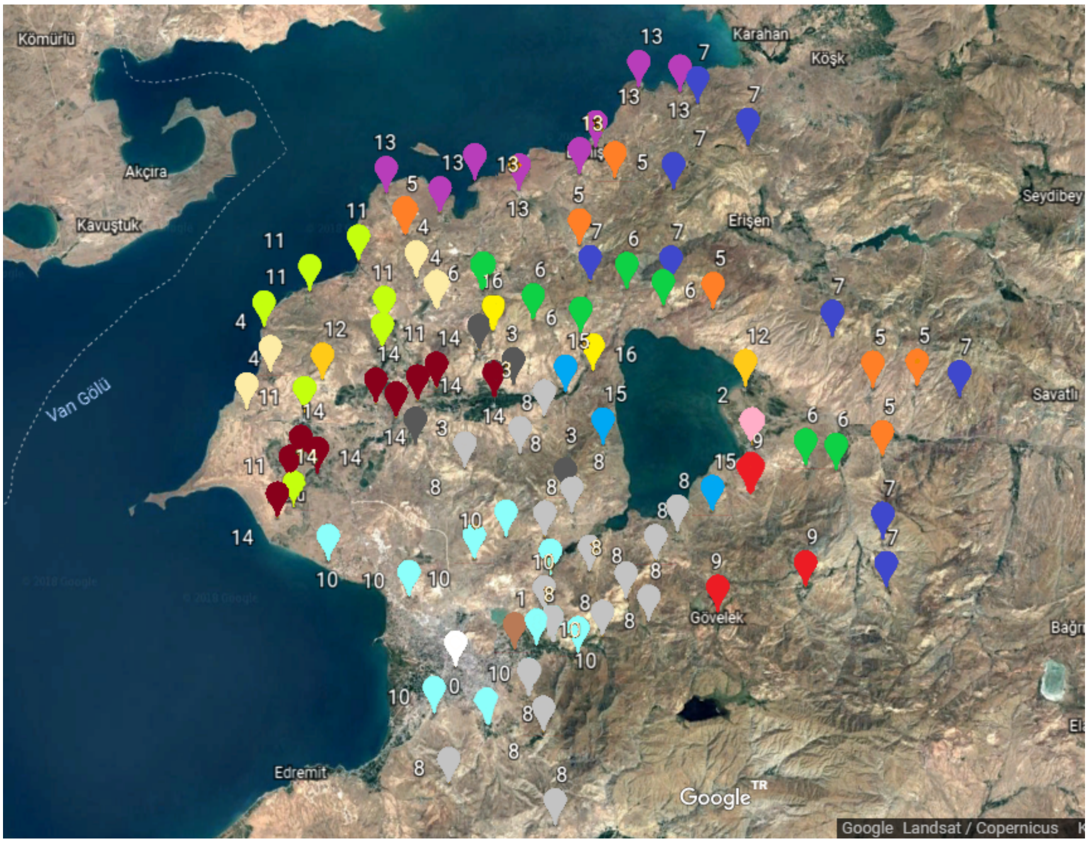

Figure 5.5: Representation of the Van Case Study network on map with 16 clusters (map was retrieved from Google Earth)

3.06 in the Solomon instances. If the distance between nodes is smaller than or equal to 0.20, we fix the node selected in the Node Selection Model (in order to change the solution).

Tabu tenures for local search moves

In the improvement part of the MSTS, we use *2-opt*, *swap*, $Replace_{1-1}$ and $Replace_{1-2}$ moves in order to shorten the route and increasing the number of nodes selected. During the algorithm iterations, solution attributes that belong to recently selected nodes are kept in the tabu list until a predetermined number of iterations. The main motivation on the implementation of tabu tenures is to prevent cycling by restricting the neighborhood of the solution.

$tabuTenure_{2opt}$, $tabuTenure_{swap}$, $tabuTenure_{remove}$ and $tabuTenure_{insert}$ are the implemented tabu tenure values for the local search moves. We analyze the effect of tabu tenure on solution by trying various alternatives (between 3 and 10) in our computational experiments.

Proportion of nodes to remove in diversification

Preliminary results of the heuristic approach show that there are many common nodes in the heuristic solution and the best solution found by CPLEX. At least half of the selected nodes are common in both solutions in general. To avoid being stuck at a local optimum, we remove a proportion of selected nodes to find new starting solution for the improvement part of the algorithm. In the computational analysis, we try alternative removal proportion values (varying in increments of 0.05 between 0.3 and 0.5)

Number of restarts

Since the computation time of each iteration is small (0.3 seconds on average), having a large number of restarts does not increase computational time of the MSTS heuristic excessively. Additionally, increasing the number of restarts increases the chance of finding the global optimum. In some of the instances where the optimality gap of heuristic results is high, the objective function value is able to improved only in the further (i.e., after the 100th restart) iterations. Thus, increasing the number of restarts improves the solution. In the computational study, number of restarts is selected as 200 where either div_1 or div_2 is applied as a restarting strategy.

Number of sub-iterations for local search moves

To decide on number of sub-iterations for local search moves, we observe the change in the objective function in one restart in the test instances. Accordingly, we apply 500 sub-iterations for the *2-opt* and *swap* moves and 1,000 sub-iterations for $Replace_{1-1}$ and $Replace_{1-2}$ moves in each restart.

Aspiration criterion

The best solution found is selected as an aspiration criterion in this approach where tabu status of a solution attribute is overridden if a tabu-active move improves the incumbent solution.

5.3 Computational Results

In our computational experiments, all problem instances were solved by using the solver CPLEX version 12.8.0, which is implemented in IBM ILOG Optimization Studio. Heuristic algorithms were coded in Java programming language using the NetBeans IDE 8.1. All of the computational studies are conducted on an Intel(R) Core(TM) i7-477OS CPU @3.10GHz, 16GB RAM Windows 10 computer.

5.3.1 Results for Random Instances

We first present the results obtained within the 2-hour by the CPLEX solver as benchmarks. Since the problem is not solvable for large instances, some of the solutions have positive optimality gaps. The solutions of the MSTS heuristic are then compared with those found by CPLEX in order to measure the quality of our approach, in terms of the solution time and objective value.

In the Table 5.3, best CPLEX solution (z*), percent gap from the best lower bound, solution time (CPU time in seconds), travel time and assessment time (in hours) and the number of nodes visited are presented. In addition to these values, sequence of the nodes and the sample size values assigned to each selected node is provided in Table B.1 in Appendix. B

Table 5.3: MIP results with CPLEX solver

Instance							
#	**Name**	**z***	**Gap %**	**CPU**	**# visited**	**Travel Time**	**Assessment Time**
1	30N_10C_tmax3_geo	0.396	0.00	1.9	4	1.69	1.28
2	30N_10C_tmax3_hier	0.396	0.00	8.3	4	1.69	1.28
3	30N_10C_tmax3_rand	0.396	0.00	12.5	4	1.54	1.46
4	30N_10C_tmax5_geo	0.697	0.00	2.3	7	3.34	1.66
5	30N_10C_tmax5_hier	0.697	0.00	446.2	7	2.91	2.08
6	30N_10C_tmax5_rand	0.798	0.00	2.0	8	2.75	2.22
7	30N_10C_tmax7_geo	0.898	0.00	78.5	9	4.53	2.40
8	30N_10C_tmax7_hier	0.998	0.00	78.4	10	4.60	2.38
9	30N_10C_tmax7_rand	0.998	0.00	3.8	10	3.79	3.16
10	30N_15C_tmax3_geo	0.263	0.00	5.9	4	1.71	1.24
11	30N_15C_tmax3_hier	0.263	0.00	9.8	4	1.65	1.33
12	30N_15C_tmax3_rand	0.263	0.00	11.2	4	1.65	1.33
13	30N_15C_tmax5_geo	0.463	0.00	23.0	7	2.99	2.00
14	30N_15C_tmax5_hier	0.530	0.00	8.1	8	2.99	2.00
15	30N_15C_tmax5_rand	0.530	0.00	6.0	8	2.83	2.14
16	30N_15C_tmax7_geo	0.664	0.00	64.0	10	4.48	2.52
17	30N_15C_tmax7_hier	0.731	0.00	128.0	11	4.13	2.86
18	30N_15C_tmax7_rand	0.731	0.00	183.2	11	4.05	2.94
19	50N_10C_tmax3_geo	0.396	0.00	19.3	4	1.87	1.12
20	50N_10C_tmax3_hier	0.396	0.00	232.9	4	1.65	1.34
21	50N_10C_tmax3_rand	0.396	0.00	265.1	4	1.65	1.34
22	50N_10C_tmax5_geo	0.697	0.00	39.5	7	3.24	1.76
23	50N_10C_tmax5_hier	0.797	0.00	2631.9	8	2.94	2.03
24	50N_10C_tmax5_rand	0.797	0.00	410.5	8	3.05	1.95
25	50N_10C_tmax7_geo	0.898	0.00	2645.1	9	4.84	2.16
26	50N_10C_tmax7_hier	0.998	0.02	7202.6	10	4.11	2.88
27	50N_10C_tmax7_rand	0.998	0.01	7203.7	10	3.99	2.96
28	50N_10C_tmax10_geo	0.999	0.01	7206.8	10	5.44	4.56
29	50N_10C_tmax10_hier	0.999	0.00	2714.8	10	3.84	6.16
30	50N_10C_tmax10_rand	0.999	0.00	7201.7	10	4.13	5.84
31	50N_15C_tmax3_geo	0.263	0.00	52.5	4	1.87	1.12
32	50N_15C_tmax3_hier	0.263	0.00	459.0	4	1.66	1.33
33	50N_15C_tmax3_rand	0.263	0.00	530.0	4	1.66	1.33
34	50N_15C_tmax5_geo	0.463	3.35	7200.2	7	2.84	2.16
35	50N_15C_tmax5_hier	0.530	2.02	7200.2	8	2.90	2.08
36	50N_15C_tmax5_rand	0.530	0.00	5491.8	8	2.88	2.08
37	50N_15C_tmax7_geo	0.664	6.86	7200.2	10	4.07	2.91
38	50N_15C_tmax7_hier	0.797	0.00	167.0	12	4.11	2.88
39	50N_15C_tmax7_rand	0.731	7.79	7201.9	11	3.96	3.04
40	50N_15C_tmax10_geo	0.931	6.21	7201.4	14	5.76	4.24

Table 5.3 MIP results with CPLEX solver – continued

Instance							
#	Name	z*	Gap %	CPU	# visited	Travel Time	Assessment Time
41	50N_15C_tmax10_hier	0.998	0.01	7200.3	15	5.36	4.64
42	50N_15C_tmax10_rand	0.998	0.01	7203.5	15	5.74	4.26
43	50N_20C_tmax5_geo	0.346	7.87	7200.4	7	2.79	2.16
44	50N_20C_tmax5_hier	0.446	0.00	42.2	9	2.83	2.13
45	50N_20C_tmax5_rand	0.396	4.08	7200.2	8	2.75	2.25
46	50N_20C_tmax7_geo	0.547	0.00	1741.1	11	4.17	2.83
47	50N_20C_tmax7_hier	0.597	0.88	7200.4	12	3.80	3.20
48	50N_20C_tmax7_rand	0.547	12.36	7200.8	11	3.79	3.20
49	50N_20C_tmax10_geo	0.747	5.23	7200.4	15	5.86	4.11
50	50N_20C_tmax10_hier	0.848	1.54	7203.3	17	5.54	4.40
51	50N_20C_tmax10_rand	0.848	1.82	7202.0	17	5.69	4.31
52	50N_25C_tmax5_geo	0.316	0.00	1704.0	8	2.84	2.16
53	50N_25C_tmax5_hier	0.356	0.00	81.5	9	2.83	2.13
54	50N_25C_tmax5_rand	0.356	0.00	65.7	9	2.79	2.19
55	50N_25C_tmax7_geo	0.476	0.00	45.3	12	4.08	2.88
56	50N_25C_tmax7_hier	0.476	5.47	7200.4	12	3.73	3.23
57	50N_25C_tmax7_rand	0.476	4.99	7200.4	12	3.77	3.23
58	50N_25C_tmax10_geo	0.637	3.06	7202.1	16	5.64	4.35
59	50N_25C_tmax10_hier	0.717	1.29	7200.4	18	5.57	4.43
60	50N_25C_tmax10_rand	0.717	0.00	6135.2	18	5.54	4.46
61	75N_15C_tmax5_geo	0.530	0.00	5046.5	8	3.08	1.91
62	75N_15C_tmax5_hier	0.597	8.09	7200.3	9	2.43	2.56
63	75N_15C_tmax5_rand	0.664	0.00	289.7	10	2.57	2.40
64	75N_15C_tmax7_geo	0.664	16.60	7204.0	10	3.89	3.10
65	75N_15C_tmax7_hier	0.798	12.79	7201.4	12	3.46	3.53
66	75N_15C_tmax7_rand	0.798	14.28	7202.7	12	3.46	3.52
67	75N_15C_tmax10_geo	0.931	7.19	7203.3	14	5.76	4.23
68	75N_15C_tmax10_hier	0.931	7.20	7207.9	14	5.25	4.75
69	75N_15C_tmax10_rand	0.998	0.01	7202.7	15	4.32	5.66
70	75N_25C_tmax5_geo	0.316	9.56	7201.6	8	2.68	2.31
71	75N_25C_tmax5_hier	0.436	0.00	3071.6	11	2.35	2.64
72	75N_25C_tmax5_rand	0.396	4.08	7200.3	10	2.34	2.64
73	75N_25C_tmax7_geo	0.476	6.49	7201.8	12	3.86	3.11
74	75N_25C_tmax7_hier	0.597	0.00	1989.4	15	3.08	3.87
75	75N_25C_tmax7_rand	0.557	4.76	7200.3	14	3.31	3.68
76	75N_25C_tmax10_geo	0.717	1.38	7201.5	18	5.68	4.31
77	75N_25C_tmax10_hier	0.797	2.58	7201.0	20	5.19	4.96
78	75N_25C_tmax10_rand	0.797	3.21	7200.9	20	5.03	4.96
Averages	Geo	0.592	2.84	3765.1	9.4	3.81	2.64
	Hier	0.653	1.61	3511.1	10.5	3.48	2.97
	Rand	0.653	2.21	4116.5	10.4	3.42	3.02
	Overall	0.633	2.22	3797.5	10.1	3.57	2.87

Table 5.3 reports the solutions within two hours time limitation. We restrict the solution time as two hours, since MIP model is not able to find optimal solution for many instances with CPLEX solver within reasonable time interval. When the instances that cannot be solved optimally in two hours are solved without time limit, MIP model can not provide an optimal solution even within 72 hours.

According to presented solutions, objective function value is primarily affected by the cluster size and time limitation, which determines the proportion of nodes visited. Additionally, cluster type has an effect the objective function value. For the same cluster size and time limitation, number of nodes visited depends on the clustering structure of the network. Impact of the cluster type on objective value can be seen in Table 5.3 where the average objective value is smaller in *geographical* instances than the average objective value of the *hierarchical* and *random* instances. *Geographical* instances have the highest average percent gap value of 2.84% where the overall average percent gap is 2.22%. In overall, 36 out of 78 instances are not able to be solved optimally within two hours time limitation. These are commonly larger-sized instances. Only 4 out of 18 instances with 75 nodes were solved optimally. These instances have smaller time limitations (Tmax values). When average travel times in CPLEX solutions are considered, we can conclude that constructed routes are longer in *geographical* instances. This observation provides a reason for why *geographical* instances have worse objective value and average percent gap than the others. As a result of having longer travel times, time spent in selected nodes for assessment is smaller in geographically clustered instances. Additionally, experiments point out that computational complexity of the network increases with its size. However, the number of clusters has larger impact on computational time than the number of nodes. For example, 8 out of 12 instances with 50-node and 10-cluster are solved optimally within 20 minutes on average, whereas the number of instances that are solved optimally is 5 out of 12 instances with 50-node 15-cluster within 22 minutes on average.

Benchmark solutions point out that the PDNARP is solvable in reasonable solution times for only small-sized networks. On the other hand, for the more complex networks (larger number of nodes and larger cluster sizes), MIP model may not be able to solve this problem within reasonable time. Especially in the immediate response phase, assessment should be completed quickly. Therefore, this solution method can not guarantee the optimal solution for real-sized instances.

Three versions of the MSTS heuristic are tried in order to compare performances and to obtain the most promising set of parameters. These versions differ in terms of the restart strategies.

The first version of MSTS is deterministic, where only div_2 procedure is used during the algorithm (denoted by $MSTS_d$). The second version is hybrid, where both div_1 and div_2 procedures are used in the algorithm (denoted by $MSTS_h$). The third version is random, where only div_1 procedure is used during the algorithm (denoted by $MSTS_r$). For each heuristic, we can calculate the percent gap of the solution as:

$$\frac{z^*(i) - z_{heur}(i)}{z^*(i)} \tag{5.1}$$

where $z^*(i)$ denotes the best CPLEX solution and $z_{heur}(i)$ denotes the heuristic solution of instance i.

In Tables B.3, B.4 and B.5 in Appendix B, the percent gap values of the heuristic solutions from the best solutions found by CPLEX solver are summarized for the different tabu tenure alternatives varying from 3 to 10 for each local search move. In the tables, first two columns define the instances where instance name includes number of nodes (*N*), number of clusters (*C*), total travel and assessment time limit (*tmax*) and cluster types (*geo, hier* and *rand*), and remaining columns includes $\%$ gap values for the corresponding tabu tenure (as an example, the setting $tabuTenure_{2opt}$=3, $tabuTenure_{swap}$=3, $tabuTenure_{remove}$=3 and $tabuTenure_{insert}$=3 is denoted by t_3). In Tables B.3, B.4 and B.5, diversification parameter is taken as 0.45.

Table B.3 shows the solutions of the $MSTS_d$, where at least 63 out of 78 instances are able to find best CPLEX solutions for all tabu tenure alternatives. Bold values shows the best percent gaps found with different tabu tenures for each instance. In Figures 5.6, 5.7, 5.8 and 5.9, average percent gap values for three MSTS alternatives are summarized for eight tabu tenure alternatives. Average gap value is smaller than or equal to 0.662$\%$ for all tabu tenure alternatives where the maximum gap is 14.37$\%$. There exists four instances where optimality gap is larger than 0.02$\%$ for any tabu tenure alternatives (see instances 4, 53, 71 and 76 in Table B.3 in Appendix B). Since we are measuring the accuracy (our primary objetive) as a proportion of selected nodes and our network design does not include very large number of clusters (at most 25), selecting only one missing node increases percent gap a lot. The effect of missing one node in the solution increases in the small-sized instances. For example, the highest percent gap (14.36%) belongs to a small-sized instance ($30N\_10C\_tmax5\_geo$). Additionally, computational results shows that, $MSTS_d$ is able to find better solutions than best cplex solutions where the CPLEX solver is not able to find optimal solution within two hours time limitation (see instances 26, 48, 66 and 68 in Table B.3 in Appendix B). For the geographical instances, many tabu tenure alternatives provide same solutions, and the best solutions belong to tabu tenures t_6, t_7, t_8 and t_9. On the other hand, solutions of the hierarchical instances point out the effectiveness of small tabu tenures in hierarchical cluster types. Solutions of the random instances are better than the solutions of geographical and hierarchical ones in all tabu tenure alternatives. This observation, shows the strength of $MSTS_d$ heuristic on randomly clustered networks. Moreover, among the tabu tenure alternatives, larger ones provides better percent gap values for randomly clustered instances.

