



MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



ÇOK AMAÇLI ÇOK ATAMALI ANA DÜĞÜM AĞ TASARIMI PROBLEMİ İÇİN METASEZGİSEL YAKLAŞIMLAR

İBRAHİM DEMİR

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Programı

DANIŞMAN

Dr. Öğr. Üyesi Fatma CORUT ERGİN

EŞ-DANIŞMAN

Dr. Öğr. Üyesi Berna KİRAZ

İSTANBUL, 2019



MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES



HEURISTIC APPROACHES FOR MULTI- OBJECTIVE MULTIPLE ALLOCATION HUB LOCATION PROBLEM

İBRAHİM DEMİR

MASTER THESIS

Department of Computer Engineering

Thesis Supervisor

Asst. Prof. Dr. Fatma CORUT ERGİN

Thesis CO- Supervisor

Asst. Prof. Dr. Berna KİRAZ

ISTANBUL, 2019

MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES IN
PURE AND APPLIED SCIENCES

İbrahim DEMİR, a Master of Science student of Marmara University Institute for Graduate Studies in Pure and Applied Sciences, defended his thesis entitled “**Heuristic approaches for multi-objective multiple allocation hub location problem**”, on Jul 11, 2019 and has been found to be satisfactory by the jury members.

Jury Members

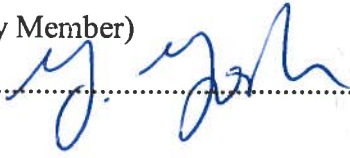
Asst.Prof.Dr. Fatma CORUT ERGİN(Advisor)

Marmara University


Asst.Prof.Dr. Ali Haydar ÖZER (Jury Member)

Marmara University


Assoc.Prof.Dr. Yusuf YASLAN (Jury Member)

Istanbul Technical University


APPROVAL

Marmara University Institute for Graduate Studies in Pure and Applied Sciences Executive Committee approves that İbrahim DEMİR be granted the degree of Master of Science in Department of Computer Engineering, Computer Engineering Program on 21.08.2019
(Resolution no: 2019/17.02).

Director of the Institute

Prof. Dr. Bülent EKİCİ



ACKNOWLEDGEMENT

I would like to express my emphatic gratitude for the support, advice, and extensive knowledge so generously provided by my advisors, Assist. Prof. Fatma CORUT ERGİN and Assist. Prof. Berna KİRAZ, throughout my studies, including their steadfast commitment to very early weekly meetings.

Furthermore, I would like to thank members of thesis committee, Assist. Prof. Fatma CORUT ERGİN, Assist. Prof. Ali Haydar ÖZER, Assoc. Prof. Yusuf YASLAN for their valuable contributions.

In addition, I would like to thank Prof. Haluk Rahmi TOPÇUOĞLU for his guidance in selecting a topic of study, as well as his advice on the journal.

Moreover, I would like to thank all of my professors in the Computer Engineering Departments of Marmara University and Anadolu University, as well as in the Electrical & Electronics Engineering Department of Anadolu University, with special thanks to Prof. Haluk Rahmi TOPÇUOĞLU, Assoc. Prof. Cüneyt AKINLAR, Prof. Ömer Nezi̇h GEREK, Assoc. Prof. Emin GERMEN, for helping to build my engineering background and inspiring both my work and studies.

Finally, I would like to thank my family, especially my wife and my mother, for their unwavering patience, encouragement, and support throughout my studies.

Contents

•	ACKNOWLEDGEMENT	i
•	ÖZET	ii
•	ABSTRACT	iii
•	SYMBOLS	iv
•	ABBREVIATIONS	vi
•	LIST OF FIGURES	vii
•	LIST OF TABLES.....	ix
•	1. INTRODUCTION	1
•	2. PROBLEM STATEMENT AND RELATED WORK	4
•	2.1. MOCMAHLP NETWORK AND ROUTING	6
•	3. METAHEURISTICS FOR MULTI-OBJECTIVE HLP	11
•	3.1. NSGA-II	11
•	3.2. AMOSA	12
•	4. MATHEMATICAL MODEL.....	13
•	5. METAHEURISTIC APPROACHES	18
•	5.1. SOLUTION REPRESENTATION	19
•	5.2. REPAIR FUNCTION	20
•	5.3. ROUTING	21
•	5.4. NSGA-II BASED APPROACH.....	23
•	5.4.1. Crossover OPERATOR	23
•	5.4.2. MUTATION OPERATORS.....	24
•	5.4.2.1. ARC MUTATIONS	24
•	5.4.2.2. HUB MUTATION	26
•	5.5. AMOSA BASED APPROACH.....	26
•	5.5.1. HUB SWITCH NEIGHBOR	27