In order to improve solution quality of the MSTS algorithm, we also implement a version where random initial starts are also involved in the procedure. We implement a random restart starting from 10th restart with increments of 10 up to 200. Table B.4 shows the solutions of this hybrid method. According to Table B.4, at least 67 out of 78 instances are able to find best CPLEX solutions for all tabu tenure alternatives which highlights the improvement gathered by the implementation of random diversification strategy. In this case, average gap value is smaller than or equal to 0.59% for all tabu tenure alternatives where the maximum gap is 14.37%. Although, results of the $MSTS_h$ does not improve the worst solution found in $MSTS_d$, average gap % decreases with the addition of random restarts. There exist only three instances where optimality gap is not able to decreased under 0.01% for any tabu tenure alternatives (see instances 4, 71 and 76) . Additionally, as in $MSTS_d$, there are five instances where $MSTS_h$ is able to find better solutions than best CPLEX solutions (see instances 26, 48, 65, 66 and 68). For the geographical instances, the best average gap values belong to tabu tenures 2, 3 and 10 (see Table 5.6). When solutions

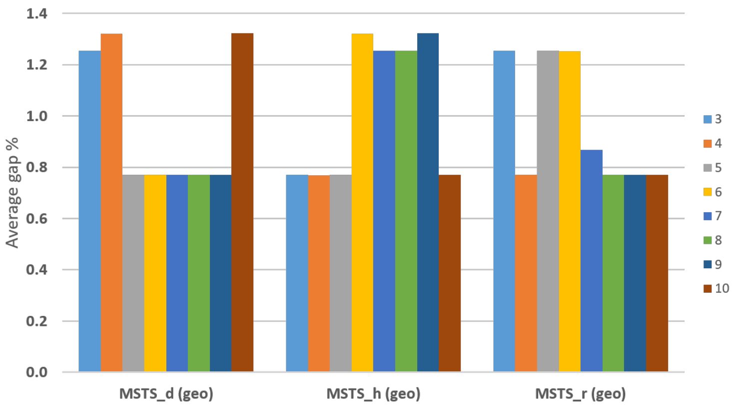

Figure 5.6: Change in average gap % of geographical instances according to alternative MSTS versions

are compared with the average percent gap obtained by $MSTS_d$, we may conclude that randomness does not improve heuristic solutions of the geographical instances, whereas a significant improvement was observed in the $MSTS_h$ solutions of the hierarchical instances where average percent gap decreases from 0.515 to -0.048. That means $MSTS_h$ provides at least best CPLEX solution for all of the hierarchical instances (see Table 5.7). This improvement is obtained in large tabu tenure alternatives (t_9 and t_{10}). According to Table 5.8, $MSTS_h$ does not have a significant improvement on the random instances as in geographical instances, only the best tabu tenure alternative, which is t_3, is different in $MSTS_h$.

After realizing the impact of the diversification based on randomization, we also try the pure random restart strategy. Table B.5 shows the solutions of the pure random start version. Applying pure random restart strategy does not improve either solutions of $MSTS_d$ or $MSTS_h$ further, where the average gap is at most 0.58%, and the maximum gap is 14.36%.

After the analyses made on the diversification strategy effects and tabu tenure alternatives, we also analyze the results of using different diversification parameters, which determine the proportion of nodes that will remain in the following restart. The change in the average percent gap value according to changing diversification parameter is given in Figure 5.10 From the preliminary observations, there are many common nodes selected by both the MSTS heuristic and CPLEX solver. For this reason, we keep at least 50% of the currently selected nodes in the following restart.

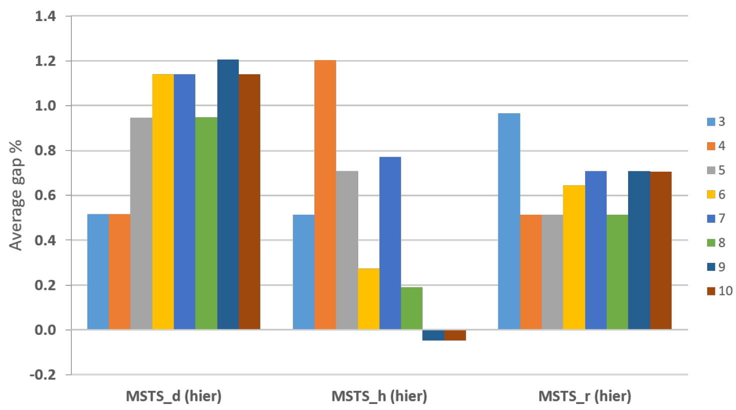

Figure 5.7: Change in average gap % of hierarchical instances according to alternative MSTS versions

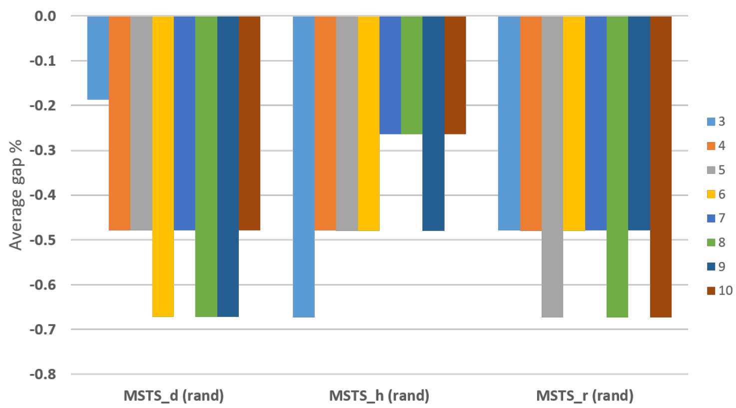

Figure 5.8: Change in average gap % of random instances according to alternative MSTS versions

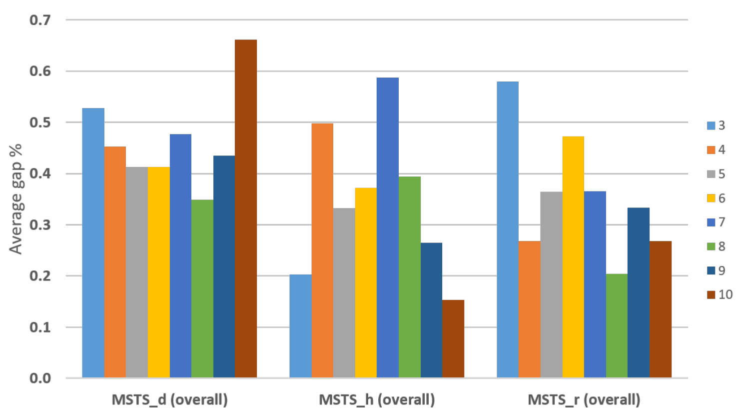

Figure 5.9: Change in average gap % of all instances according to alternative MSTS versions

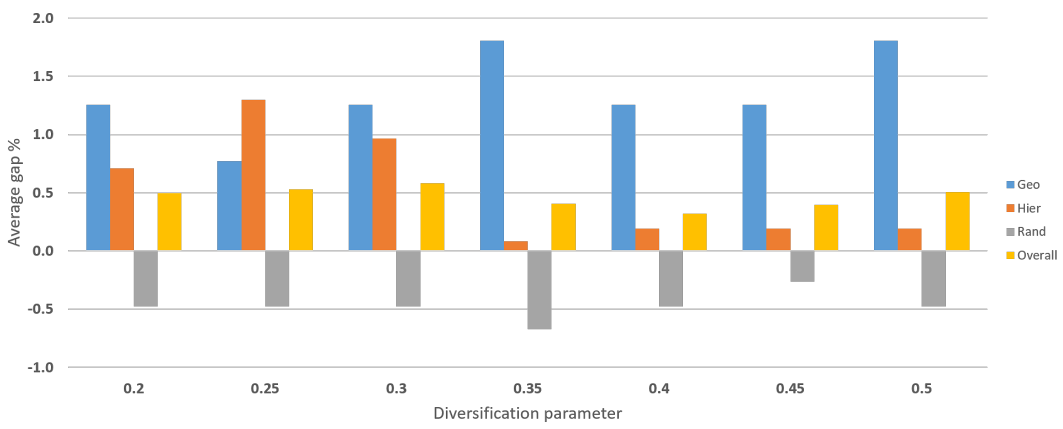

Figure 5.10: Change in average gap % according to diversification parameter

Our trials include seven diversification parameter alternatives (same parameter is used for both div_1 and div_2), and the percent gap values of the $MSTS_h$ solutions start from the third column of Table B.6. By changing the diversification parameter, in most of the instances, common solutions are obtained. The worst average percent gap belongs to diversification parameter of 0.3 (represented by $p_{0.30}$) where average gap values vary in between 0.32 and 0.53. There is not any outstanding parameter value which is better than the others in all of the instances. $p_{0.25}$ provides better average percent gap value for geographical instances, $p_{0.35}$ provides better value for hierarchical and random instances. In overall, smallest percent gap value is obtained by the diversification parameter of $p_{0.40}$.

All three versions of the MSTS are solvable within seconds. There is no significant difference among the MSTS versions in terms of computational time. While small-sized instances are able to be solved within 10 seconds, the large-sized instances are solved in a minute. Recorded minimum CPU time is 4.8 seconds, and maximum CPU time is 63.7 seconds where the average is equal to 21.0 seconds. Objective function values and CPU times for tabu tenure trials (with $p_{0.45}$) are available in Table B.2 in Appendix B. Additionally, computational time increases with the number of clusters. An example figure that shows the impact of cluster size on computational time is provided for the random instances with 50 nodes in Figure 5.11.

There exists three instances (4, 71 and 76) where optimality gap cannot be decreased under 0.01% for any tabu tenure and diversification parameter alternatives. Best per-

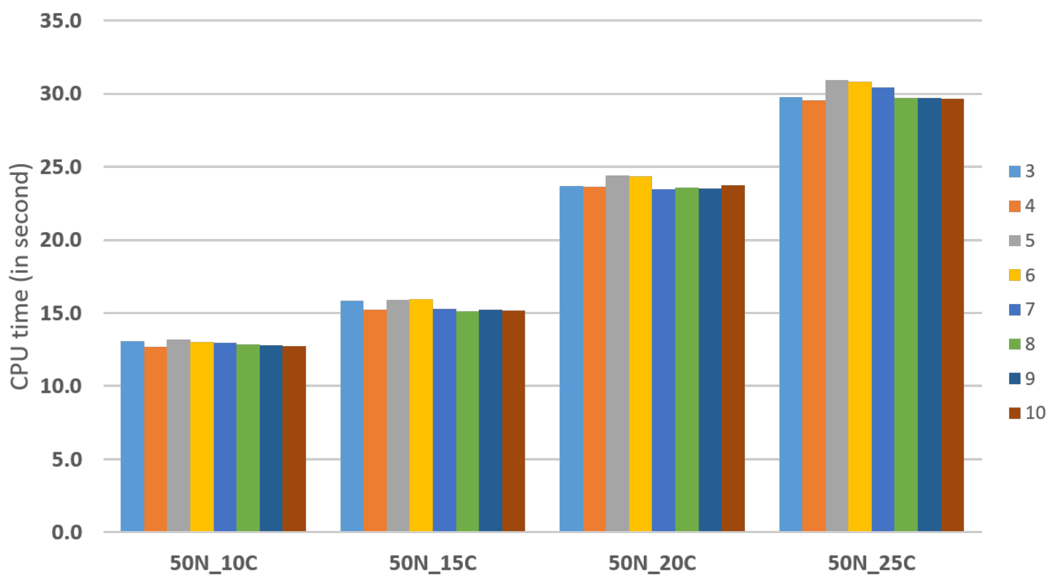

Figure 5.11: Change in average CPU time according to cluster size

cent gap values are 14.365%, 11.258% and 5.590%, respectively. For these instances, all versions and parameter alternatives of MTST heuristic find a solution with one missing node according respect to the best CPLEX solution. The percent gap values are high due to the primary objective (maximizing the proportion of selected nodes) of the PDNARP. These instances are not common in terms of node size, cluster size, time limitation or cluster type. So, the underlying reason for lack of finding one more solution may be restricted solution space due to the tight time limitation for the corresponding instances. In Figure 5.12, routes constructed by CPLEX solver and $MTST_h$ heuristic are provided for the instance ($30N\_10C\_tmax5\_geo$) where the difference among node selections are seen. CPLEX solution is denoted by black and $MTST_h$ heuristic solution is denoted by orange. Travel time in CPLEX solution is 3.34 hours and assessment time is 1.42 hours. Therefore, the total time spent is equal to 4.76 hours. On the other hand, in $MTST_h$ heuristic solution travel time is 2.68 hours and sampling time is 1.92 hours. Hence the total time spent is equal to 4.60 hours. In the heuristic solution, it is not possible to add one more node to route without exceeding $tmax$. MSTS heuristic is stuck in a local optimum, so it cannot find a path with same number of nodes. Similar observations are made for instances 71 and 76 as well.

Computational experiments on the test instances show that adding a random start to

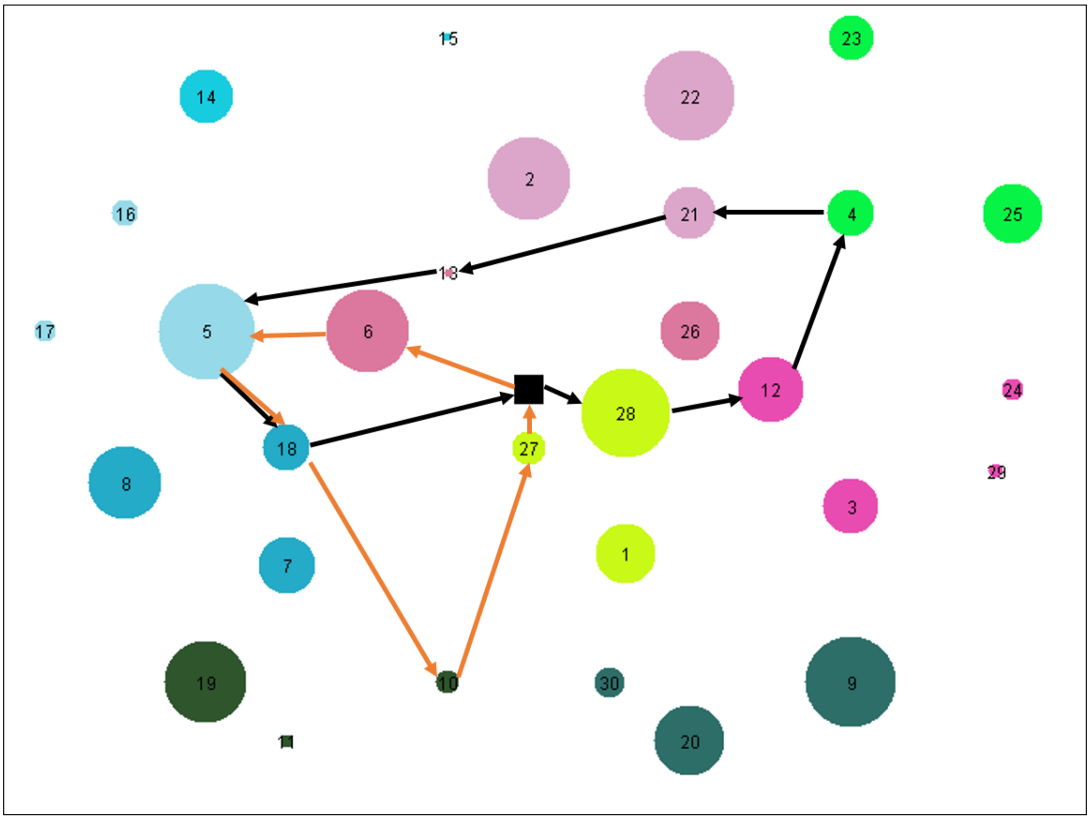

Figure 5.12: Routes constructed by CPLEX solver and $MTST_h$ heuristic for the instance # 4 ($30N\_10C\_tmax5\_geo$), where black and orange routes represents results of the CPLEX and MSTS, respectively

the algorithm improves the final solution. $MTST_h$ heuristic is an efficient solution approach for the PDNARP, which provides near-optimal solutions within a minute. It is able to find best solution in at least 80% of the instances for each parameter setting.

5.3.2 Case Study Results

In Table 5.4 the best CPLEX solutions (within two hours) for the case study instances are presented. These solutions show that, PDNARP is not solvable for the real-sized instances. According to solutions presented in Table 5.4, the minimum optimality gap is 4.6 % and the average optimality gap is 25.07 % for the case study instances.

Since $MSTS_h$ provides the best solutions among the compared versions of the presented heuristic, we implemented the $MSTS_h$ in the case study. Percent gap values according to changing parameter settings are summarized in Table 5.5. Additionally, objective function values and CPU times of both CPLEX and $MSTS_h$ algorithm are provided in Table C.2 in Appendix C. $MSTS_h$ is able to find the best CPLEX solution for the real-sized instances within 90 seconds.

Table 5.4: CPLEX solutions of the Van Case Study Instances

#	Name	z*	Gap %	CPU	# visited	Travel Time	Sampling Time
1	93N_16C_tmax10	0.434	45.62	7200.5	7	7.83	2.17
2	93N_16C_tmax15	0.747	31.34	7200.4	12	11.52	3.47
3	93N_16C_tmax20	0.936	6.73	7202.4	15	13.90	6.10
4	93N_23C_tmax10	0.388	28.02	7200.5	9	7.26	2.73
5	93N_23C_tmax15	0.606	29.43	7211.5	14	11.42	3.57
6	93N_23C_tmax20	0.693	39.56	7203.3	16	14.80	5.19
7	93N_23C_tmax25	0.955	4.60	7205.8	22	18.53	6.47
8	93N_29C_tmax10	0.306	23.98	7200.5	9	7.49	2.50
9	93N_29C_tmax15	0.445	36.79	7204.8	13	11.28	3.71
10	93N_29C_tmax20	0.687	16.25	7226.7	20	14.99	5.01
11	93N_29C_tmax25	0.825	13.46	7204.5	24	18.38	6.62
	Average	0.638	25.07	7205.5	15	12.49	4.32

In the analyses made on the test instances, diversification parameter selection does not direct us select to one of the tried diversification parameters. For this reason, we try different parameter alternatives on the case study instances. Furthermore, the tabu tenure alternatives are also analyzed. In Table 5.5, diversification parameter is denoted by p_{div}. For each instance parameter alternatives of $p_{0.40}$, $p_{0.45}$ and $p_{0.50}$ are tried for each tabu tenure alternatives. According to the solutions $p_{div} = 0.40$ provides the best solutions for all case study instances where $MSTS_h$ solutions are 4.198% better than the best CPLEX results on average. There is not any instance where $MSTS_h$ is worse than CPLEX solver. Best improvement is obtained in instance 6, where $MSTS_h$ algorithm solution is 18.861% better than the best CPLEX solution. Case study results show that, presented $MSTS_h$ is able to find best CPLEX solutions found within two hour time limitation in real-sized problems within a minute.