• 5.5.2.	ARC SWITCH NEIGHBOR	28
• 5.5.2.1.	SWAP ALL ARCS-1	28
• 5.5.2.2.	SWAP ALL ARCS-2	28
• 5.5.2.3.	ADD/REMOVE K-ARCS	28
• 5.5.2.4.	ADAPTIVE ADD/REMOVE K-ARCS	30
• 5.5.2.5.	RANDOM ADD/REMOVE K-ARCS	30
• 5.5.2.1.	TEMPERATURE BASED ADD/REMOVE K-ARCS	30
• 6.	EXPERIMENTAL STUDY	31
• 6.1.	MONITORING INTERFACE APPLICATION	31
• 6.2.	DATA SET	36
• 6.2.1.	TURKISH DATA SET	36
• 6.2.2.	AP DATA SET	38
• 6.3.	PERFORMANCE METRIC	39
• 6.4.	SOLUTIONS FOR SMALL DATA SETS	40
• 6.4.1.	5-NODE REDUCED EAGEAN AREA DATA SET	41
• 6.4.2.	7-NODE REDUCED TURKISH DATA SET	42
• 6.5.	EXPERIMENTS ON AP DATA SET	45
• 6.6.	EXPERIMENTS ON TURKISH DATA SET	50
• 7.	CONCLUSION AND FUTURE WORK	61
• 8.	CV	62
• 9.	REFERENCES	64

ÖZET

ÇOK AMAÇLI ÇOK ATAMALI ANA DÜĞÜM AĞ TASARIMI PROBLEMİ İÇİN METASEZGİSEL YAKLAŞIMLAR

Bu tezde, çok amaçlı çok atamalı ana düğüm ağ tasarımı ve yönlendirme problemi için yeni bir model sunuyoruz. Model, ağdaki ana düğümlerin atanmasını, ana düğümler arası ve ana düğüm ve düğümler arası ağın tasarımı, ve her bir kaynak-hedef düğüm ikilileri için ağ içerisindeki yönlendirmenin tasarlanmasını kapsar. Seçilen ana düğümlerin herbirinin kendi içerisinde bağlı olma koşulu yoktur, ve ağdaki her bir ana düğüm ve düğümler arası bağlantıların önceden belirlenmiş kapasite kriterleri vardır. Çok amaçlı problemin amaçları; toplam sabit masrafların ve yönlendirme masraflarının toplamının minimizasyonu ve yönlendirmedeki maksimum ulaştırma süresinin minimizasyonudur. Bu tezde çok amaçlı problem için matematiksel formül tasarlanmış ve bu alanda sıklıkla kullanılan çok amaçlı genetik algoritma ve benzetimli tavlama tabanlı meta-sezgisel çözümler sunulmuştur. Tasarlanan matematiksel formülasyonu kullanarak, 5 düğümlü ve 7 düğümlü küçük ağlar için optimum çözümü bulabiliyoruz. Geliştirdiğimiz sezgisel yaklaşımın performansını gerçek verilerle ölçmek için, hesaplama deneyleri 20 düğüme indirgenmiş Avustralya posta veri seti ve Türk posta sistemi veri seti üzerinde yapılmıştır. Karşılaştırmalı kıyaslamalar geliştirilmiş tüm sezgisel operatörler için yapılmış, en iyi konfigürasyonların sonuçları tartışılmıştır. Sonuçlar, tasarladığımız sezgisel yaklaşımın makul süreler içerisinde; Avustralya posta sistemi için 15 saniyeden kısa bir sürede ve Türk posta sistemi için 10 dakikadan kısa bir sürede olası çözümler bulabilmiştir.

ABSTRACT

HEURISTIC APPROACHES FOR MULTI-OBJECTIVE MULTIPLE ALLOCATION HUB LOCATION PROBLEM

In this thesis, we propose a new model for the multi-objective multiple allocation hub network design and routing problem which contains determining the location of hubs, the design of hub network, and the routing of commodities between source-destination pairs in the given network. The selected hubs are not assumed to be fully connected, and each node and arc in the network has capacity constraints. The multiple objectives of the problem are the minimization of total fixed and transportation costs and the minimization of the maximum travel time required for routing. We propose a mathematical formulation for the multi-objective problem and present 2 meta-heuristic solutions, one based on a well-known multi-objective evolutionary algorithm and the other based on simulated annealing algorithm. Using the proposed formulation, we are able to find optimal solution for small networks of 5 nodes and 7 nodes. To evaluate the performance of our heuristic approach on real data, the computational experiments are conducted on reduced AP data set with 20 nodes and Turkish postal system data set. Comparative benchmarks are applied to a number of heuristic operators and results of best configurations are discussed. The results demonstrate that our heuristic approach can find feasible solutions to the problem in reasonable execution time, which is less than 15 seconds for AP data set and less than 10 minutes for Turkish postal system data set.