Table 5.5: $MSTS_h$ solutions of the Van Case Study Instances with different tabu tenures and diversification parameters

Instance		% Gap								
#	Name	p_{div}	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
1	93N_16C_tmax10	$p_{0.4}$	0.003	0.003	0.007	0.003	**0.000**	**0.000**	0.003	0.003
		$p_{0.45}$	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**
		$p_{0.5}$	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**	**8.670**
2	93N_16C_tmax15	$p_{0.4}$	-8.367	**-8.375**	-8.367	-8.367	0.008	-8.375	-8.367	-0.005
		$p_{0.45}$	-8.371	-8.371	-8.371	**-8.375**	**-8.375**	**-8.375**	-8.371	**-8.375**
		$p_{0.5}$	-0.005	**-8.375**	-8.371	-8.371	-8.371	**-8.375**	-0.005	-8.367
3	93N_16C_tmax20	$p_{0.4}$	-6.653	-6.656	**-6.664**	-6.663	**-6.664**	-6.655	-6.653	**-6.664**
		$p_{0.45}$	-6.653	-6.653	-6.669	-6.656	-6.656	**-6.671**	-6.653	-6.665
		$p_{0.5}$	**-6.671**	-6.657	-6.661	**-6.671**	-6.653	-6.661	**-6.671**	-6.659
4	93N_23C_tmax10	$p_{0.4}$	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**
		$p_{0.45}$	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**
		$p_{0.5}$	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**
5	93N_23C_tmax15	$p_{0.4}$	7.182	0.006	7.182	7.184	**0.000**	7.184	7.182	**0.000**
		$p_{0.45}$	0.005	**0.000**	0.004	7.184	7.184	7.182	0.005	7.182
		$p_{0.5}$	7.189	7.182	7.184	**7.180**	7.185	7.184	7.189	7.184
6	93N_23C_tmax20	$p_{0.4}$	**-18.861**	-18.850	-12.563	-12.571	-12.560	-18.852	**-18.861**	-12.562
		$p_{0.45}$	-12.561	-18.852	-12.568	**-18.855**	-18.854	-12.572	-12.561	-12.563
		$p_{0.5}$	-18.850	-18.850	-12.571	**-18.852**	-18.850	-12.568	-18.850	-12.565
7	93N_23C_tmax25	$p_{0.4}$	0.014	**0.004**	0.024	0.019	0.007	0.005	0.014	0.014
		$p_{0.45}$	0.013	**0.007**	**0.007**	0.020	0.019	0.013	0.013	0.008
		$p_{0.5}$	0.019	**0.008**	0.019	0.018	0.019	0.023	0.019	0.014
8	93N_29C_tmax10	$p_{0.4}$	0.003	**0.000**	0.001	0.001	0.003	**0.000**	0.003	**0.000**
		$p_{0.45}$	0.001	0.001	0.001	**0.000**	**0.000**	0.001	0.001	0.001
		$p_{0.5}$	0.003	**0.000**	**0.000**	0.001	0.003	0.003	0.003	0.003
9	93N_29C_tmax15	$p_{0.4}$	0.000	**-15.541**	**-15.544**	**-15.544**	-7.760	-7.765	0.000	-7.761
		$p_{0.45}$	-7.761	**-15.544**	**-15.544**	-7.765	-7.761	**-15.544**	-7.761	-7.767
		$p_{0.5}$	-7.763	-7.763	-7.761	-7.761	-7.761	-7.761	-7.763	**-7.767**
10	93N_29C_tmax20	$p_{0.4}$	-0.002	-0.006	**-5.033**	-5.032	-0.007	-0.003	-0.002	**-5.033**
		$p_{0.45}$	-0.009	**-5.033**	-0.006	-0.009	-0.001	**-5.033**	-0.009	-0.009
		$p_{0.5}$	**-5.033**	-5.031	-0.009	**-5.033**	-0.009	**-5.033**	**-5.033**	-0.009
11	93N_29C_tmax25	$p_{0.4}$	**-4.178**	-4.176	-4.176	**-4.178**	-4.176	0.002	**-4.178**	**-4.178**
		$p_{0.45}$	-4.177	**-4.179**	0.002	-4.178	-4.178	-4.176	-4.177	-4.178
		$p_{0.5}$	**-4.179**	-4.176	**-4.179**	-4.176	**-4.179**	**-4.179**	**-4.179**	-4.178
	Averages	$p_{0.4}$	-2.806	**-4.872**	-4.103	-4.105	-2.832	-3.133	-2.806	-3.290
		$p_{0.45}$	-2.804	**-4.542**	-3.134	-2.724	-2.723	-3.319	-2.804	-2.154
		$p_{0.5}$	-2.420	-3.181	-2.153	**-3.182**	-2.723	-2.609	-2.420	-2.152
		Overall	-2.677	**-4.198**	-3.130	-3.337	-2.759	-3.020	-2.677	-2.532

CHAPTER 6

CONCLUSION

World is a target of many natural and human-inflicted disasters some of which have catastrophic impacts on the affected countries and the on lives of millions of people. In the immediate aftermath of a disaster, needs assessment operations are performed first to understand the effect of the disaster so that effective response can be provided. An effective needs assessment process provides reliable estimations on the beneficiaries' needs.

Each disaster may require different needs assessment plan due to its unique dynamics (such as type, magnitude, population characteristics in the impact region, etc.). To start response operations as soon as possible, rapid needs assessment activities should be completed within limited time. Thus, conducting surveys in the whole affected region to collect information is not possible. Sampling techniques are used to estimate beneficiaries' needs within limited time. Reliable estimates depend on (1) how well the sampled data reflects the actual situation (accuracy) and (2) how certain the obtained information is (precision). An accurate estimation can be obtained by conducting surveys in as many sites as possible. On the other hand, spending more time at each visited site provides more precise estimations. Time limitation in rapid needs assessment process causes an inherent trade-off between allocated time for traveling among sites and colloecting information in each visited site.

In this thesis, we define the Post-Disaster Needs Assessment Routing Problem and present a mathematical model which involves the trade-off between accuracy and precision of the sampling process. Our motivation in this study is providing a quick and efficient assessment plan for the needs assessment, to support relief agencies to develop effective sampling plans to obtain reliable estimates on collected information. To the best of our knowledge, there exist no study in IE/OR literature, which presents a mathematical model that addresses routing decisions of efficient routing scheme of assessment teams together with the time spent in visited sites in order to collect reliable information.

Throughout this thesis, a mixed integer programming formulation is presented for the PDNARP. Moreover, to find near-optimal solutions for large-scale networks quickly, we develop an effective tabu search metaheuristic. Multi-Start Tabu Search Heuristic is developed as a solution approach and its three versions ($MSTS_d$, $MSTS_h$ and $MSTS_r$) are presented. During computational experiments, two instance sets are used: (i) random instance set modified from well-known Solomon's instances and (ii) case study instance set based on real-world data from the 2011 Van Earthquake.

In the computational study, solution performance of the MSTS versions and mixed integer program are compared. Moreover, effects of network characteristics (number of nodes and clusters in network, cluster types and total time limitation) and parameter settings (tabu tenures, diversification parameter) used in the proposed MSTS algorithm are analyzed. In these experiments, we show that mixed integer programming model is not able to find optimal solution for the realistic problem instances within reasonable time limit (72 hours). Moreover, experiments point out that computational complexity of the network increases with its size and the number of clusters has also a significant effect on computational time. The proposed MSTS algorithm is able to find high-quality solutions for realistic problem instances efficiently. The MSTS heuristic is solvable within 90 seconds for realistic problem instances. MSTS versions are able to find better solutions than the best CPLEX for the solutions with optimality gap. Diversification strategies improves objective value of the MSTS heuristic. Effectiveness of the heuristic parameters may change with respect to cluster type of the network. Computational experiments show that the best diversification parameter for *hierarchical* and *random* networks provides the worst average percent gap value for the *geographical* networks. Additionally, setting large tabu tenure value provides best heuristic solution for *geographical* and *random* networks, however, same tabu tanure value does not give the best heuristic solution for *hierarchical* networks with respect to other tabu tenure alternatives. Finally, we found that the MSTS version with hybrid restart strategy performs better than the deterministic and random versions of MSTS.

Since in the IE/OR literature, there are only a few studies on the post-disaster needs assessment operations, there can be many future research directions. Firstly, our study assumes that there is a single assessment team. Multiple-team version of this study is the first possible research direction which is also easy to adapt to our solution approach. Additionally, in this study we focus on a deterministic needs assessment routing problem, whereas immediate aftermath of a disaster may involve in various uncertain parameters. These uncertainties can be included in the assessment plan within a dynamic framework where the decisions on where to visit and when to stop information collection at visited site can be made during the assessment process. In the dynamic problem setting, the estimated prevalence value is updated by assessment teams after each visit and survey made. Updates on the estimates will change the sample size requirement for the remaining visits. However, in this dynamic approach, assessment teams need to change their plan at each step which may require well-trained personnel or volunteers in assessment teams. Another future direction may be adapting alternative performance metrics to our solution approach in order to measure the reliability (accuracy and precision) of the estimations. Furthermore, a bi-criteria approach can be studied which includes accuracy and precision objectives.

In this study, since we focus on the rapid assessment phase, it is assumed that relief distribution starts after the completion of the needs assessment. However, this problem setting is also applicable in detailed or continual assessment phases where time limitation is wider than the rapid assessment phase, and assessment operations performed together with relief distribution operations. Hence, needs assessment problem can be combined with relief distribution decisions where both operations starts together and total time limitation affects both needs assessment and relief distribution operations. Moreover, for the detailed or continual phase of the assessment where assessments are performed in wider time periods, this problem can be modeled with objective of minimizing total time spent for traveling and information collection which ensures predetermined level of accuracy and precision.

REFERENCES

[1] A hierarchical clustering and routing procedure for large scale disaster relief logistics planning.

[2] Morbidity and mortality weekly report. Technical report, Centers for Disease Control (US). Epidemiology Program Office, US Dept. of Health and Human Services, 2000.

[3] Methodology rapid assessment for humanitarian assistance. Technical report, REDLAC, REDLAC, Panama, 2006.

[4] Guidelines for assessment in emergencies. Technical report, ICRC / International Federation of the Red Cross and Red Crescent Societies, Geneva, Switzerland, 2008.

[5] Alternative sampling designs for emergency settings: A guide for survey planning, data collection and analysis. Technical report, FANTA-2 Project, Food and Nutrition Technical Assistance Project II (FANTA-2) Academy for Educational Development, 2009.

[6] Rapid assessment sampling in emergency situations. Technical report, UNICEF, Thailand: UNICEF APSSC, 2010.

[7] Qualitative and quantitative research techniques for humanitarian needs assessment. Technical report, ACAPS, 2012.

[8] Sampling methods and sample size calculation for the smart methodology. 01 2012.

[9] How sure are you? judging quality and usability of data collected during rapid needs assessments. Technical report, ACAPS, 2013.

[10] E. Angelelli, R. Mansini, and M. Vindigni. Exploring greedy criteria for the

dynamic traveling purchaser problem. *Central European Journal of Operations Research*, 17(2):141, 2009.

[11] C. Archetti, D. Feillet, A. Hertz, and M. G. Speranza. The capacitated team orienteering and profitable tour problems. *Journal of the Operational Research Society*, 60(6):831–842, 2009.

[12] B. An exact -constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits.

[13] E. Balas. The prize collecting traveling salesman problem: Ii. polyhedral results. *Networks*, 25(4):199–216, 1995.

[14] B. Balcik. Site selection and vehicle routing for post-disaster rapid needs assessment. *Transportation research part E: logistics and transportation review*, 101:30–58, 2017.

[15] B. Balcik, B. M. Beamon, and K. Smilowitz. Last mile distribution in humanitarian relief. *Journal of Intelligent Transportation Systems*, 12(2):51–63, 2008.

[16] R. Baldacci, E. Bartolini, and G. Laporte. Some applications of the generalized vehicle routing problem. *Journal of the operational research society*, 61(7):1072–1077, 2010.

[17] Barth. Metaheuristic approach for the clustered vrp.

[18] M. Battarra and Erdo. Exact algorithms for the clustered vehicle routing problem.

[19] Bekta. Formulations and branch-and-cut algorithms for the generalized vehicle routing problem.

[20] A. Benini. A computer simulation of needs assessments in disasters. the impact of sample size, logistical difficulty and measurement error. Technical report, Washington DC, 2012.

[21] D. Bienstock, M. X. Goemans, D. Simchi-Levi, and D. Williamson. A note on the prize collecting traveling salesman problem. *Mathematical programming*, 59(1-3):413–420, 1993.

[22] N. Binkin, K. Sullivan, N. Staehling, and P. Nieburg. Rapid nutrition surveys: how many clusters are enough? *Disasters*, 16(2):97–103, 1992.

[23] S. Boussier, D. Feillet, and M. Gendreau. An exact algorithm for team orienteering problems. *4or*, 5(3):211–230, 2007.

[24] B. Bozkaya, E. Erkut, and G. Laporte. A tabu search heuristic and adaptive memory procedure for political districting. *European Journal of Operational Research*, 144(1):12–26, 2003.

[25] S. E. Butt and T. M. Cavalier. A heuristic for the multiple tour maximum collection problem. *Computers & Operations Research*, 21(1):101–111, 1994.

[26] I.-M. Chao, B. L. Golden, and E. A. Wasil. A fast and effective heuristic for the orienteering problem. *European journal of operational research*, 88(3):475–489, 1996.

[27] W. G. Cochran. *Sampling techniques*. John Wiley & Sons, 2007.

[28] J.-F. Cordeau. A tabu search heuristic for periodic and multi-depot vehicle routing problems.

[29] T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan. Random-restart reactive tabu search algorithm for detection in large-mimo systems. *IEEE Communications Letters*, 14(12):1107–1109, 2010.

[30] M. Dell'Amico, F. Maffioli, and A. Sciomachen. A lagrangian heuristic for the prize collectingtravelling salesman problem. *Annals of Operations Research*, 81:289–306, 1998.

[31] M. Dell'Amico, F. Maffioli, and V. On prize-collecting tours and the asymmetric travelling salesman problem.

[32] M. Fischetti and P. Toth. An additive approach for the optimal solution of the prize collecting traveling salesman problem. *Vehicle routing: Methods and studies*, 231:319–343, 1988.

[33] G. Galindo and R. Batta. Review of recent developments in or/ms research in disaster operations management. *European Journal of Operational Research*, 230(2):201–211, 2013.

[34] M. Gendreau, A. Hertz, and G. Laporte. A tabu search heuristic for the vehicle routing problem. *Management science*, 40(10):1276–1290, 1994.

[35] M. Gendreau, G. Laporte, and S. A tabu search heuristic for the vehicle routing problem with stochastic demands and customers.

[36] M. Gendreau, G. Laporte, and F. Semet. A tabu search heuristic for the undirected selective travelling salesman problem.

[37] G. Ghiani and G. Importa. An efficient transformation of the generalized vehicle routing problem. *European Journal of Operational Research*, 122(1):11–17, 2000.

[38] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers & operations research*, 13(5):533–549, 1986.

[39] F. Glover. Tabu search—part i. *ORSA Journal on Computing*, 1(3):190–206, 1989.

[40] B. L. Golden, L. Levy, and R. Vohra. The orienteering problem. *Naval research logistics*, 34(3):307–318, 1987.

[41] P. Hansen. The steepest ascent mildest descent heuristic for combinatorial programming. In *Congress on numerical methods in combinatorial optimization, Capri, Italy*, pages 70–145, 1986.

[42] M. Haouari and J. C. Siala. A hybrid lagrangian genetic algorithm for the prize collecting steiner tree problem. *Computers & Operations Research*, 33(5):1274–1288, 2006.

[43] R. H. Henderson and T. Sundaresan. Cluster sampling to assess immunization coverage: a review of experience with a simplified sampling method. *Bulletin of the World Health Organization*, 60(2):253, 1982.

[44] A. Hertz and D. de Werra. Using tabu search techniques for graph coloring. *Computing*, 39(4):345–351, 1987.

[45] N. Hu. Tabu search method with random moves for globally optimal design. *International Journal for Numerical Methods in Engineering*, 35(5):1055–1070, 1992.

[46] Y.-C. Huang, H.-T. Yang, and C.-L. Huang. Solving the capacitor placement problem in a radial distribution system using tabu search approach. *IEEE Transactions on power Systems*, 11(4):1868–1873, 1996.

[47] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651 – 666, 2010. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR).

[48] M. G. Kantor and M. B. Rosenwein. The orienteering problem with time windows. *Journal of the Operational Research Society*, 43(6):629–635, 1992.

[49] L. Ke, C. Archetti, and Z. Feng. Ants can solve the team orienteering problem. *Computers & Industrial Engineering*, 54(3):648–665, 2008.

[50] A. Koscalova. Assessment toolkit. Technical report, Medecins Sans Fronteers-OCG, Vienna Evaluation Unit, 2012.

[51] A. Koscalova. Assessment toolkit. Technical report, Medecins Sans Fronteers-OCG, Vienna Evaluation Unit, 2017.

[52] G. Laporte and S. Martello. The selective travelling salesman problem. *Discrete applied mathematics*, 26(2-3):193–207, 1990.

[53] G. Laporte, J. Riera-Ledesma, and Salazar-Gonz. A branch-and-cut algorithm for the undirected traveling purchaser problem.

[54] W.-M. Lin, F.-S. Cheng, and M.-T. Tsay. An improved tabu search for economic dispatch with multiple minima. *IEEE Transactions on power systems*, 17(1):108–112, 2002.

[55] E. Luis, I. S. Dolinskaya, and K. R. Smilowitz. Disaster relief routing: Integrating research and practice. *Socio-economic planning sciences*, 46(1):88–97, 2012.

[56] J. Malilay, W. D. Flanders, and D. Brogan. A modified cluster-sampling method for post-disaster rapid assessment of needs. *Bulletin of the World Health Organization*, 74(4):399, 1996.

[57] H. Millar and M. Kiragu. A time-based formulation and upper bounding scheme for the selective travelling salesperson problem. *Journal of the Operational Research Society*, 48(5):511–518, 1997.

[58] C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*, 7(4):326–329, 1960.

[59] L. Moccia and J.-F. Cordeau. An incremental tabu search heuristic for the generalized vehicle routing problem with time windows.

[60] M. Navidadham, M. Arbabsadeghi, A. A. Bayat, and F. Didehvar. Solving generalized vehicle routing problem by parallel universes and tabu search. In *Computing, Communication and Networking Technologies (ICCCNT), 2015 6th International Conference on*, pages 1–7. IEEE, 2015.

[61] V. H. Nguyen and T. T. T. Nguyen. Approximating the asymmetric profitable tour. *Electronic Notes in Discrete Mathematics*, 36:907–914, 2010.

[62] C.-M. Pintea, C. Chira, D. Dumitrescu, and P. C. Pop. Sensitive ants in solving the generalized vehicle routing problem. *arXiv preprint arXiv:1208.5341*, 2012.

[63] P. Pop. Heuristic algorithms for solving the generalized vehicle routing problem.

[64] P. Pop. New mathematical models of the generalized vehicle routing problem and extensions.

[65] P. Pop and C. Chira. A hybrid approach based on genetic algorithms for solving the clustered vehicle routing problem. In *Evolutionary Computation (CEC), 2014 IEEE Congress on*, pages 1421–1426. IEEE, 2014.

[66] P. Pop and C. Pop-Sitar. A new efficient transformation of the generalized vehicle routing problem into the classical vehicle routing problem. *Yugoslav Journal of Operations Research*, 21(2):187–198, 2011.

[67] P. C. Pop, O. Matei, C. P. Sitar, and C. Chira. A genetic algorithm for solving the generalized vehicle routing problem. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 119–126. Springer, 2010.

[68] N.-H. Quttineh and T. Larsson. Military aircraft mission planning: a generalized vehicle routing model with synchronization and precedence.

[69] R. Ramesh, Y.-S. Yoon, and M. H. Karwan. An optimal algorithm for the orienteering tour problem. *ORSA Journal on Computing*, 4(2):155–165, 1992.

[70] R. Ravi and F. S. Salman. Approximation algorithms for the traveling purchaser problem and its variants in network design. In *European Symposium on Algorithms*, pages 29–40. Springer, 1999.

[71] M. Re. The 10 biggest natural disasters worldwide by economic damage from 1980 to 2016 (in billion u.s. dollars).

[72] M. Re. The 10 most significant natural disasters worldwide by death toll from 1980 to 2017.

[73] J. Riera-Ledesma and Salazar-Gonz. A column generation approach for a school bus routing problem with resource constraints.

[74] A. Seth, D. Klabjan, and P. M. Ferreira. Analyses of advanced iterated tour partitioning heuristics for generalized vehicle routing problems. *Networks*, 61(4):290–308, 2013.

[75] M. Sevaux and S. Hamiltonian paths in large clustered routing problems.

[76] M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations research*, 35(2):254–265, 1987.

[77] Taillard. A tabu search heuristic for the vehicle routing problem with soft time windows.

[78] H. Tang and E. Miller-Hooks. A tabu search heuristic for the team orienteering problem. *Computers & Operations Research*, 32(6):1379–1407, 2005.

[79] M. F. Tasgetiren and A. E. Smith. A genetic algorithm for the orienteering problem. In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, volume 2, pages 910–915. IEEE, 2000.

[80] A. Vervaeck and J. Daniell. Earthquake van – ercis, turkey - 604 dead, large aftershock 5.6 hits van.

[81] T. Vidal, M. Battarra, A. Subramanian, and G. Erdogan. Hybrid metaheuristics for the clustered vehicle routing problem.

[82] Q. Wang, X. Sun, B. L. Golden, and J. Jia. Using artificial neural networks to solve the orienteering problem. *Annals of Operations Research*, 61(1):111–120, 1995.

[83] D. C. Whybark. Issues in managing disaster relief inventories. *International Journal of Production Economics*, 108(1-2):228–235, 2007.