SYMBOLS

N	Number of nodes ($N = V $)
H	Set of possible hub locations
a_u	The initial node of arc $a \in E$
a_v	The terminal node of arc $a \in E$
T_a	Time required to transfer over arc $a \in E$
F_a	Total amount of flow traversing arc $a \in E$
C_a	Cost of delivery on arc $a \in E$
λ_a	Fixed cost of arc $a \in E$
Q_a	Capacity of arc $a \in E$
i_k	The source node of commodity k
j_k	The destination node of commodity k
D_k	Total demand of flow for commodity k
D_{kp}	Demand of flow for package p of commodity k
f_a^k	The flow of commodity k on arc $a \in E$
κ_h	Fixed cost of hub $h \in H$
Q_h	Capacity of hub $h \in H$
χ	Collection discount factor
α	Discount factor of hub to hub transportation
δ	Distribution discount factor

Z_a	Decision variable of arc $a \in E$
x_i	Hub decision variable of node i
x'_i	Inversed hub decision variable of node i
A_{kp}^a	Decision variable of routing
D_{kp}	Decision variable of demand
f_a^k	Total flow of commodity k on arc a
ψ	Hub capacity multiplier
Q_c	Capacity of city c
P_c	Population of city c
F^c	Total flow demand of city c
ω	Link capacity multiplier
d_a	Distance of arc a
T_a	Time required to travel through arc a
popsize	Population size
p_c	Crossover probability
p_{bfm}	Bit-flip mutation probability
p_{sm}	Swap mutation probability
N_{ipt}	Number of iterations per temperature
$N_{maxIter}$	Maximum number of iterations
N_{hci}	Number of hill climbing iterations
N_{sl}	Soft Limit

ABBREVIATIONS

AMOSA	Simulated Annealing-Based Multi-objective Optimization Algorithm
AP	Australian Post
HLP	Hub Location Problem
MOABC	Multiple Artificial Bee Colony
MOCMAHLP	Multi-Objective Capacitated Multiple Allocation Hub Location Problem
MOEA	Multi-Objective Evolutionary Algorithm
MOIWO	Multi-Objective Invasive Weed Optimization
NSGA	Non-dominated Sorting Genetic Algorithm
SMOSA	The Suppapitnarm Multi-Objective Simulated Annealing

LIST OF FIGURES

Figure 2-1 An example of P-Hub Location Problem while $P = 5$ for a 9-node network, with nodes 3,4,5,6 and 7 selected as hubs	7
Figure 2-2 An example multi allocation hub location problem with 9 nodes	8
Figure 2-3 A sample network for not interconnected multi allocation hub location problem	9
Figure 2-4 A Sample routing for capacitated multi allocation not interconnected hub location problem	10
Figure 5-1 An example of hub decision array for a 9-node network	19
Figure 5-2 An example of link decision array for a 9-node network, with nodes 3,4,5,6 and 7 being hubs.....	20
Figure 5-3 The illustration of resulting network for the hub decision and link decision arrays in Figure 5-1 and Figure 5-2	20
Figure 5-3 Routing example of three source-destination pairs for the given 9-node network in Figure 5-3	22
Figure 5-4 Crossover operation on hub decision and link decision. Figures at top illustrate parents and figures at bottom illustrate resulted offspring	24
Figure 5-5 Swap All Arcs Mutation - First decision variables shows the initial mapping, second decision variables shows after bit flip operation on only node 2, third decision variables shows after swapping all arcs of nodes 4 and 5 and repairing	29
Figure 6-1 Monitoring interface application – Main view of Turkish dataset	32
Figure 6-2 Monitoring interface application – Show Only Hubs section.....	33
Figure 6-3 Monitoring interface application - Show All section.....	33
Figure 6-4 Monitoring interface application - Show Only Hubs section - Routing from Istanbul to Bursa	34
Figure 6-5 Monitoring interface application - Show Only Hubs section - Routing from Edirne to Hakkari.....	35