APPENDIX A

CALCULATED PARAMETERS IN PREPROCESSING

Table A.1: Sample size (SS) and sampling time (ST) requirements of random test instances with 30 nodes for each option (unit survey time=0.01)

		PL=0.05		PL=0.075		PL=0.1		PL=0.125		PL=0.150		PL=0.175		PL=0.2	
#	**Population Size**	**SS1**	**ST1**	**SS2**	**ST2**	**SS3**	**ST3**	**SS4**	**ST4**	**SS5**	**ST5**	**SS6**	**ST6**	**SS7**	**ST7**
1	11088	372	3.72	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
2	15961	376	3.76	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
3	10098	371	3.71	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
4	8275	368	3.68	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
5	18603	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
6	15901	376	3.76	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
7	10614	371	3.71	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
8	13806	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
9	17562	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
10	3440	346	3.46	163	1.63	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
11	1229	293	2.93	151	1.51	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
12	12195	373	3.73	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
13	240	148	1.48	101	1.01	69	0.69	50	0.50	37	0.37	28	0.28	22	0.22
14	9828	370	3.70	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
15	506	219	2.19	128	1.28	81	0.81	55	0.55	40	0.40	30	0.30	23	0.23
16	4185	352	3.52	165	1.65	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
17	3256	344	3.44	163	1.63	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
18	8289	368	3.68	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
19	15773	376	3.76	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
20	13194	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
21	9723	370	3.70	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
22	17589	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
23	8091	367	3.67	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
24	3210	344	3.44	163	1.63	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
25	11201	372	3.72	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
26	11004	372	3.72	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
27	5481	360	3.60	166	1.66	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
28	17177	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
29	1703	314	3.14	156	1.56	91	0.91	60	0.60	42	0.42	31	0.31	24	0.24
30	4867	357	3.57	165	1.65	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24

Table A.2: Sample size (SS) and sampling time (ST) requirements of random test instances with 50 nodes for each option (unit survey time=0.01)

		PL=0.05		PL=0.075		PL=0.1		PL=0.125		PL=0.150		PL=0.175		PL=0.2	
#	**Population Size**	**SS1**	**ST1**	**SS2**	**ST2**	**SS3**	**ST3**	**SS4**	**ST4**	**SS5**	**ST5**	**SS6**	**ST6**	**SS7**	**ST7**
1	7615	366.00	3.66	168.00	1.68	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
2	12798	373.00	3.73	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
3	19525	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
4	10916	372.00	3.72	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
5	18374	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
6	6352	363.00	3.63	167.00	1.67	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
7	3050	342.00	3.42	162.00	1.62	94.00	0.94	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
8	1351	300.00	3.00	152.00	1.52	90.00	0.90	59.00	0.59	42.00	0.42	31.00	0.31	24.00	0.24
9	13573	374.00	3.74	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
10	13212	374.00	3.74	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
11	10381	371.00	3.71	168.00	1.68	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
12	160	114.00	1.14	83.00	0.83	61.00	0.61	45.00	0.45	34.00	0.34	27.00	0.27	21.00	0.21
13	11790	373.00	3.73	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
14	8460	368.00	3.68	168.00	1.68	95.00	0.95	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
15	2315	330.00	3.30	160.00	1.60	93.00	0.93	60.00	0.60	42.00	0.42	31.00	0.31	24.00	0.24
16	19058	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
17	17339	376.00	3.76	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
18	12951	374.00	3.74	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
19	4147	352.00	3.52	165.00	1.65	94.00	0.94	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
20	10454	371.00	3.71	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
21	1190	291.00	2.91	150.00	1.50	89.00	0.89	59.00	0.59	42.00	0.42	31.00	0.31	24.00	0.24
22	1429	303.00	3.03	153.00	1.53	91.00	0.91	59.00	0.59	42.00	0.42	31.00	0.31	24.00	0.24
23	15495	375.00	3.75	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
24	15030	375.00	3.75	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
25	3165	343.00	3.43	163.00	1.63	94.00	0.94	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
26	7466	366.00	3.66	167.00	1.67	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
27	7771	367.00	3.67	168.00	1.68	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
28	13383	374.00	3.74	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
29	3268	344.00	3.44	163.00	1.63	94.00	0.94	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
30	18129	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
31	13663	374.00	3.74	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
32	6643	364.00	3.64	167.00	1.67	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
33	10274	371.00	3.71	168.00	1.68	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
34	7616	366.00	3.66	168.00	1.68	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
35	19692	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
36	9411	370.00	3.70	168.00	1.68	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
37	7359	366.00	3.66	167.00	1.67	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
38	5318	359.00	3.59	166.00	1.66	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
39	4048	351.00	3.51	164.00	1.64	94.00	0.94	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
40	12758	373.00	3.73	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
41	6309	363.00	3.63	167.00	1.67	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
42	3014	341.00	3.41	162.00	1.62	94.00	0.94	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
43	10928	372.00	3.72	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
44	18882	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
45	19508	377.00	3.77	170.00	1.70	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
46	14830	375.00	3.75	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
47	12791	373.00	3.73	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24
48	7837	367.00	3.67	168.00	1.68	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
49	6017	362.00	3.62	167.00	1.67	95.00	0.95	61.00	0.61	43.00	0.43	32.00	0.32	24.00	0.24
50	13195	374.00	3.74	169.00	1.69	96.00	0.96	62.00	0.62	43.00	0.43	32.00	0.32	24.00	0.24

Table A.3: Sample size (SS) and sampling time (ST) requirements of random test instances with 75 nodes for each option (unit survey time=0.01)

		PL=0.05		PL=0.075		PL=0.1		PL=0.125		PL=0.150		PL=0.175		PL=0.2	
#	**Population Size**	**SS1**	**ST1**	**SS2**	**ST2**	**SS3**	**ST3**	**SS4**	**ST4**	**SS5**	**ST5**	**SS6**	**ST6**	**SS7**	**ST7**
1	17119	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
2	656	243	2.43	136	1.36	84	0.84	57	0.57	41	0.41	30	0.30	24	0.24
3	6310	363	3.63	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
4	9689	370	3.70	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
5	11420	372	3.72	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
6	17014	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
7	7938	367	3.67	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
8	2261	329	3.29	159	1.59	93	0.93	60	0.60	42	0.42	31	0.31	24	0.24
9	3295	345	3.45	163	1.63	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
10	13144	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
11	8192	367	3.67	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
12	5057	358	3.58	166	1.66	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
13	16958	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
14	13463	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
15	5656	360	3.60	166	1.66	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
16	15296	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
17	14392	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
18	403	197	1.97	121	1.21	78	0.78	54	0.54	39	0.39	30	0.30	23	0.23
19	17137	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
20	7293	365	3.65	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
21	2945	340	3.40	162	1.62	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
22	1266	295	2.95	151	1.51	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
23	7945	367	3.67	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
24	8982	369	3.69	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
25	19842	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
26	15372	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
27	19737	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
28	3765	349	3.49	164	1.64	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
29	7449	366	3.66	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
30	8536	368	3.68	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
31	7115	365	3.65	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
32	12261	373	3.73	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
33	4112	352	3.52	164	1.64	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
34	10622	371	3.71	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
35	5158	358	3.58	166	1.66	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
36	15660	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
37	14380	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
38	4813	356	3.56	165	1.65	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
39	152	110	1.10	81	0.81	60	0.60	44	0.44	34	0.34	27	0.27	21	0.21
40	19103	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
41	6142	362	3.62	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
42	13485	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
43	11104	372	3.72	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
44	13066	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24

Table A.3 Sample size (SS) and sampling time (ST) requirements of random test instances with 75 nodes for each option (unit survey time=0.01) – continued

		PL=0.05		PL=0.075		PL=0.1		PL=0.125		PL=0.150		PL=0.175		PL=0.2	
#	**Population Size**	**SS1**	**ST1**	**SS2**	**ST2**	**SS3**	**ST3**	**SS4**	**ST4**	**SS5**	**ST5**	**SS6**	**ST6**	**SS7**	**ST7**
45	8043	367	3.67	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
46	19118	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
47	8098	367	3.67	168	1.68	95	0.95	62	0.62	43	0.43	32	0.32	24	0.24
48	15893	376	3.76	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
49	12284	373	3.73	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
50	6752	364	3.64	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
51	6299	363	3.63	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
52	13770	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
53	14658	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
54	19236	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
55	6709	364	3.64	167	1.67	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
56	1916	321	3.21	157	1.57	92	0.92	60	0.60	42	0.42	31	0.31	24	0.24
57	19167	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
58	5821	361	3.61	166	1.66	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
59	19611	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
60	13671	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
61	17597	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
62	10167	371	3.71	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
63	15321	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
64	12891	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
65	8962	369	3.69	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
66	2885	340	3.40	162	1.62	93	0.93	61	0.61	43	0.43	32	0.32	24	0.24
67	14713	375	3.75	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
68	13844	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
69	11828	373	3.73	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
70	18496	377	3.77	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
71	11021	372	3.72	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
72	12821	374	3.74	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
73	9590	370	3.70	168	1.68	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
74	17132	376	3.76	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
75	16375	376	3.76	169	1.69	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24

Table A.4: 2011 Van Earthquake Case Study Raw Data

					2011 Data	Projected based on 2000 data			
id	**Name**	**Latitude**	**Longitude**	**Elevation (meters)**	**Population**	**0-14 Total**	**65+ Total**	**Disabled Total**	**Female with children**
1	Bostaniçi	38.508214	43.435471	1777	21461	10981.51	446.73	608.09	1996.64
2	Erçek	38.653578	43.651936	1827	3769	1757.85	105.54	64.50	319.61
3	Adıgüzel	38.6971	43.43384	1781	838	459.10	19.76	11.62	83.47
4	Ağartı	38.705138	43.211496	1780	155	69.94	8.82	1.89	12.72
5	Ağzıkara	38.694721	43.761717	1998	1234	647.72	24.79	24.79	117.77
6	Akçaören	38.733197	43.495531	1910	502	282.44	13.24	13.24	51.35
7	Akçift	38.835429	43.58016	2372	156	73.60	3.15	1.89	13.38
8	Akın	38.412542	43.376205	2043	206	114.56	1.05	0.00	20.83
9	Aktaş	38.620961	43.65047	2017	779	662.21	25.34	25.34	120.40
10	Alabayır	38.571857	43.398385	1776	900	462.56	30.35	13.60	84.10
11	Alaköy	38.676196	43.243253	1736	886	292.61	55.25	15.35	53.20
12	Arısu	38.626929	43.229821	1677	401	200.98	11.43	6.67	36.54
13	Arıtoprak	38.552495	43.773218	2163	1119	616.49	27.20	29.79	112.09
14	Aşağıçıtlı	38.563939	43.504043	2016	341	155.00	9.00	3.00	28.18
15	Aşağıgölalan	38.752745	43.570719	1939	302	173.83	9.58	4.42	31.61
16	Aşıt	38.637255	43.390453	2082	365	203.08	7.16	0.89	36.92
17	Atmaca	38.69998	43.25969	1863	555	215.68	19.23	21.98	39.21
18	Bağdaşan	38.843585	43.526602	2022	301	167.22	7.09	3.04	30.40
19	Baklatepe	38.592523	43.583608	1942	418	214.49	11.97	5.99	39.00
20	Bakraçlı	38.449858	43.462422	2199	1243	670.48	23.95	20.68	121.91
21	Bardakçı	38.570649	43.264832	1660	5073	1386.32	47.61	43.80	252.06
22	Beşçatak	38.528696	43.557866	2113	527	261.43	9.34	6.22	47.53
23	Çakırbey	38.906096	43.585889	1763	482	230.17	14.44	11.73	41.85
24	Çalımlı	38.645366	43.770289	1933	1240	685.56	22.80	24.23	124.65

Table A.4 2011 Van Earthquake Case Study Raw Data – continued

					2011 Data	**Projected based on 2000 data**			
id	**Name**	**Latitude**	**Longitude**	**Elevation (meters)**	**Population**	**0-14 Total**	**65+ Total**	**Disabled Total**	**Female with children**
25	Çıtören	38.60074	43.218319	1699	760	414.44	11.48	10.33	75.35
26	Çobanoğlu	38.588665	43.427575	1836	312	172.85	8.70	2.17	31.43
27	Çolpan	38.909402	43.548074	1664	250	112.65	6.54	2.18	20.48
28	Çomaklı	38.618866	43.481866	2041	222	110.54	5.48	4.57	20.10
29	Dağönü	38.763186	43.247137	1703	376	186.93	9.31	7.88	33.99
30	Değirmenarkı	38.636182	43.728318	1904	430	226.45	5.09	15.27	41.17
31	Değirmenköy	38.517616	43.515716	2043	1465	720.96	38.02	20.37	131.08
32	Değirmenözü	38.764825	43.5374	1910	253	142.91	3.18	13.76	25.98
33	Derebey	38.795972	43.494329	1948	587	339.18	16.80	8.40	61.67
34	Dereüstü	38.560918	43.468251	1917	340	174.76	4.76	7.13	31.77
35	Dibekdüzü	38.641361	43.239266	1687	278	121.63	10.43	9.27	22.11
36	Dibekli	38.687687	43.840206	2189	482	262.48	16.81	21.48	47.72
37	Dilimli	38.674186	43.462422	1864	872	474.32	29.07	21.14	86.24
38	Ermişler	38.86621	43.508942	1802	1778	963.74	49.83	52.09	175.23
39	Esenpınar	38.803731	43.334513	1862	347	169.72	9.73	0.00	30.86
40	Gedelova	38.606005	43.487649	1938	371	165.82	14.32	1.19	30.15
41	Gedikbulak	38.835496	43.43869	1745	1201	542.73	33.67	11.22	98.68
42	Göllü	38.72168	43.313928	1734	328	112.69	14.66	10.08	20.49
43	Gölyazı	38.588095	43.46293	1895	386	204.21	6.23	3.74	37.13
44	Gövelek	38.53588	43.620393	2268	1142	636.58	32.10	25.68	115.74
45	Gülsünler	38.684773	43.346114	1700	244	102.64	7.20	3.60	18.66
46	Güvençli	38.740561	43.315802	1745	1408	762.99	42.32	38.47	138.72
47	Halkalı	38.833423	43.316832	1728	737	377.41	24.49	14.47	68.62
48	Hıdırköy	38.720826	43.40273	1816	618	300.85	11.95	7.60	54.70

Table A.4 2011 Van Earthquake Case Study Raw Data – continued

					2011 Data	Projected based on 2000 data			
id	**Name**	**Latitude**	**Longitude**	**Elevation (meters)**	**Population**	**0-14 Total**	**65+ Total**	**Disabled Total**	**Female with children**
49	Ilıkaynak	38.75077	43.616188	1936	1193	725.41	26.22	21.85	131.89
50	Irgat	38.606341	43.615451	1929	497	259.05	9.38	8.21	47.10
51	Kalecik	38.545312	43.338654	1725	3101	1507.78	44.10	41.35	274.14
52	Karaağaç	38.846861	43.494086	1814	368	211.13	7.08	5.90	38.39
53	Karagündüz	38.69586	43.645713	1812	1009	417.18	30.72	32.34	75.85
54	Karakoç	38.587357	43.770504	2089	1218	677.37	31.80	27.56	123.16
55	Karpuzalanı	38.511102	43.455555	1810	3049	1631.19	78.24	82.98	296.58
56	Kasımoğlu	38.687855	43.416485	1725	517	186.88	18.11	7.41	33.98
57	Kavuncu	38.513755	43.470275	1831	1563	762.82	28.73	31.60	138.70
58	Kavurma	38.462595	43.361957	1721	162	78.00	7.50	1.50	14.18
59	Kaymaklı	38.639467	43.700423	1982	717	356.57	18.36	12.56	64.83
60	Kelle	38.866845	43.647902	2184	112	63.78	3.11	1.56	11.60
61	Kevenli	38.476102	43.449025	2009	2611	1204.78	77.48	42.61	219.05
62	Kıratlı	38.534873	43.462586	1893	992	489.89	29.32	12.22	89.07
63	Koçköy	38.70708	43.506625	1869	300	167.56	7.32	5.12	30.47
64	Kolsatan	38.648115	43.440492	1977	182	75.04	4.79	11.18	13.64
65	Kozluca	38.654382	43.515937	1951	170	95.55	2.48	2.90	17.37
66	Köşebaşı	38.545413	43.537266	2137	744	423.03	19.68	11.07	76.92
67	Kumluca	38.842649	43.397956	1662	553	274.75	10.50	5.25	49.95
68	Kurubaş	38.455034	43.409936	1812	2029	1064.32	43.03	44.16	193.51
69	Meydancık	38.769744	43.504114	2105	144	88.18	0.81	1.62	16.03
70	Mollakasım	38.678776	43.190081	1787	147	49.36	2.17	2.71	8.97
71	Ocaklı	38.73467	43.415537	1905	581	320.98	12.44	8.71	58.36
72	Ortaca	38.553032	43.700087	2157	2112	1099.08	50.81	50.81	199.83

Table A.4 2011 Van Earthquake Case Study Raw Data – continued

					2011 Data	Projected based on 2000 data			
id	**Name**	**Latitude**	**Longitude**	**Elevation (meters)**	**Population**	**0-14 Total**	**65+ Total**	**Disabled Total**	**Female with children**
73	Otluca	38.693884	43.363366	1724	193	60.55	7.57	1.89	11.01
74	Özkaynak	38.672444	43.326459	1683	397	197.03	10.29	7.35	35.82
75	Özyurt	38.737214	43.205101	1676	139	56.80	5.98	3.74	10.33
76	Pirgarip	38.764524	43.406096	1925	754	397.41	22.56	13.96	72.26
77	Sağlamtaş	38.897212	43.601854	2034	241	136.12	4.46	10.04	24.75
78	Sarmaç	38.506065	43.493571	2004	371	175.80	5.17	14.22	31.96
79	Satıbey	38.691975	43.481605	2004	146	69.87	9.85	0.90	12.70
80	Şahgeldi	38.773023	43.344998	1849	394	204.92	8.45	12.68	37.26
81	Tabanlı	38.751239	43.363502	1779	203	107.30	2.90	2.90	19.51
82	Taşkonak	38.383517	43.472714	2024	375	200.17	10.14	5.07	36.39
83	Tevekli	38.632964	43.254704	1666	223	95.10	3.28	3.28	17.29
84	Topaktaş	38.609024	43.232989	1710	571	256.72	20.72	6.91	46.68
85	Yalınağaç	38.571555	43.563917	2029	414	211.18	5.97	3.58	38.40
86	Yatıksırt	38.695726	43.80214	2033	1204	660.41	29.80	15.50	120.07
87	Yaylıkaya	38.818913	43.36637	1693	650	323.58	19.87	4.73	58.83
88	Yemlice	38.74257	43.452272	2077	476	264.02	7.71	15.42	48.00
89	Yeniköşk	38.654416	43.344483	1783	413	229.07	7.90	16.93	41.65
90	Yeşilsu	38.784599	43.291426	1660	672	302.78	24.68	5.69	55.05
91	Yukarıgölalan	38.769343	43.577929	2212	808	445.79	13.37	18.95	81.05
92	Yukarıgüneyce	38.730786	43.724699	2133	199	108.21	5.50	6.42	19.67
93	Yumrutepe	38.682461	43.308098	1694	432	163.96	15.71	12.76	29.81

Table A.5: Sample size (SS) and sampling time (ST) requirements of case study instances for each option (unit survey time=0.01)

		PL=0.05		PL=0.075		PL=0.1		PL=0.125		PL=0.150		PL=0.175		PL=0.2	
#	**Population Size**	**SS1**	**ST1**	**SS2**	**ST2**	**SS3**	**ST3**	**SS4**	**ST4**	**SS5**	**ST5**	**SS6**	**ST6**	**SS7**	**ST7**
1	21461	378	3.78	170	1.70	96	0.96	62	0.62	43	0.43	32	0.32	24	0.24
2	3769	349	3.49	164	1.64	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
3	838	264	2.64	142	1.42	87	0.87	58	0.58	41	0.41	31	0.31	24	0.24
4	155	111	1.11	82	0.82	60	0.60	45	0.45	34	0.34	27	0.27	21	0.21
5	1234	294	2.94	151	1.51	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
6	502	218	2.18	128	1.28	81	0.81	55	0.55	40	0.40	30	0.30	23	0.23
7	156	112	1.12	82	0.82	60	0.60	45	0.45	34	0.34	27	0.27	21	0.21
8	206	135	1.35	94	0.94	66	0.66	48	0.48	36	0.36	28	0.28	22	0.22
9	779	258	2.58	141	1.41	86	0.86	58	0.58	41	0.41	31	0.31	24	0.24
10	900	270	2.70	144	1.44	87	0.87	58	0.58	41	0.41	31	0.31	24	0.24
11	886	269	2.69	144	1.44	87	0.87	58	0.58	41	0.41	31	0.31	24	0.24
12	401	197	1.97	120	1.20	78	0.78	54	0.54	39	0.39	30	0.30	23	0.23
13	1119	287	2.87	149	1.49	89	0.89	59	0.59	42	0.42	31	0.31	24	0.24
14	341	181	1.81	114	1.14	76	0.76	53	0.53	39	0.39	29	0.29	23	0.23
15	302	170	1.70	110	1.10	74	0.74	52	0.52	38	0.38	29	0.29	23	0.23
16	365	188	1.88	117	1.17	77	0.77	53	0.53	39	0.39	29	0.29	23	0.23
17	555	228	2.28	131	1.31	82	0.82	56	0.56	40	0.40	30	0.30	24	0.24
18	301	170	1.70	110	1.10	73	0.73	52	0.52	38	0.38	29	0.29	23	0.23
19	418	201	2.01	122	1.22	79	0.79	54	0.54	39	0.39	30	0.30	23	0.23
20	1243	294	2.94	151	1.51	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
21	5073	358	3.58	166	1.66	95	0.95	61	0.61	43	0.43	32	0.32	24	0.24
22	527	223	2.23	130	1.30	82	0.82	56	0.56	40	0.40	30	0.30	24	0.24
23	482	215	2.15	127	1.27	81	0.81	55	0.55	40	0.40	30	0.30	23	0.23
24	1240	294	2.94	151	1.51	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
25	760	256	2.56	140	1.40	86	0.86	57	0.57	41	0.41	31	0.31	24	0.24
26	312	173	1.73	111	1.11	74	0.74	52	0.52	38	0.38	29	0.29	23	0.23
27	250	152	1.52	102	1.02	70	0.70	50	0.50	37	0.37	28	0.28	22	0.22
28	222	141	1.41	97	0.97	68	0.68	49	0.49	36	0.36	28	0.28	22	0.22
29	376	191	1.91	118	1.18	77	0.77	53	0.53	39	0.39	30	0.30	23	0.23
30	430	204	2.04	123	1.23	79	0.79	54	0.54	39	0.39	30	0.30	23	0.23
31	1465	305	3.05	154	1.54	91	0.91	60	0.60	42	0.42	31	0.31	24	0.24
32	253	153	1.53	103	1.03	70	0.70	50	0.50	37	0.37	28	0.28	23	0.23
33	587	233	2.33	133	1.33	83	0.83	56	0.56	40	0.40	30	0.30	24	0.24
34	340	181	1.81	114	1.14	76	0.76	53	0.53	39	0.39	29	0.29	23	0.23
35	278	162	1.62	107	1.07	72	0.72	51	0.51	38	0.38	29	0.29	23	0.23
36	482	215	2.15	127	1.27	81	0.81	55	0.55	40	0.40	30	0.30	23	0.23
37	872	267	2.67	143	1.43	87	0.87	58	0.58	41	0.41	31	0.31	24	0.24
38	1778	317	3.17	156	1.56	92	0.92	60	0.60	42	0.42	31	0.31	24	0.24
39	347	183	1.83	115	1.15	76	0.76	53	0.53	39	0.39	29	0.29	23	0.23
40	371	189	1.89	118	1.18	77	0.77	53	0.53	39	0.39	29	0.29	23	0.23
41	1201	292	2.92	150	1.50	89	0.89	59	0.59	42	0.42	31	0.31	24	0.24
42	328	178	1.78	113	1.13	75	0.75	52	0.52	38	0.38	29	0.29	23	0.23
43	386	193	1.93	119	1.19	78	0.78	54	0.54	39	0.39	30	0.30	23	0.23
44	1142	288	2.88	149	1.49	89	0.89	59	0.59	42	0.42	31	0.31	24	0.24
45	244	150	1.50	101	1.01	70	0.70	50	0.50	37	0.37	28	0.28	22	0.22
46	1408	302	3.02	153	1.53	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
47	737	253	2.53	139	1.39	86	0.86	57	0.57	41	0.41	31	0.31	24	0.24
48	618	238	2.38	134	1.34	84	0.84	56	0.56	40	0.40	30	0.30	24	0.24

Table A.5 Sample size (SS) and sampling time (ST) requirements of case study instances for each option (unit survey time=0.01) – continued

		PL=0.05		PL=0.075		PL=0.1		PL=0.125		PL=0.150		PL=0.175		PL=0.2	
#	**Population Size**	**SS1**	**ST1**	**SS2**	**ST2**	**SS3**	**ST3**	**SS4**	**ST4**	**SS5**	**ST5**	**SS6**	**ST6**	**SS7**	**ST7**
49	1193	291	2.91	150	1.50	89	0.89	59	0.59	42	0.42	31	0.31	24	0.24
50	497	217	2.17	128	1.28	81	0.81	55	0.55	40	0.40	30	0.30	23	0.23
51	3101	342	3.42	162	1.62	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
52	368	189	1.89	117	1.17	77	0.77	53	0.53	39	0.39	29	0.29	23	0.23
53	1009	279	2.79	147	1.47	88	0.88	58	0.58	41	0.41	31	0.31	24	0.24
54	1218	293	2.93	150	1.50	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
55	3049	342	3.42	162	1.62	94	0.94	61	0.61	43	0.43	32	0.32	24	0.24
56	517	221	2.21	129	1.29	82	0.82	56	0.56	40	0.40	30	0.30	23	0.23
57	1563	309	3.09	155	1.55	91	0.91	60	0.60	42	0.42	31	0.31	24	0.24
58	162	115	1.15	84	0.84	61	0.61	45	0.45	34	0.34	27	0.27	22	0.22
59	717	251	2.51	139	1.39	85	0.85	57	0.57	41	0.41	31	0.31	24	0.24
60	112	87	0.87	68	0.68	52	0.52	40	0.40	32	0.32	25	0.25	20	0.20
61	2611	335	3.35	161	1.61	93	0.93	61	0.61	43	0.43	31	0.31	24	0.24
62	992	278	2.78	146	1.46	88	0.88	58	0.58	41	0.41	31	0.31	24	0.24
63	300	169	1.69	110	1.10	73	0.73	52	0.52	38	0.38	29	0.29	23	0.23
64	182	124	1.24	89	0.89	64	0.64	47	0.47	35	0.35	27	0.27	22	0.22
65	170	119	1.19	86	0.86	62	0.62	46	0.46	35	0.35	27	0.27	22	0.22
66	744	254	2.54	140	1.40	86	0.86	57	0.57	41	0.41	31	0.31	24	0.24
67	553	227	2.27	131	1.31	82	0.82	56	0.56	40	0.40	30	0.30	24	0.24
68	2029	324	3.24	158	1.58	92	0.92	60	0.60	42	0.42	31	0.31	24	0.24
69	144	105	1.05	79	0.79	58	0.58	44	0.44	34	0.34	26	0.26	21	0.21
70	147	107	1.07	80	0.80	59	0.59	44	0.44	34	0.34	26	0.26	21	0.21
71	581	232	2.32	133	1.33	83	0.83	56	0.56	40	0.40	30	0.30	24	0.24
72	2112	326	3.26	159	1.59	92	0.92	60	0.60	42	0.42	31	0.31	24	0.24
73	193	129	1.29	91	0.91	65	0.65	47	0.47	36	0.36	28	0.28	22	0.22
74	397	196	1.96	120	1.20	78	0.78	54	0.54	39	0.39	30	0.30	23	0.23
75	139	103	1.03	77	0.77	58	0.58	43	0.43	33	0.33	26	0.26	21	0.21
76	754	255	2.55	140	1.40	86	0.86	57	0.57	41	0.41	31	0.31	24	0.24
77	241	149	1.49	101	1.01	69	0.69	50	0.50	37	0.37	28	0.28	22	0.22
78	371	189	1.89	118	1.18	77	0.77	53	0.53	39	0.39	29	0.29	23	0.23
79	146	106	1.06	79	0.79	59	0.59	44	0.44	34	0.34	26	0.26	21	0.21
80	394	195	1.95	120	1.20	78	0.78	54	0.54	39	0.39	30	0.30	23	0.23
81	203	134	1.34	93	0.93	66	0.66	48	0.48	36	0.36	28	0.28	22	0.22
82	375	191	1.91	118	1.18	77	0.77	53	0.53	39	0.39	30	0.30	23	0.23
83	223	142	1.42	97	0.97	68	0.68	49	0.49	36	0.36	28	0.28	22	0.22
84	571	230	2.30	132	1.32	83	0.83	56	0.56	40	0.40	30	0.30	24	0.24
85	414	200	2.00	122	1.22	79	0.79	54	0.54	39	0.39	30	0.30	23	0.23
86	1204	292	2.92	150	1.50	90	0.90	59	0.59	42	0.42	31	0.31	24	0.24
87	650	242	2.42	136	1.36	84	0.84	57	0.57	41	0.41	30	0.30	24	0.24
88	476	213	2.13	126	1.26	81	0.81	55	0.55	40	0.40	30	0.30	23	0.23
89	413	200	2.00	122	1.22	79	0.79	54	0.54	39	0.39	30	0.30	23	0.23
90	672	245	2.45	137	1.37	85	0.85	57	0.57	41	0.41	31	0.31	24	0.24
91	808	261	2.61	142	1.42	86	0.86	58	0.58	41	0.41	31	0.31	24	0.24
92	199	132	1.32	93	0.93	65	0.65	48	0.48	36	0.36	28	0.28	22	0.22
93	432	204	2.04	123	1.23	79	0.79	54	0.54	39	0.39	30	0.30	23	0.23

APPENDIX B

SOLUTIONS OF RANDOM TEST INSTANCES

Table B.1: Route and option assignment in CPLEX solutions of random test instances

Instance			
#	Name	Route	option
1	30N_10C_tmax3_geo	0-28-12-26-21-0	6-6-6-6
2	30N_10C_tmax3_hier	0-28-12-26-21-0	6-6-6-6
3	30N_10C_tmax3_rand	0-6-13-28-27-0	6-4-6-6
4	30N_10C_tmax5_geo	0-28-12-4-21-13-5-18-0	7-7-7-7-7-7-7
5	30N_10C_tmax5_hier	0-27-10-11-19-7-18-6-0	6-6-7-6-6-6-7
6	30N_10C_tmax5_rand	0-6-13-2-22-21-26-12-28-0	6-7-7-7-6-6-6-7
7	30N_10C_tmax7_geo	0-21-4-12-1-30-10-18-5-6-0	7-7-7-7-6-6-6-7-7
8	30N_10C_tmax7_hier	0-13-14-16-17-5-18-7-19-11-27-0	7-7-7-7-7-7-7-7-7-7
9	30N_10C_tmax7_rand	0-7-18-6-13-2-22-21-26-12-28-0	6-6-6-6-6-6-6-6-6-6
10	30N_15C_tmax3_geo	0-28-26-21-13-0	6-6-6-6
11	30N_15C_tmax3_hier	0-13-6-18-27-0	5-6-6-6
12	30N_15C_tmax3_rand	0-13-6-18-27-0	5-6-6-6
13	30N_15C_tmax5_geo	0-6-18-7-11-10-1-27-0	7-6-7-7-6-6-6
14	30N_15C_tmax5_hier	0-27-1-30-20-9-3-12-26-0	7-7-6-7-7-7-7-7
15	30N_15C_tmax5_rand	0-6-13-2-21-4-26-12-28-0	7-7-7-6-6-6-7-7
16	30N_15C_tmax7_geo	0-26-4-22-2-13-18-7-10-1-27-0	7-7-7-7-6-7-7-6-7-7
17	30N_15C_tmax7_hier	0-6-13-2-22-21-4-25-24-29-3-28-0	7-7-7-7-6-6-7-6-7-7-7
18	30N_15C_tmax7_rand	0-27-1-30-10-11-19-7-18-6-13-26-0	6-7-6-6-7-7-7-6-7-7-7
19	50N_10C_tmax3_geo	0-27-31-18-6-0	6-7-7-6
20	50N_10C_tmax3_hier	0-27-50-12-26-0	6-6-6-5
21	50N_10C_tmax3_rand	0-12-50-1-27-0	6-6-6-5
22	50N_10C_tmax5_geo	0-13-42-37-18-31-1-27-0	7-6-7-7-7-7-7
23	50N_10C_tmax5_hier	0-26-12-3-33-9-20-1-27-0	6-6-7-7-7-7-7-7
24	50N_10C_tmax5_rand	0-18-7-31-1-50-12-26-40-0	7-7-7-7-7-6-7-7
25	50N_10C_tmax7_geo	0-26-4-21-42-37-8-48-31-1-0	7-7-7-7-7-7-7-7-7
26	50N_10C_tmax7_hier	0-6-37-16-8-46-47-19-10-31-27-0	6-6-6-7-7-6-6-7-7-6
27	50N_10C_tmax7_rand	0-40-21-41-2-13-18-48-7-31-27-0	6-7-6-6-7-6-6-6-7-6
28	50N_10C_tmax10_geo	0-13-42-37-48-7-31-33-34-29-12-0	6-5-5-5-5-5-4-5-5-3
29	50N_10C_tmax10_hier	0-26-12-3-29-34-9-20-33-50-28-0	4-3-4-4-4-4-4-4-4-4
30	50N_10C_tmax10_rand	0-40-21-26-12-50-1-31-7-45-18-0	4-4-4-3-4-6-4-4-4-4
31	50N_15C_tmax3_geo	0-27-31-18-6-0	6-7-7-6
32	50N_15C_tmax3_hier	0-28-12-21-40-0	6-6-5-6
33	50N_15C_tmax3_rand	0-28-12-21-40-0	6-6-5-6
34	50N_15C_tmax5_geo	0-6-8-48-7-31-1-27-0	6-7-6-6-6-6-6
35	50N_15C_tmax5_hier	0-7-48-47-19-11-10-31-27-0	6-7-7-6-7-7-7-7

Table B.1 Route and option assignment in CPLEX solutions of random test instances – continued

Instance			
#	Name	Route	option
36	50N_15C_tmax5_rand	0-26-40-21-2-42-37-6-13-0	7-7-7-7-6-7-6-7
37	50N_15C_tmax7_geo	0-6-18-8-47-19-10-30-1-12-28-0	6-6-7-7-6-6-6-6-7
38	50N_15C_tmax7_hier	0-6-5-8-46-47-19-11-10-31-1-50-28-0	7-7-7-7-7-7-7-7-7-7-7-7
39	50N_15C_tmax7_rand	0-27-1-31-7-48-8-45-17-37-13-40-0	6-6-7-6-6-7-7-7-6-7-7
40	50N_15C_tmax10_geo	0-28-3-33-9-30-10-19-47-8-5-16-37-42-2-0	6-6-6-6-6-7-6-6-7-7-6-6-6-6
41	50N_15C_tmax10_hier	0-6-5-8-46-36-47-19-11-10-31-1-9-33-50-27-0	6-7-7-6-6-6-6-6-6-6-6-6-6-6-6
42	50N_15C_tmax10_rand	0-40-21-22-15-42-37-5-45-8-31-1-50-33-12-28-0	6-6-7-7-6-6-6-7-7-7-7-6-6-6
43	50N_20C_tmax5_geo	0-13-6-8-48-7-31-27-0	6-6-7-6-6-6-6
44	50N_20C_tmax5_hier	0-13-2-41-22-21-26-12-28-27-0	7-7-7-7-7-7-7-7-7
45	50N_20C_tmax5_rand	0-40-21-12-3-33-50-1-27-0	7-6-5-7-6-7-6-7
46	50N_20C_tmax7_geo	0-28-12-50-30-10-19-48-8-18-6-13-0	7-6-7-7-7-6-7-7-7-6-7
47	50N_20C_tmax7_hier	0-27-28-12-26-21-4-39-23-22-41-2-13-0	7-7-5-7-6-7-6-7-6-7-7-7
48	50N_20C_tmax7_rand	0-18-7-10-30-20-9-33-3-50-1-27-0	7-6-6-6-7-7-6-7-6-6-6
49	50N_20C_tmax10_geo	0-13-42-37-16-5-18-8-48-19-10-30-9-33-12-28-0	7-6-6-7-7-6-7-6-6-7-7-7-6-6-7
50	50N_20C_tmax10_hier	0-27-28-50-1-31-10-19-47-48-7-18-5-17-16-37-13-6-0	7-7-7-7-7-7-6-7-6-6-7-7-7-7-6-7-7
51	50N_20C_tmax10_rand	0-27-1-30-10-31-7-19-36-47-48-8-45-17-16-37-13-40-0	7-7-7-7-7-6-6-7-7-7-6-7-7-7-7-7-7
52	50N_25C_tmax5_geo	0-27-31-7-18-6-13-40-26-0	7-7-6-7-6-7-7-6
53	50N_25C_tmax5_hier	0-40-21-26-12-3-33-50-28-27-0	7-7-7-7-7-7-7-7-7
54	50N_25C_tmax5_rand	0-40-21-26-12-3-33-50-1-27-0	7-7-7-6-7-7-7-7-7
55	50N_25C_tmax7_geo	0-26-40-13-6-18-7-48-47-19-31-1-27-0	7-7-7-7-7-7-7-7-7-7-7-7
56	50N_25C_tmax7_hier	0-13-37-42-43-15-41-22-21-26-12-28-27-0	7-6-6-7-7-6-7-7-6-6-7-7
57	50N_25C_tmax7_rand	0-27-31-1-50-33-3-12-26-21-40-13-6-0	6-7-6-7-7-7-6-6-7-7-7-6
58	50N_25C_tmax10_geo	0-40-13-6-18-7-48-47-36-19-10-32-30-33-50-12-28-0	7-7-6-7-6-6-7-6-6-7-6-7-7-7-6-7
59	50N_25C_tmax10_hier	0-28-12-26-40-2-42-37-16-17-45-8-46-47-19-11-10-31-27-0	7-6-7-7-7-6-7-7-7-7-7-7-7-7-7-7-7-7
60	50N_25C_tmax10_rand	0-26-40-21-2-42-14-44-16-5-17-45-8-48-19-11-10-31-27-0	7-7-6-7-7-7-7-7-7-7-7-6-7-7-7-7-7-7
61	75N_15C_tmax5_geo	0-53-26-21-2-42-59-18-52-0	7-7-7-7-7-7-7-7
62	75N_15C_tmax5_hier	0-53-21-72-74-22-41-2-13-6-0	7-6-6-7-7-6-7-6-6
63	75N_15C_tmax5_rand	0-27-69-1-50-12-21-73-40-58-53-0	7-7-7-7-7-7-7-7-7-7
64	75N_15C_tmax7_geo	0-18-48-19-62-10-1-26-21-40-53-0	6-6-6-6-7-6-6-6-6-6
65	75N_15C_tmax7_hier	0-53-21-72-74-56-39-25-55-54-12-1-69-0	7-6-7-7-6-5-6-6-7-6-6-6
66	75N_15C_tmax7_rand	0-27-6-61-16-44-14-2-73-21-40-58-53-0	6-6-7-6-6-6-7-6-6-7-6-7
67	75N_15C_tmax10_geo	0-59-18-48-19-62-30-51-9-29-55-4-21-40-53-0	6-7-6-6-6-6-6-6-6-6-6-6-7-7
68	75N_15C_tmax10_hier	0-26-54-55-25-67-39-56-74-73-58-5-45-18-52-0	6-6-6-6-6-6-5-6-6-6-6-6-4-6
69	75N_15C_tmax10_rand	0-53-58-40-21-73-41-2-13-6-18-48-62-31-69-27-0	6-5-6-5-6-5-5-6-6-4-6-5-5-6-6
70	75N_25C_tmax5_geo	0-53-13-59-61-5-18-69-27-0	6-7-7-6-6-7-6-6
71	75N_25C_tmax5_hier	0-13-2-41-22-74-72-73-21-40-58-53-0	7-7-7-7-7-7-7-7-7-7-7
72	75N_25C_tmax5_rand	0-28-26-21-72-74-22-41-2-58-53-0	6-7-6-7-7-7-6-7-7-7
73	75N_25C_tmax7_geo	0-53-13-59-60-18-8-47-19-11-10-31-27-0	7-6-7-7-7-7-6-7-7-7-6-7
74	75N_25C_tmax7_hier	0-53-40-21-73-72-74-22-75-56-39-25-55-54-12-28-0	7-7-6-7-7-7-7-7-6-7-7-7-6-6
75	75N_25C_tmax7_rand	0-53-58-40-21-72-22-75-56-4-12-50-1-69-27-0	7-6-7-6-7-7-7-7-7-6-6-7-7-7
76	75N_25C_tmax10_geo	0-53-13-59-61-5-18-8-47-19-11-10-70-51-9-33-68-12-28-0	7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7
77	75N_25C_tmax10_hier	0-69-1-70-30-20-71-35-34-29-24-4-56-75-74-72-73-21-40-58-53-0	7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-7-6-7-6-7
78	75N_25C_tmax10_rand	0-53-6-59-37-61-16-44-14-42-57-22-74-72-21-26-12-68-3-50-28-0	7-7-7-7-7-7-7-7-7-7-7-7-7-6-7-7-7-7-7-6

Table B.2: Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for random test instances

Instance		**CPLEX**		t_3		t_4		t_5		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
1	30N_10C_tmax3_geo	0.396	1.9	0.396	5.2	0.396	5.0	0.396	5.2	0.396	4.8	0.396	4.9	0.396	5.1	0.396	4.9	0.396	5.0	0.396	5.0
2	30N_10C_tmax3_hier	0.396	8.3	0.396	5.3	0.396	4.9	0.396	5.8	0.396	5.4	0.396	5.4	0.396	5.2	0.396	5.3	0.396	5.2	0.396	5.2
3	30N_10C_tmax3_rand	0.396	12.5	0.396	6.5	0.396	7.0	0.396	6.7	0.396	6.3	0.396	6.2	0.396	6.4	0.396	6.3	0.396	6.2	0.396	6.2
4	30N_10C_tmax5_geo	0.697	2.3	0.597	6.5	0.597	6.6	0.597	6.6	0.597	6.8	0.597	6.5	0.597	6.3	0.597	6.5	0.597	6.7	0.597	6.7
5	30N_10C_tmax5_hier	0.697	446.2	0.697	6.8	0.697	6.5	0.697	6.9	0.697	6.7	0.697	6.7	0.697	6.6	0.697	6.6	0.697	6.7	0.697	6.7
6	30N_10C_tmax5_rand	0.798	2.0	0.798	7.6	0.798	7.5	0.798	7.7	0.798	7.8	0.798	7.5	0.798	7.6	0.798	7.8	0.798	7.7	0.798	7.7
7	30N_10C_tmax7_geo	0.898	78.5	0.898	7.0	0.898	7.3	0.898	7.3	0.898	7.3	0.898	7.0	0.898	7.3	0.898	7.3	0.898	7.2	0.898	7.2
8	30N_10C_tmax7_hier	0.998	78.4	0.998	8.5	0.998	8.0	0.998	8.5	0.998	8.1	0.998	9.2	0.998	7.7	0.998	7.7	0.998	8.4	0.998	8.4
9	30N_10C_tmax7_rand	0.998	3.8	0.998	7.8	0.998	8.1	0.998	8.7	0.998	8.3	0.998	7.9	0.998	8.3	0.998	8.0	0.998	7.6	0.998	7.6
10	30N_15C_tmax3_geo	0.263	5.9	0.263	6.9	0.263	5.6	0.263	6.1	0.263	5.6	0.263	5.8	0.263	6.4	0.263	5.8	0.263	5.8	0.263	5.8
11	30N_15C_tmax3_hier	0.263	9.8	0.263	6.6	0.263	6.9	0.263	7.2	0.263	6.8	0.263	6.6	0.263	6.7	0.263	6.9	0.263	6.7	0.263	6.7
12	30N_15C_tmax3_rand	0.263	11.2	0.263	6.9	0.263	6.8	0.263	6.7	0.263	6.8	0.263	6.6	0.263	6.9	0.263	6.8	0.263	6.7	0.263	6.7
13	30N_15C_tmax5_geo	0.463	23.0	0.463	9.0	0.463	8.8	0.463	9.2	0.463	9.5	0.463	9.0	0.463	8.7	0.463	9.0	0.463	9.0	0.463	9.0
14	30N_15C_tmax5_hier	0.530	8.1	0.530	9.1	0.530	9.2	0.530	9.0	0.530	9.3	0.530	9.4	0.530	8.9	0.530	9.3	0.530	9.0	0.530	9.0
15	30N_15C_tmax5_rand	0.530	6.0	0.530	9.2	0.530	9.7	0.530	10.5	0.530	9.7	0.530	9.6	0.530	9.9	0.530	9.5	0.530	9.5	0.530	9.5
16	30N_15C_tmax7_geo	0.664	64.0	0.664	11.9	0.664	11.6	0.664	11.9	0.664	11.8	0.664	11.7	0.664	11.5	0.664	11.6	0.664	11.4	0.664	11.4
17	30N_15C_tmax7_hier	0.731	128.0	0.731	11.3	0.731	11.4	0.731	12.3	0.731	13.7	0.731	11.3	0.731	12.6	0.731	12.8	0.731	12.1	0.731	12.1
18	30N_15C_tmax7_rand	0.731	183.2	0.731	11.2	0.731	11.8	0.731	11.7	0.731	11.6	0.731	11.6	0.731	11.1	0.731	11.2	0.731	10.9	0.731	10.9
19	50N_10C_tmax3_geo	0.396	19.3	0.396	8.0	0.396	8.4	0.396	10.0	0.396	9.3	0.396	9.8	0.396	8.6	0.396	8.1	0.396	8.4	0.396	8.4

Table B.2 Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for random test instances – continued

Instance		**CPLEX**		t_3		t_4		t_5		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
20	50N_10C_tmax3_hier	0.396	232.9	0.396	10.5	0.396	9.9	0.396	10.9	0.396	10.1	0.396	10.0	0.396	10.4	0.396	10.1	0.396	10.3	0.396	10.3
21	50N_10C_tmax3_rand	0.396	265.1	0.396	10.5	0.396	9.7	0.396	9.5	0.396	9.9	0.396	9.8	0.396	9.5	0.396	9.7	0.396	9.6	0.396	9.6
22	50N_10C_tmax5_geo	0.697	39.5	0.697	12.3	0.697	12.7	0.697	11.5	0.597	12.1	0.697	12.8	0.697	12.9	0.597	11.1	0.697	12.6	0.697	12.6
23	50N_10C_tmax5_hier	0.797	2631.9	0.797	11.8	0.797	11.7	0.797	12.2	0.797	12.1	0.797	10.7	0.797	10.9	0.797	10.9	0.797	10.9	0.797	10.9
24	50N_10C_tmax5_rand	0.797	410.5	0.797	11.7	0.797	11.6	0.797	11.8	0.797	11.7	0.797	11.3	0.797	11.5	0.797	11.5	0.797	12.0	0.797	12.0
25	50N_10C_tmax7_geo	0.898	2645.1	0.898	16.7	0.898	16.3	0.898	16.4	0.898	17.0	0.898	19.0	0.898	16.8	0.898	18.1	0.898	18.8	0.898	18.8
26	50N_10C_tmax7_hier	0.998	7202.6	0.998	11.0	0.998	10.9	0.998	12.8	0.998	11.7	0.998	11.0	0.998	11.2	0.998	12.5	0.998	11.1	0.998	11.1
27	50N_10C_tmax7_rand	0.998	7203.7	0.998	13.6	0.998	11.8	0.998	12.0	0.998	12.4	0.998	11.5	0.998	11.8	0.998	11.3	0.998	11.6	0.998	11.6
28	50N_10C_tmax10_geo	0.999	7206.8	0.999	21.3	0.999	22.3	0.999	21.6	0.998	21.6	0.998	22.6	0.999	22.9	0.999	23.3	0.999	20.8	0.999	20.8
29	50N_10C_tmax10_hier	0.999	2714.8	0.999	14.0	0.999	13.4	0.999	14.3	0.999	13.2	0.999	13.3	0.999	13.5	0.999	12.9	0.999	12.9	0.999	12.9
30	50N_10C_tmax10_rand	0.999	7201.7	0.999	15.3	0.999	13.5	0.999	15.1	0.999	15.3	0.999	13.8	0.999	14.1	0.999	13.7	0.999	13.7	0.999	13.7
31	50N_15C_tmax3_geo	0.263	52.5	0.263	9.0	0.263	8.7	0.263	8.8	0.263	8.9	0.263	8.6	0.263	8.8	0.263	8.7	0.263	8.5	0.263	8.5
32	50N_15C_tmax3_hier	0.263	459.0	0.263	10.2	0.263	10.8	0.263	10.9	0.263	10.5	0.263	10.4	0.263	10.2	0.263	10.8	0.263	10.3	0.263	10.3
33	50N_15C_tmax3_rand	0.263	530.0	0.263	11.3	0.263	11.4	0.263	12.0	0.263	11.7	0.263	11.6	0.263	11.4	0.263	11.6	0.263	11.2	0.263	11.2
34	50N_15C_tmax5_geo	0.463	7200.2	0.463	14.3	0.463	13.7	0.463	14.0	0.463	14.2	0.463	13.6	0.463	13.4	0.463	13.4	0.463	13.4	0.463	13.4
35	50N_15C_tmax5_hier	0.530	7200.2	0.530	15.0	0.530	14.3	0.530	15.0	0.530	14.9	0.530	14.4	0.530	14.6	0.530	14.5	0.530	14.3	0.530	14.3
36	50N_15C_tmax5_rand	0.530	5491.8	0.530	14.4	0.530	15.2	0.530	15.0	0.530	15.4	0.530	15.0	0.530	14.5	0.530	14.8	0.530	14.8	0.530	14.8
37	50N_15C_tmax7_geo	0.664	7200.2	0.664	19.2	0.664	17.6	0.664	18.5	0.664	18.1	0.664	17.6	0.664	17.3	0.664	17.1	0.664	17.0	0.664	17.0

Table B.2 Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for random test instances – continued

Instance		**CPLEX**		t_3		t_4		t_5		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
38	50N_15C_tmax7_hier	0.797	167.0	0.797	18.9	0.797	17.2	0.797	19.1	0.797	18.2	0.797	18.6	0.797	17.6	0.797	17.6	0.797	18.5	0.797	18.5
39	50N_15C_tmax7_rand	0.731	7201.9	0.731	18.3	0.731	18.1	0.731	18.7	0.731	19.0	0.731	17.9	0.731	17.7	0.731	17.8	0.731	18.3	0.731	18.3
40	50N_15C_tmax10_geo	0.931	7201.4	0.931	20.7	0.931	18.7	0.931	19.7	0.931	19.8	0.931	18.9	0.931	19.1	0.931	19.2	0.931	18.9	0.931	18.9
41	50N_15C_tmax10_hier	0.998	7200.3	0.998	19.3	0.998	17.8	0.998	19.0	0.998	19.1	0.998	17.8	0.998	18.1	0.998	17.9	0.998	17.9	0.998	17.9
42	50N_15C_tmax10_rand	0.998	7203.5	0.998	19.0	0.998	19.2	0.998	19.8	0.998	21.1	0.998	18.9	0.998	18.7	0.998	18.9	0.998	18.8	0.998	18.8
43	50N_20C_tmax5_geo	0.346	7200.4	0.346	16.4	0.346	16.7	0.346	17.0	0.346	16.9	0.346	16.5	0.346	16.8	0.346	16.2	0.346	16.9	0.346	16.9
44	50N_20C_tmax5_hier	0.446	42.2	0.446	18.9	0.396	19.2	0.446	19.3	0.446	19.5	0.446	19.3	0.446	18.9	0.446	18.9	0.446	19.0	0.446	19.0
45	50N_20C_tmax5_rand	0.396	7200.2	0.396	18.0	0.396	17.5	0.396	18.4	0.396	18.3	0.396	17.8	0.396	17.8	0.396	17.6	0.396	17.8	0.396	17.8
46	50N_20C_tmax7_geo	0.547	1741.1	0.547	22.0	0.547	21.8	0.547	22.5	0.547	22.7	0.547	21.8	0.547	22.0	0.547	22.0	0.547	22.3	0.547	22.3
47	50N_20C_tmax7_hier	0.597	7200.4	0.597	27.3	0.597	26.8	0.597	28.2	0.597	28.3	0.597	26.8	0.597	26.9	0.597	27.6	0.597	27.4	0.597	27.4
48	50N_20C_tmax7_rand	0.547	7200.8	0.597	23.1	0.597	23.1	0.597	23.4	0.597	23.6	0.597	23.1	0.597	23.1	0.597	23.1	0.597	23.2	0.597	23.2
49	50N_20C_tmax10_geo	0.747	7200.4	0.747	28.6	0.747	28.4	0.747	28.8	0.747	29.3	0.747	27.8	0.747	28.3	0.747	27.8	0.747	27.9	0.747	27.9
50	50N_20C_tmax10_hier	0.848	7203.3	0.848	30.8	0.848	31.3	0.848	32.9	0.848	31.1	0.848	30.5	0.848	30.3	0.848	30.6	0.848	30.7	0.848	30.7
51	50N_20C_tmax10_rand	0.848	7202.0	0.848	28.0	0.848	28.0	0.848	29.0	0.848	29.4	0.848	27.6	0.848	28.0	0.848	27.6	0.848	28.3	0.848	28.3
52	50N_25C_tmax5_geo	0.316	1704.0	0.316	19.5	0.316	19.8	0.316	20.4	0.316	20.1	0.316	20.1	0.316	19.7	0.316	19.7	0.316	19.5	0.316	19.5
53	50N_25C_tmax5_hier	0.356	81.5	0.316	22.0	0.316	21.8	0.316	23.0	0.356	23.0	0.316	22.6	0.316	22.4	0.356	22.4	0.356	22.5	0.356	22.5
54	50N_25C_tmax5_rand	0.356	65.7	0.356	22.1	0.356	21.6	0.356	23.3	0.356	22.1	0.356	22.1	0.356	21.9	0.356	21.5	0.356	21.9	0.356	21.9
55	50N_25C_tmax7_geo	0.476	45.3	0.476	27.4	0.476	27.0	0.476	28.0	0.476	28.6	0.476	28.1	0.476	27.5	0.476	27.2	0.476	26.8	0.476	26.8

Table B.2 Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for random test instances – continued

Instance		**CPLEX**		t_3		t_4		t_5		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
56	50N_25C_tmax7_hier	0.476	7200.4	0.476	31.3	0.476	31.2	0.476	32.9	0.476	33.0	0.476	32.1	0.476	31.2	0.476	32.1	0.476	31.3	0.476	31.3
57	50N_25C_tmax7_rand	0.476	7200.4	0.476	29.4	0.476	29.3	0.476	30.9	0.476	30.5	0.476	30.6	0.476	29.4	0.476	29.9	0.476	29.9	0.476	29.9
58	50N_25C_tmax10_geo	0.637	7202.1	0.637	37.3	0.637	36.8	0.637	38.9	0.637	38.1	0.637	37.4	0.637	36.5	0.637	36.8	0.637	36.7	0.637	36.7
59	50N_25C_tmax10_hier	0.717	7200.4	0.717	39.7	0.717	39.8	0.717	40.8	0.717	40.7	0.717	41.0	0.717	39.4	0.717	39.5	0.717	39.5	0.717	39.5
60	50N_25C_tmax10_rand	0.717	6135.2	0.717	39.3	0.717	38.7	0.717	40.1	0.717	41.5	0.677	40.1	0.677	39.2	0.717	38.4	0.677	38.9	0.677	38.9
61	75N_15C_tmax5_geo	0.530	5046.5	0.530	20.7	0.530	20.0	0.530	20.7	0.530	20.5	0.463	19.9	0.463	20.2	0.530	20.1	0.530	20.5	0.530	20.5
62	75N_15C_tmax5_hier	0.597	7200.3	0.597	23.6	0.597	24.0	0.597	24.0	0.597	24.5	0.597	23.4	0.597	23.8	0.597	23.5	0.597	23.5	0.597	23.5
63	75N_15C_tmax5_rand	0.664	289.7	0.664	24.3	0.664	24.0	0.664	24.8	0.664	24.8	0.664	23.8	0.664	23.9	0.664	23.8	0.664	24.1	0.664	24.1
64	75N_15C_tmax7_geo	0.664	7204.0	0.664	25.3	0.664	25.5	0.664	26.5	0.664	26.3	0.664	25.4	0.664	25.4	0.664	25.6	0.664	25.2	0.664	25.2
65	75N_15C_tmax7_hier	0.798	7201.4	0.798	26.1	0.798	25.8	0.798	27.6	0.798	27.4	0.798	26.3	0.864	26.6	0.864	26.5	0.864	26.7	0.864	26.7
66	75N_15C_tmax7_rand	0.798	7202.7	0.864	27.1	0.864	27.4	0.864	27.7	0.864	28.1	0.864	26.6	0.864	27.2	0.864	27.1	0.864	27.0	0.864	27.0
67	75N_15C_tmax10_geo	0.931	7203.3	0.931	27.8	0.931	27.2	0.931	28.3	0.931	28.6	0.931	27.3	0.931	27.7	0.931	27.3	0.931	27.3	0.931	27.3
68	75N_15C_tmax10_hier	0.931	7207.9	0.998	27.6	0.998	28.3	0.998	28.3	0.998	28.3	0.998	27.1	0.998	27.8	0.998	27.5	0.998	27.5	0.998	27.5
69	75N_15C_tmax10_rand	0.998	7202.7	0.998	27.4	0.998	27.4	0.998	28.9	0.998	28.3	0.998	27.4	0.998	27.3	0.998	27.0	0.998	27.7	0.998	27.7
70	75N_25C_tmax5_geo	0.316	7201.6	0.316	32.1	0.316	32.0	0.316	32.9	0.316	32.1	0.316	32.1	0.316	31.9	0.316	32.0	0.316	31.3	0.316	31.3
71	75N_25C_tmax5_hier	0.436	3071.6	0.396	41.1	0.396	41.2	0.396	43.4	0.396	40.3	0.396	43.5	0.396	42.5	0.396	40.9	0.396	40.7	0.396	40.7
72	75N_25C_tmax5_rand	0.396	7200.3	0.396	39.6	0.396	39.1	0.396	41.5	0.396	40.4	0.396	41.1	0.396	40.4	0.396	39.3	0.396	39.2	0.396	39.2
73	75N_25C_tmax7_geo	0.476	7201.8	0.476	44.6	0.476	44.4	0.476	45.2	0.476	44.7	0.476	44.9	0.476	44.6	0.476	43.9	0.476	43.4	0.476	43.4

Table B.2 Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for random test instances – continued

Instance		**CPLEX**		t_3		t_4		t_5		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
74	75N_25C_tmax7_hier	0.597	1989.4	0.597	49.5	0.557	48.7	0.597	51.0	0.597	52.0	0.557	51.2	0.597	50.7	0.597	48.1	0.597	49.9	0.597	49.9
75	75N_25C_tmax7_rand	0.557	7200.3	0.557	50.4	0.557	50.9	0.557	52.4	0.557	52.3	0.557	51.3	0.557	51.6	0.557	50.8	0.557	51.0	0.557	51.0
76	75N_25C_tmax10_geo	0.717	7201.5	0.677	61.3	0.677	60.6	0.677	62.4	0.677	62.1	0.677	60.6	0.677	60.0	0.677	59.9	0.677	60.3	0.677	60.3
77	75N_25C_tmax10_hier	0.797	7201.0	0.797	60.5	0.797	60.4	0.757	62.1	0.757	61.9	0.797	60.9	0.797	59.4	0.757	58.3	0.757	59.1	0.757	59.1
78	75N_25C_tmax10_rand	0.797	7200.9	0.797	62.0	0.757	60.9	0.757	63.7	0.757	62.5	0.757	61.6	0.757	60.9	0.757	61.5	0.757	60.5	0.757	60.5
	Ave (Geo)	0.592	3765.1	0.587	20.4	0.587	20.1	0.587	20.7	0.583	20.6	0.584	20.4	0.584	20.2	0.583	20.1	0.587	20.1	0.587	20.1
	Ave (Hier)	0.653	3511.1	0.653	21.4	0.649	21.2	0.651	22.2	0.653	21.9	0.651	21.5	0.656	21.3	0.656	21.2	0.656	21.2	0.656	21.2
	Ave (Rand)	0.653	4116.5	0.657	21.3	0.656	21.1	0.656	21.9	0.656	21.9	0.654	21.2	0.654	21.2	0.656	21.0	0.654	21.1	0.654	21.1
	Ave (Overall)	0.633	3797.5	0.632	21.0	0.631	20.8	0.631	21.6	0.631	21.5	0.630	21.0	0.631	20.9	0.631	20.8	0.632	20.8	0.632	20.8

Table B.3: $MSTS_d$ Algorithm Solutions with different tabu tenures for random instances

Instance		% Gap with tabu tenure							
#	**Name**	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
1	30N_10C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
2	30N_10C_tmax3_hier	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**
3	30N_10C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
4	30N_10C_tmax5_geo	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**
5	30N_10C_tmax5_hier	**0.000**	0.004	0.004	0.004	0.004	0.004	0.004	0.004
6	30N_10C_tmax5_rand	**0.010**	0.013	0.013	**0.010**	0.013	**0.010**	0.013	0.013
7	30N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
8	30N_10C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
9	30N_10C_tmax7_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.008	**0.000**	0.003
10	30N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
11	30N_15C_tmax3_hier	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**
12	30N_15C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
13	30N_15C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
14	30N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
15	30N_15C_tmax5_rand	12.603	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**
16	30N_15C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
17	30N_15C_tmax7_hier	0.007	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
18	30N_15C_tmax7_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
19	50N_10C_tmax3_geo	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**
20	50N_10C_tmax3_hier	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**
21	50N_10C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
22	50N_10C_tmax5_geo	**0.004**	14.362	**0.004**	**0.004**	**0.004**	**0.004**	**0.004**	14.361
23	50N_10C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
24	50N_10C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
25	50N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
26	50N_10C_tmax7_hier	-0.005	-0.003	**-0.008**	-0.003	**-0.008**	-0.003	0.000	0.000
27	50N_10C_tmax7_rand	0.003	**0.000**	**0.000**	0.001	0.003	0.003	0.003	**0.000**
28	50N_10C_tmax10_geo	0.007	0.003	0.005	0.007	**0.000**	0.007	0.007	**0.000**
29	50N_10C_tmax10_hier	0.005	0.007	0.007	**0.000**	**0.000**	**0.000**	0.003	0.003
30	50N_10C_tmax10_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
31	50N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
32	50N_15C_tmax3_hier	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**	**0.020**
33	50N_15C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
34	50N_15C_tmax5_geo	**0.000**	**0.000**	0.008	0.012	0.004	**0.000**	**0.000**	0.016
35	50N_15C_tmax5_hier	0.003	0.003	**0.000**	**0.000**	**0.000**	0.003	**0.000**	**0.000**
36	50N_15C_tmax5_rand	0.007	0.004	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
37	50N_15C_tmax7_geo	**0.003**	**0.003**	0.013	**0.003**	**0.003**	0.011	**0.003**	0.005
38	50N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
39	50N_15C_tmax7_rand	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	0.002	**-0.003**	**-0.003**
40	50N_15C_tmax10_geo	0.002	**0.000**	0.002	0.002	0.002	**0.000**	**0.000**	**0.000**
41	50N_15C_tmax10_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.004	0.004	0.005

Table B.3 $MSTS_d$ Algorithm Solutions with different tabu tenures for random instances – continued

Instance		% Gap with tabu tenure							
#	Name	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
42	50N_15C_tmax10_rand	**0.000**	0.003	0.001	**0.000**	**0.000**	**0.000**	0.002	0.003
43	50N_20C_tmax5_geo	0.012	**0.000**	0.012	0.008	0.008	**0.000**	**0.000**	0.012
44	50N_20C_tmax5_hier	**0.000**	**0.000**	11.224	11.224	11.224	11.227	11.224	11.224
45	50N_20C_tmax5_rand	0.006	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
46	50N_20C_tmax7_geo	**0.002**	0.005	0.007	0.007	**0.002**	0.007	**0.002**	**0.002**
47	50N_20C_tmax7_hier	0.008	0.008	**0.006**	0.008	**0.006**	0.008	0.008	0.008
48	50N_20C_tmax7_rand	-9.155	-9.155	**-9.158**	-9.155	-9.155	-9.155	-9.155	-9.155
49	50N_20C_tmax10_geo	0.007	0.002	**0.000**	0.002	0.004	0.002	0.002	0.007
50	50N_20C_tmax10_hier	0.003	0.003	0.003	0.003	**0.002**	0.003	0.006	0.003
51	50N_20C_tmax10_rand	0.003	**0.001**	0.003	**0.001**	**0.001**	0.003	**0.001**	0.003
52	50N_25C_tmax5_geo	**0.000**	**0.000**	0.003	0.003	0.003	**0.000**	0.003	0.003
53	50N_25C_tmax5_hier	**11.258**	**11.258**	**11.258**	**11.258**	**11.258**	**11.258**	**11.258**	**11.258**
54	50N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
55	50N_25C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
56	50N_25C_tmax7_hier	0.005	0.007	0.005	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**
57	50N_25C_tmax7_rand	**0.002**	**0.002**	0.004	**0.002**	**0.002**	0.005	0.005	**0.002**
58	50N_25C_tmax10_geo	0.007	**0.005**	0.007	0.007	**0.005**	0.007	0.006	0.007
59	50N_25C_tmax10_hier	0.002	**0.000**	0.001	0.002	0.001	0.003	0.001	0.002
60	50N_25C_tmax10_rand	**0.000**	0.001	0.003	0.001	0.003	0.003	0.001	0.003
61	75N_15C_tmax5_geo	12.595	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
62	75N_15C_tmax5_hier	0.009	0.009	**0.000**	0.007	0.009	0.012	0.015	**0.000**
63	75N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
64	75N_15C_tmax7_geo	0.005	0.003	**0.000**	0.003	0.005	0.005	0.013	0.011
65	75N_15C_tmax7_hier	-0.001	-0.001	0.000	-0.001	0.006	0.006	-0.001	**-0.002**
66	75N_15C_tmax7_rand	-8.370	**-8.373**	**-8.373**	-8.369	-8.370	-8.371	-8.371	-8.371
67	75N_15C_tmax10_geo	0.008	**0.006**	0.011	**0.006**	0.008	0.007	**0.006**	0.008
68	75N_15C_tmax10_hier	-7.153	-7.162	-7.151	-7.162	-7.159	-7.155	**-7.164**	-7.151
69	75N_15C_tmax10_rand	0.005	**0.000**	0.006	0.003	**0.000**	0.003	0.002	0.005
70	75N_25C_tmax5_geo	0.007	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.003
71	75N_25C_tmax5_hier	**9.183**	9.190	9.188	9.188	9.190	9.190	9.190	9.188
72	75N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
73	75N_25C_tmax7_geo	**0.002**	0.004	**0.002**	**0.002**	**0.002**	**0.002**	0.004	**0.002**
74	75N_25C_tmax7_hier	0.007	0.007	**0.004**	0.007	**0.004**	0.006	6.728	0.007
75	75N_25C_tmax7_rand	**0.004**	**0.004**	**0.004**	**0.004**	**0.004**	**0.004**	**0.004**	**0.004**
76	75N_25C_tmax10_geo	5.591	5.591	5.591	**5.590**	**5.590**	5.592	5.591	5.591
77	75N_25C_tmax10_hier	**0.001**	0.003	0.003	5.030	5.031	0.003	0.003	5.030
78	75N_25C_tmax10_rand	**0.001**	5.031	5.026	0.003	5.030	0.002	0.002	5.031
Averages	Geo	1.255	1.321	0.771	**0.770**	**0.770**	**0.770**	**0.770**	1.323
	Hier	**0.515**	**0.515**	0.946	1.140	1.140	0.947	1.205	1.140
	Rand	-0.187	-0.479	-0.479	**-0.672**	-0.479	**-0.672**	**-0.672**	-0.479
	Overall	0.527	0.453	0.413	0.412	0.477	**0.349**	0.434	0.662

Table B.4: $MSTS_h$ Algorithm Solutions with different tabu tenures for random instances

Instance		% Gap with tabu tenure							
#	**Name**	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
1	30N_10C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
2	30N_10C_tmax3_hier	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**
3	30N_10C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
4	30N_10C_tmax5_geo	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**
5	30N_10C_tmax5_hier	**0.000**	0.004	0.004	0.004	0.004	0.004	0.004	0.004
6	30N_10C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
7	30N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
8	30N_10C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
9	30N_10C_tmax7_rand	0.011	0.013	0.016	**0.003**	0.008	0.008	**0.003**	0.013
10	30N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
11	30N_15C_tmax3_hier	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**
12	30N_15C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
13	30N_15C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
14	30N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
15	30N_15C_tmax5_rand	0.004	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
16	30N_15C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	0.003	**0.000**	0.003	**0.000**
17	30N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
18	30N_15C_tmax7_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
19	50N_10C_tmax3_geo	**0.007**	**0.007**	**0.007**	**0.007**	0.014	**0.007**	**0.007**	**0.007**
20	50N_10C_tmax3_hier	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
21	50N_10C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
22	50N_10C_tmax5_geo	0.004	**0.000**	0.004	14.358	0.004	0.004	14.361	0.004
23	50N_10C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
24	50N_10C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
25	50N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
26	50N_10C_tmax7_hier	**-0.013**	0.000	0.000	-0.008	0.000	**-0.013**	0.000	-0.008
27	50N_10C_tmax7_rand	**0.000**	0.003	0.003	0.003	0.003	0.003	0.003	**0.000**
28	50N_10C_tmax10_geo	**0.000**	0.005	0.003	0.007	0.008	0.003	0.003	**0.000**
29	50N_10C_tmax10_hier	0.007	0.010	**0.005**	0.007	0.007	0.008	0.010	0.007
30	50N_10C_tmax10_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
31	50N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
32	50N_15C_tmax3_hier	**0.013**	**0.013**	0.020	**0.013**	**0.013**	**0.013**	**0.013**	0.020
33	50N_15C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
34	50N_15C_tmax5_geo	0.004	**0.000**	0.004	**0.000**	**0.000**	0.004	**0.000**	**0.000**
35	50N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
36	50N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
37	50N_15C_tmax7_geo	0.003	0.003	0.003	0.003	**0.000**	0.003	0.003	0.003
38	50N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
39	50N_15C_tmax7_rand	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	0.000	**-0.003**	**-0.003**
40	50N_15C_tmax10_geo	0.002	**0.000**	**0.000**	**0.000**	0.002	**0.000**	0.006	0.002

Table B.4 $MSTS_h$ Algorithm Solutions with different tabu tenures for random instances – continued

Instance		% Gap with tabu tenure							
#	**Name**	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
41	50N_15C_tmax10_hier	**0.000**	**0.000**	0.004	**0.000**	0.004	**0.000**	**0.000**	**0.000**
42	50N_15C_tmax10_rand	0.001	**0.000**	**0.000**	0.003	**0.000**	0.002	0.003	**0.000**
43	50N_20C_tmax5_geo	0.004	**0.000**	**0.000**	**0.000**	0.004	0.004	0.012	0.004
44	50N_20C_tmax5_hier	**0.000**	11.224	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
45	50N_20C_tmax5_rand	**0.000**	0.006	**0.000**	**0.000**	0.006	0.006	0.006	**0.000**
46	50N_20C_tmax7_geo	0.005	0.003	0.007	0.005	0.005	**0.002**	0.005	**0.002**
47	50N_20C_tmax7_hier	0.008	0.006	0.006	0.006	**0.000**	**0.000**	0.008	0.006
48	50N_20C_tmax7_rand	**-9.158**	-9.155	**-9.158**	-9.155	**-9.158**	-9.155	-9.155	-9.155
49	50N_20C_tmax10_geo	0.007	0.002	0.002	0.002	0.007	0.002	0.002	**0.000**
50	50N_20C_tmax10_hier	0.003	0.005	0.003	0.003	0.003	0.003	0.003	**0.002**
51	50N_20C_tmax10_rand	**0.001**	**0.001**	**0.001**	**0.001**	**0.001**	0.003	**0.001**	**0.001**
52	50N_25C_tmax5_geo	**0.000**	**0.000**	0.003	**0.000**	0.003	**0.000**	0.003	**0.000**
53	50N_25C_tmax5_hier	11.252	11.252	11.252	**0.000**	11.252	11.252	**0.000**	**0.000**
54	50N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
55	50N_25C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
56	50N_25C_tmax7_hier	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**
57	50N_25C_tmax7_rand	**0.000**	0.002	0.002	0.002	0.002	0.005	0.002	0.002
58	50N_25C_tmax10_geo	**0.000**	**0.000**	0.003	0.007	0.005	0.007	0.003	0.007
59	50N_25C_tmax10_hier	**0.001**	**0.001**	0.002	**0.001**	**0.001**	0.002	**0.001**	**0.001**
60	50N_25C_tmax10_rand	0.003	**0.000**	0.001	0.003	5.588	5.591	**0.000**	5.588
61	75N_15C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	12.595	12.595	**0.000**	**0.000**
62	75N_15C_tmax5_hier	0.009	0.009	0.009	0.010	0.007	0.009	**0.000**	0.009
63	75N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
64	75N_15C_tmax7_geo	0.008	0.005	0.003	**0.000**	0.003	0.003	0.003	0.010
65	75N_15C_tmax7_hier	0.001	0.006	0.006	-0.001	0.006	**-8.364**	**-8.364**	**-8.364**
66	75N_15C_tmax7_rand	-8.370	-8.371	**-8.373**	-8.370	-8.369	-8.371	-8.371	-8.371
67	75N_15C_tmax10_geo	**0.000**	0.010	0.010	0.009	0.007	0.006	0.009	0.010
68	75N_15C_tmax10_hier	-7.153	-7.166	-7.164	**-7.168**	-7.166	-7.166	-7.160	**-7.168**
69	75N_15C_tmax10_rand	**0.000**	**0.000**	**0.000**	0.005	0.002	0.002	0.003	0.003
70	75N_25C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
71	75N_25C_tmax5_hier	9.188	9.188	9.185	9.190	9.188	9.190	**9.183**	9.188
72	75N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
73	75N_25C_tmax7_geo	**0.000**	0.002	0.002	0.004	0.004	0.002	0.004	0.002
74	75N_25C_tmax7_hier	**0.000**	6.726	0.004	0.006	6.726	0.006	**0.000**	0.006
75	75N_25C_tmax7_rand	0.004	0.004	0.004	0.004	0.004	**0.002**	**0.002**	0.004
76	75N_25C_tmax10_geo	5.591	5.591	5.593	**5.587**	5.590	5.590	5.590	5.590
77	75N_25C_tmax10_hier	**0.000**	**0.000**	5.030	5.027	0.001	0.003	5.029	5.028
78	75N_25C_tmax10_rand	**0.002**	5.030	5.026	5.027	5.030	5.030	5.030	5.030
Averages	Geo	**0.769**	**0.769**	0.770	1.321	1.255	1.254	1.322	**0.769**
	Hier	0.513	1.204	0.707	0.274	0.772	0.191	**-0.048**	**-0.048**
	Rand	**-0.673**	-0.479	-0.480	-0.479	-0.264	-0.264	-0.479	-0.264
	Overall	0.203	0.498	0.332	0.372	0.587	0.394	0.265	**0.152**

Table B.5: $MSTS_r$ Algorithm Solutions with different tabu tenures for random instances

Instance		% Gap with tabu tenure							
#	**Name**	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
1	30N_10C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
2	30N_10C_tmax3_hier	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**
3	30N_10C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
4	30N_10C_tmax5_geo	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**
5	30N_10C_tmax5_hier	**0.000**	0.004	0.004	0.004	0.004	0.004	0.004	0.004
6	30N_10C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
7	30N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
8	30N_10C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
9	30N_10C_tmax7_rand	0.013	0.008	0.003	**0.000**	0.008	**0.000**	0.016	0.003
10	30N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	0.000	**0.000**	**0.000**	**0.000**
11	30N_15C_tmax3_hier	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**
12	30N_15C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
13	30N_15C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
14	30N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
15	30N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
16	30N_15C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.003	0.003	0.003
17	30N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
18	30N_15C_tmax7_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
19	50N_10C_tmax3_geo	**0.007**	**0.007**	**0.007**	**0.007**	0.014	0.014	0.014	**0.007**
20	50N_10C_tmax3_hier	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
21	50N_10C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
22	50N_10C_tmax5_geo	0.004	0.004	**0.000**	**0.000**	**0.000**	0.004	0.004	**0.000**
23	50N_10C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
24	50N_10C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
25	50N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
26	50N_10C_tmax7_hier	0.000	-0.008	**-0.013**	0.000	**-0.013**	**-0.013**	0.000	-0.008
27	50N_10C_tmax7_rand	**0.000**	**0.000**	0.003	0.003	0.003	0.003	0.003	**0.000**
28	50N_10C_tmax10_geo	**0.000**	0.003	**0.000**	**0.000**	0.003	0.005	0.003	0.005
29	50N_10C_tmax10_hier	0.005	0.010	0.005	0.007	0.003	0.008	**0.000**	0.005
30	50N_10C_tmax10_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
31	50N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
32	50N_15C_tmax3_hier	**0.013**	0.020	**0.013**	**0.013**	**0.013**	0.020	**0.013**	**0.013**
33	50N_15C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
34	50N_15C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.004	**0.000**	**0.000**
35	50N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
36	50N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
37	50N_15C_tmax7_geo	0.003	0.003	**0.000**	0.003	0.003	0.003	0.003	0.003
38	50N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
39	50N_15C_tmax7_rand	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**	**-0.003**
40	50N_15C_tmax10_geo	0.002	**0.000**	0.002	**0.000**	**0.000**	0.002	**0.000**	**0.000**

Table B.5 $MSTS_r$ Algorithm Solutions with different tabu tenures for random instances – continued

Instance		% Gap with tabu tenure							
#	**Name**	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
41	50N_15C_tmax10_hier	**0.000**	0.007	**0.000**	0.002	0.002	**0.000**	0.009	0.002
42	50N_15C_tmax10_rand	0.002	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
43	50N_20C_tmax5_geo	**0.000**	0.012	**0.000**	**0.000**	**0.000**	**0.000**	0.008	0.012
44	50N_20C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	11.224
45	50N_20C_tmax5_rand	**0.000**	0.003	0.003	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
46	50N_20C_tmax7_geo	0.005	0.005	0.007	0.005	0.005	0.005	**0.002**	0.005
47	50N_20C_tmax7_hier	**0.000**	**0.000**	0.006	0.008	0.006	0.006	0.008	0.006
48	50N_20C_tmax7_rand	**-9.158**	-9.155	-9.155	-9.155	-9.155	-9.155	**-9.158**	-9.155
49	50N_20C_tmax10_geo	0.005	0.005	0.002	0.002	0.002	**0.000**	0.002	0.002
50	50N_20C_tmax10_hier	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**
51	50N_20C_tmax10_rand	**0.001**	**0.001**	**0.001**	**0.001**	0.003	**0.001**	0.003	0.002
52	50N_25C_tmax5_geo	**0.000**	**0.000**	0.003	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
53	50N_25C_tmax5_hier	11.252	11.252	11.252	11.252	11.252	11.252	11.252	**0.000**
54	50N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
55	50N_25C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
56	50N_25C_tmax7_hier	**0.003**	0.005	**0.003**	**0.003**	**0.003**	0.007	**0.003**	**0.003**
57	50N_25C_tmax7_rand	**0.002**	**0.002**	**0.002**	**0.002**	0.005	**0.002**	**0.002**	**0.002**
58	50N_25C_tmax10_geo	0.007	0.007	0.007	**0.000**	0.007	0.007	0.007	0.007
59	50N_25C_tmax10_hier	**0.001**	**0.001**	**0.001**	0.003	0.003	**0.001**	0.002	**0.001**
60	50N_25C_tmax10_rand	**0.000**	**0.000**	0.001	0.003	**0.000**	0.003	0.001	0.003
61	75N_15C_tmax5_geo	12.595	**0.000**	12.595	12.595	12.595	**0.000**	**0.000**	**0.000**
62	75N_15C_tmax5_hier	0.006	0.004	0.009	0.009	0.012	**0.000**	0.009	0.012
63	75N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
64	75N_15C_tmax7_geo	0.003	0.000	0.000	0.005	**-10.049**	0.000	0.003	0.006
65	75N_15C_tmax7_hier	0.004	-0.002	0.006	**-8.364**	0.006	0.013	0.006	-0.001
66	75N_15C_tmax7_rand	-8.371	**-8.372**	-8.370	-8.371	-8.371	-8.370	-8.370	-8.371
67	75N_15C_tmax10_geo	0.007	0.008	**0.006**	**0.006**	**0.006**	0.015	**0.006**	0.013
68	75N_15C_tmax10_hier	-7.160	-7.156	**-7.166**	-7.156	-7.155	-7.160	-7.162	-7.158
69	75N_15C_tmax10_rand	0.010	**0.000**	0.005	0.003	0.006	0.002	0.006	0.003
70	75N_25C_tmax5_geo	**0.000**	0.007	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
71	75N_25C_tmax5_hier	9.190	**9.185**	9.190	**9.185**	9.188	9.188	9.188	9.190
72	75N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
73	75N_25C_tmax7_geo	0.002	**0.000**	0.002	0.002	**0.000**	**0.000**	0.004	0.004
74	75N_25C_tmax7_hier	6.725	0.007	0.004	6.726	0.004	**0.000**	0.005	0.005
75	75N_25C_tmax7_rand	**0.002**	0.004	0.004	0.004	0.004	**0.002**	**0.002**	0.004
76	75N_25C_tmax10_geo	**5.590**	**5.590**	5.591	5.591	**5.590**	5.591	5.591	5.593
77	75N_25C_tmax10_hier	5.030	**0.001**	0.003	5.030	5.028	0.003	5.027	5.030
78	75N_25C_tmax10_rand	5.030	5.030	**0.002**	5.026	5.030	**0.002**	5.026	**0.002**
Averages	Geo	1.254	**0.770**	1.253	1.253	0.867	**0.770**	**0.770**	**0.770**
	Hier	0.965	0.514	**0.513**	0.644	0.707	0.514	0.707	0.706
	Rand	-0.479	-0.480	**-0.673**	-0.480	-0.479	**-0.673**	-0.479	**-0.673**
	Overall	0.580	0.268	0.365	0.473	0.365	**0.204**	0.333	0.268

Table B.6: $MSTS_h$ heuristic solutions with different diversification parameters for random instances

Instance		% Gap with diversification parameter						
#	**Name**	$p_{0.20}$	$p_{0.25}$	$p_{0.30}$	$p_{0.35}$	$p_{0.40}$	$p_{0.45}$	$p_{0.50}$
1	30N_10C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
2	30N_10C_tmax3_hier	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**	**0.007**
3	30N_10C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
4	30N_10C_tmax5_geo	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**	**14.365**
5	30N_10C_tmax5_hier	**0.000**	0.004	**0.000**	0.008	0.004	0.004	0.004
6	30N_10C_tmax5_rand	0.010	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
7	30N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
8	30N_10C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
9	30N_10C_tmax7_rand	**0.000**	0.013	0.011	**0.000**	0.016	0.008	**0.000**
10	30N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
11	30N_15C_tmax3_hier	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	**0.013**	0.020
12	30N_15C_tmax3_rand	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**	**0.006**
13	30N_15C_tmax5_geo	0.001	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
14	30N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
15	30N_15C_tmax5_rand	0.007	**0.000**	0.003	0.004	**0.000**	**0.000**	**0.000**
16	30N_15C_tmax7_geo	0.003	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
17	30N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
18	30N_15C_tmax7_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
19	50N_10C_tmax3_geo	**0.000**	**0.000**	**0.000**	0.007	0.007	0.007	0.014
20	50N_10C_tmax3_hier	0.006	0.006	**0.000**	0.006	0.006	0.006	0.020
21	50N_10C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
22	50N_10C_tmax5_geo	**0.000**	**0.000**	0.004	14.358	**0.000**	0.004	14.380
23	50N_10C_tmax5_hier	0.007	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
24	50N_10C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
25	50N_10C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
26	50N_10C_tmax7_hier	0.000	-0.005	0.000	0.000	-0.011	**-0.013**	**-0.013**
27	50N_10C_tmax7_rand	**0.000**	0.003	0.001	**0.000**	0.003	0.003	0.003
28	50N_10C_tmax10_geo	0.003	0.003	**0.000**	0.003	0.007	0.003	0.005
29	50N_10C_tmax10_hier	0.005	**0.000**	0.003	0.003	**0.000**	0.008	0.008
30	50N_10C_tmax10_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
31	50N_15C_tmax3_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
32	50N_15C_tmax3_hier	0.006	0.006	**0.000**	0.020	0.013	0.013	0.020
33	50N_15C_tmax3_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
34	50N_15C_tmax5_geo	0.012	**0.000**	**0.000**	**0.000**	0.004	0.004	**0.000**
35	50N_15C_tmax5_hier	**0.000**	**0.000**	**0.000**	0.003	**0.000**	**0.000**	**0.000**
36	50N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
37	50N_15C_tmax7_geo	0.003	0.003	0.003	**0.000**	0.011	0.003	0.003
38	50N_15C_tmax7_hier	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
39	50N_15C_tmax7_rand	0.005	**-0.003**	**-0.003**	**-0.003**	**-0.003**	0.000	**-0.003**
40	50N_15C_tmax10_geo	0.002	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**

Table B.6 $MSTS_r$ Algorithm Solutions with different diversification parameters for random instances – continued

Instance		% Gap with diversification parameter						
#	**Name**	$p_{0.20}$	$p_{0.25}$	$p_{0.30}$	$p_{0.35}$	$p_{0.40}$	$p_{0.45}$	$p_{0.50}$
41	50N_15C_tmax10_hier	**0.000**	0.007	0.005	**0.000**	0.007	**0.000**	**0.000**
42	50N_15C_tmax10_rand	**0.000**	0.002	**0.000**	0.007	**0.000**	0.002	0.003
43	50N_20C_tmax5_geo	**0.000**	**0.000**	**0.000**	**0.000**	0.012	0.004	**0.000**
44	50N_20C_tmax5_hier	0.000	11.224	11.221	0.000	11.221	**0.000**	**0.000**
45	50N_20C_tmax5_rand	0.003	0.006	**0.000**	0.006	**0.000**	0.006	0.006
46	50N_20C_tmax7_geo	0.005	0.005	0.005	0.002	0.005	0.002	0.005
47	50N_20C_tmax7_hier	0.006	0.006	0.006	0.006	0.006	**0.000**	0.006
48	50N_20C_tmax7_rand	**-9.155**	**-9.155**	**-9.155**	**-9.155**	**-9.155**	**-9.155**	**-9.155**
49	50N_20C_tmax10_geo	0.002	0.002	0.002	**0.000**	0.002	0.002	0.002
50	50N_20C_tmax10_hier	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**
51	50N_20C_tmax10_rand	**0.001**	0.003	**0.001**	**0.001**	0.003	0.003	0.003
52	50N_25C_tmax5_geo	0.003	0.003	0.003	0.003	**0.000**	**0.000**	**0.000**
53	50N_25C_tmax5_hier	11.258	11.255	**0.000**	**0.000**	**0.000**	11.252	11.252
54	50N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
55	50N_25C_tmax7_geo	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
56	50N_25C_tmax7_hier	**0.003**	0.005	**0.003**	**0.003**	**0.003**	**0.003**	**0.003**
57	50N_25C_tmax7_rand	0.002	**0.000**	0.002	0.002	0.004	0.005	0.004
58	50N_25C_tmax10_geo	0.005	**0.003**	**0.003**	0.007	0.008	0.007	0.005
59	50N_25C_tmax10_hier	0.001	**0.000**	0.001	**0.000**	0.001	0.002	0.002
60	50N_25C_tmax10_rand	0.003	**0.000**	0.001	0.003	0.001	5.591	**0.000**
61	75N_15C_tmax5_geo	12.595	**0.000**	12.595	12.595	12.595	12.595	12.595
62	75N_15C_tmax5_hier	0.012	0.012	**0.009**	0.012	**0.009**	**0.009**	0.010
63	75N_15C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
64	75N_15C_tmax7_geo	0.003	**0.000**	0.005	**0.000**	0.005	0.003	0.003
65	75N_15C_tmax7_hier	-0.002	-0.002	0.004	-0.001	**-8.364**	**-8.364**	**-8.364**
66	75N_15C_tmax7_rand	**-8.371**	**-8.371**	-8.370	-8.370	**-8.371**	**-8.371**	**-8.371**
67	75N_15C_tmax10_geo	0.008	**0.004**	0.011	0.007	0.012	0.006	0.009
68	75N_15C_tmax10_hier	**-7.172**	-7.160	-7.162	-7.168	-7.157	-7.166	-7.162
69	75N_15C_tmax10_rand	0.002	0.002	0.003	0.003	**0.000**	0.002	0.003
70	75N_25C_tmax5_geo	**0.000**	0.010	**0.000**	0.007	**0.000**	**0.000**	**0.000**
71	75N_25C_tmax5_hier	9.190	18.383	9.188	9.190	9.188	9.190	**9.185**
72	75N_25C_tmax5_rand	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
73	75N_25C_tmax7_geo	0.002	0.002	0.004	**0.000**	0.002	0.002	0.004
74	75N_25C_tmax7_hier	**0.002**	0.006	6.724	0.004	0.006	0.006	0.006
75	75N_25C_tmax7_rand	0.004	0.004	0.004	0.004	0.004	**0.002**	0.004
76	75N_25C_tmax10_geo	5.591	**5.590**	**5.590**	5.591	5.591	**5.590**	5.593
77	75N_25C_tmax10_hier	5.030	**0.001**	5.031	**0.001**	**0.001**	0.003	**0.001**
78	75N_25C_tmax10_rand	5.026	5.026	5.028	**0.003**	5.030	5.030	5.030
Averages	Geo	1.254	**0.769**	1.253	1.806	1.255	1.254	1.807
	Hier	0.707	1.299	0.964	**0.081**	0.191	0.191	0.193
	Rand	-0.479	-0.479	-0.479	**-0.672**	-0.479	-0.264	-0.479
	Overall	0.494	0.530	0.579	0.405	**0.322**	0.394	0.507

APPENDIX C

SOLUTIONS OF 2011 VAN EARTHQUAKE CASE STUDY INSTANCES

Table C.1: Route and option assignment in case study CPLEX solutions

Instance			
#	Name	Route	option #
1	93N_16C_tmax10	0-1-55-78-50-9-59-2-0	6-6-6-6-6-6-6
2	93N_16C_tmax15	0-1-55-78-2-53-49-91-15-63-79-37-56-0	7-7-6-7-6-6-6-6-6-5-6-6
3	93N_16C_tmax20	0-1-55-78-9-2-53-49-15-91-63-79-3-73-81-42-0	5-5-5-5-5-6-5-4-5-5-4-6-5-4-5
4	93N_23C_tmax10	0-50-9-2-19-85-78-57-55-1-0	6-7-6-6-6-5-7-6-6
5	93N_23C_tmax15	0-50-53-2-59-30-24-54-13-72-66-78-57-55-1-0	7-7-6-7-7-7-7-7-7-7-7-7-6-6
6	93N_23C_tmax20	0-61-1-55-57-78-31-50-9-24-36-5-30-59-2-53-58-0	5-6-6-6-5-6-6-6-6-6-6-6-6-6-6-5
7	93N_23C_tmax25	0-58-79-37-73-45-81-80-76-52-38-18-7-53-2-9-19-85-78-57-55-1-61-0	6-5-5-6-6-6-6-6-6-6-6-6-6-6-6-6-6-6-6-6-7-7
8	93N_29C_tmax10	0-1-55-78-66-85-19-9-59-2-0	7-6-7-6-6-6-7-7-6
9	93N_29C_tmax15	0-58-1-55-78-66-85-19-9-59-54-30-2-53-0	7-6-7-7-6-6-6-6-6-7-6-6-6
10	93N_29C_tmax20	0-1-55-78-66-19-2-53-49-91-15-63-79-37-73-81-39-47-90-46-93-0	7-6-7-7-7-7-6-6-6-7-7-6-7-7-7-7-7-7-7-7
11	93N_29C_tmax25	0-58-21-84-12-70-75-46-76-41-52-38-18-7-92-53-2-59-9-19-85-66-78-55-1-0	7-6-7-6-5-7-7-6-7-7-7-7-6-6-6-6-6-6-6-6-6-7-6-7

Table C.2: Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for the Van Case Study Instances

Instance			CPLEX		t_3		t_4		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	p_{div}	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
1	93N_16C_tmax10	$p_{0.4}$	0.434	7200.52	0.434	15.33	0.434	15	0.434	10.98	0.434	27.58	0.434	10.88	0.434	12.19	0.434	15.33	0.434	25.69
		$p_{0.45}$	0.434	7200.52	0.434	31.25	0.434	31.23	0.434	30.89	0.434	31.06	0.434	27.61	0.434	27.3	0.434	31.25	0.434	32.17
		$p_{0.5}$	0.434	7200.52	0.434	20.95	0.434	31.33	0.434	33.59	0.434	12.69	0.434	24.59	0.434	27.69	0.434	20.95	0.434	27.95
2	93N_16C_tmax15	$p_{0.4}$	0.747	7200.41	0.810	25.78	0.810	27.59	0.810	8.187	0.810	34.77	0.747	11.41	0.810	14.67	0.810	25.78	0.747	6.375
		$p_{0.45}$	0.747	7200.41	0.810	7.421	0.810	16.67	0.810	34.11	0.810	34.2	0.810	6.875	0.810	29.92	0.810	7.421	0.810	12.97
		$p_{0.5}$	0.747	7200.41	0.747	14.98	0.810	9.25	0.810	28.66	0.810	5.015	0.810	34.11	0.810	23.14	0.747	14.98	0.810	10.06
3	93N_16C_tmax20	$p_{0.4}$	0.936	7202.41	0.998	32.13	0.998	5.5	0.998	19.36	0.998	5.687	0.998	30.22	0.998	22.52	0.998	32.13	0.998	36.77
		$p_{0.45}$	0.936	7202.41	0.998	17.38	0.998	27.84	0.998	4.234	0.998	29.73	0.998	15.56	0.998	25.81	0.998	17.38	0.998	9.281
		$p_{0.5}$	0.936	7202.41	0.998	31.02	0.998	35.91	0.998	5.64	0.998	14.44	0.998	35.19	0.998	17.27	0.998	31.02	0.998	27.27
4	93N_23C_tmax10	$p_{0.4}$	0.388	7200.53	0.388	35.34	0.388	40.84	0.388	38.7	0.388	35.17	0.388	38.2	0.388	37.83	0.388	35.34	0.388	33.94
		$p_{0.45}$	0.388	7200.53	0.388	38.59	0.388	34.52	0.388	40.86	0.388	34.95	0.388	39.42	0.388	33.42	0.388	38.59	0.388	35.77
		$p_{0.5}$	0.388	7200.53	0.388	37.84	0.388	27.39	0.388	40.34	0.388	38.59	0.388	39.44	0.388	40.95	0.388	37.84	0.388	29.67
5	93N_23C_tmax15	$p_{0.4}$	0.606	7211.49	0.562	57.28	0.606	26.08	0.562	12.48	0.562	23.69	0.606	28.16	0.562	21.89	0.562	57.28	0.606	9.00
		$p_{0.45}$	0.606	7211.49	0.606	6.843	0.606	47.19	0.606	13.66	0.562	50.88	0.562	48.41	0.562	17.91	0.606	6.843	0.562	26.38
		$p_{0.5}$	0.606	7211.49	0.562	8.296	0.562	6.14	0.562	11.95	0.562	53.78	0.562	30.72	0.562	58.56	0.562	8.296	0.562	23.33
6	93N_23C_tmax20	$p_{0.4}$	0.693	7203.27	0.824	39.41	0.824	25.88	0.780	63.42	0.780	45.72	0.780	51.97	0.824	56.69	0.824	39.41	0.780	21.31
		$p_{0.45}$	0.693	7203.27	0.780	21.48	0.824	49.7	0.780	64.16	0.824	20.64	0.824	32.94	0.780	49.53	0.780	21.48	0.780	17.8
		$p_{0.5}$	0.693	7203.27	0.824	42.58	0.824	28.75	0.780	48.08	0.824	65.83	0.824	4.609	0.780	4.765	0.824	42.58	0.780	12.84
7	93N_23C_tmax25	$p_{0.4}$	0.955	7205.82	0.955	9.687	0.955	11.34	0.954	48.45	0.954	6.875	0.955	44.2	0.955	17.09	0.955	9.687	0.955	5.265
		$p_{0.45}$	0.955	7205.82	0.955	38.58	0.955	48.09	0.955	35.72	0.954	60.14	0.954	50.88	0.955	63.88	0.955	38.58	0.955	30.91
		$p_{0.5}$	0.955	7205.82	0.954	20.63	0.955	48.42	0.954	10.86	0.954	22.52	0.954	20.72	0.954	10.14	0.954	20.63	0.955	5.39

Table C.2 Objective function values and CPU times (in seconds) of CPLEX and $MSTS_h$ Algorithm for the Van Case Study instances – continued

Instance			**CPLEX Results**		t_3		t_4		t_5		t_6		t_7		t_8		t_9		t_{10}	
#	**Name**	p_{div}	z*	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU	z_h	CPU
8	93N_29C_tmax10	$p_{0.4}$	0.306	7200.54	0.306	32.92	0.306	30.36	0.306	17.47	0.306	16.23	0.306	37.2	0.306	12.97	0.306	32.92	0.306	3.687
		$p_{0.45}$	0.306	7200.54	0.306	17.97	0.306	25.95	0.306	7.296	0.306	30.09	0.306	22.78	0.306	11.14	0.306	17.97	0.306	19.91
		$p_{0.5}$	0.306	7200.54	0.306	9.156	0.306	22.14	0.306	28.17	0.306	21.77	0.306	30.92	0.306	37.38	0.306	9.156	0.306	32.17
9	93N_29C_tmax15	$p_{0.4}$	0.445	7204.81	0.445	58.31	0.514	12.67	0.514	10.89	0.514	10.36	0.479	14.69	0.479	8.453	0.445	58.31	0.479	8.5
		$p_{0.45}$	0.445	7204.81	0.479	8.359	0.514	6.703	0.514	11.84	0.479	9.031	0.479	31.11	0.514	4.5	0.479	8.359	0.479	33.3
		$p_{0.5}$	0.445	7204.81	0.479	30.58	0.479	8.109	0.479	14.05	0.479	12.02	0.479	7.562	0.479	52.08	0.479	30.58	0.479	8.671
10	93N_29C_tmax20	$p_{0.4}$	0.687	7226.7	0.687	15.11	0.687	55.95	0.721	16.64	0.721	74.09	0.687	12.69	0.687	58.41	0.687	15.11	0.721	57.95
		$p_{0.45}$	0.687	7226.7	0.687	54.66	0.721	47.67	0.687	54.09	0.687	30.94	0.687	63.14	0.721	47.98	0.687	54.66	0.687	51.64
		$p_{0.5}$	0.687	7226.7	0.721	74.95	0.721	29.38	0.687	75.63	0.721	75.73	0.687	33.95	0.721	36.17	0.721	74.95	0.687	56.83
11	93N_29C_tmax25	$p_{0.4}$	0.825	7204.5	0.860	83.98	0.860	55.47	0.860	81.25	0.860	67.39	0.860	31.44	0.825	71.44	0.860	83.98	0.860	19.78
		$p_{0.45}$	0.825	7204.5	0.860	11.34	0.860	9	0.825	61.19	0.860	58.02	0.860	37.56	0.860	49.23	0.860	11.34	0.860	33.36
		$p_{0.5}$	0.825	7204.5	0.860	51.17	0.860	73.58	0.860	27.58	0.860	21.97	0.860	67.14	0.860	48.67	0.860	51.17	0.860	24.08
	Averages	$p_{0.4}$	0.638	7205.55	0.661	36.84	0.671	27.88	0.666	29.8	0.666	31.6	0.658	28.28	0.661	30.38	0.661	36.84	0.661	20.75
		$p_{0.45}$	0.638	7205.55	0.664	23.08	0.674	31.32	0.664	32.55	0.664	35.43	0.664	34.21	0.666	32.78	0.664	23.08	0.660	27.59
		$p_{0.5}$	0.638	7205.55	0.661	31.1	0.667	29.13	0.660	29.5	0.667	31.3	0.664	29.9	0.663	32.44	0.661	31.1	0.660	23.48
		Overall	0.638	7205.55	0.662	30.34	0.671	29.44	0.663	30.62	0.666	32.78	0.662	30.8	0.663	31.87	0.662	30.34	0.660	23.94