Figure 6-6 Monitoring interface application - Show Only Hubs section - Routing from Trabzon to Tokat - Multiple path case.....	35
Figure 6-7 Hypervolume indicator	39
Figure 6-8 Pareto-optimal solutions obtained by brute-force method for 5-node network (Upper solutions have 1, lower solutions have 2 hubs	42
Figure 6-9 Pareto-optimal solutions obtained by brute-force method for 7-node network (From up-left corner to down-right corner the number of hubs at each case is equal to 1, 1, 2, 2, 2 and 3 hubs)	43
Figure 6-10 Box-plot of ratio of average hypervolume value of 200 runs of regarding algorithm's heuristic operators to hypervolume value of linear solution for 500 different datasets reduced from Turkish dataset. Each run is limited with 20.000 number of evaluations and default parameters are used for every neighborhood.....	44
Figure 6-11 Pareto-front solutions obtained by AMOSA algorithm on 20-Node network	46
Figure 6-12 Box-plot of hypervolume values of 100 runs for a statistical comparison of different operators of AMOSA and NSGA-II algorithms for N20 Dataset	48
Figure 6-13 Convergence plot generated by the hypervolume values versus generation counts.....	50
Figure 6-14 Box-plot of hypervolume values for a statistical comparison of different operators of AMOSA and NSGA-II algorithms.....	51
Figure 6-15 Box-plot of hypervolume values for a statistical comparison of different parameter settings for	56
Figure 6-16 The first Pareto optimal solution that represents a hub network with all hub-to-hub and hub-to-spoke links obtained by our approach.....	57
Figure 6-17 The second Pareto optimal solution that represents a hub network with all hub-to-hub and hub-to-spoke links obtained by our approach	58
Figure 6-18 Resulting pareto optimal solution of AMOSA algorithm with Random K neighborhood having the hypervolume value in Table 6-4.....	59

LIST OF TABLES

Table 2-1 Comparison of the literature on multi-objective HLP with the model in this thesis	4
Table 4-1 Variable notations.....	14
Table 6-1 Execution times of brute-force method and our heuristic for different network sizes	45
Table 6-2 Average HyperVolume values of 200 runs for 20-Node dataset at certain number of evaluations.....	47
Table 6-3 Statistical comparison of test results obtained in N20 dataset. s^+ indicates algorithm on row section is statistically better than algorithm at column section. s^- indicates statistically worse. + sign indicates algorithm at row is better than the one at column. – sign indicates the opposite of + ...	49
Table 6-4 Average HyperVolume values of 20 runs for Turkish dataset (Number of evaluation= 20.000	51
Table 6-5 Statistical comparison of test results obtained in original Turkish dataset. s^+ indicates algorithm on row section is statistically better than algorithm at column section. s^- indicates statistically worse. + sign indicates algorithm at row is better than the one at column. – sign indicates the opposite of + ...	52
Table 6-6 Simple Mutation – arc mutation ratio parameter tuning (Number of evaluation= 20.000	53
Table 6-7 Variable notations.....	53
Table 6-8 Random K Neighborhood – number of iterations per temperature parameter tuning by stabilizing α parameter to keep maximum number of evaluations 20.000 by means of average hypervolume and maximum hypervolume	54
Table 6-9 Hypervolume results and standard deviation for our approach with various population sizes popsize and fixed parameter settings ($p_c = 0.9, p_{bfm} = \frac{1}{n}, p_{sm} = 0.01$)	55

Table 6-10 Hypervolume results and standard deviation for our approach with various crossover probabilities p_c and fixed parameter settings	55
Table 6-11 Hypervolume results and standard deviation for our approach with various p_{bfm} and p_{sm} and fixed values of $popsiz$ = 200 and p_c = 0.8.....	57
Table 6-12 Routing scenarios for some source-destination pairs in the given Pareto optimal solutions.....	60



1. INTRODUCTION

There are plenty of applications that need transfer of people, commodities and data between source-destination pair of nodes, such as, air transportation services, public postal services, cargo delivery systems, and telecommunication systems. The networks for these applications are usually designed based on hub-and-spoke networks. In these networks, hubs are established so as to collect and distribute the flow between source and destination pairs, and spokes are the non-hub nodes that are allocated to hub nodes in the network. Hub location problem (HLP), in general, deals with the decision of hub and spoke nodes in a given network, as well as connections between hubs and spokes, i.e. the assignment of spokes to hubs, and the routing of flow between each pair of nodes.

Several variations of HLP have been studied in the literature. The problem can be modeled as single allocation HLP [1] or multiple allocation HLP [4] considering the assignment of spokes to hubs in the network. Multiple allocation HLP allows spokes to be assigned to more than one hub, whereas, in single allocation spokes can be assigned to only one of the hubs. If capacities of hubs are considered, the HLP can be capacitated [5] or uncapacitated [8]. Also, in a HLP, the number of hubs to be established can be predetermined as in p-HLP [4] or determining the number of hubs can be among the decisions that should be made. Most of the studies on HLP consider that the hubs are fully connected to each other. However, in real world applications such as less-than-truckload networks and telecommunications networks [10], it is not practical to establish a connection between each hub pair. The design of connections between hubs is referred as hub network design [11] and considered within different applications in the literature.

In this thesis, a Multi-Objective Capacitated Multiple Allocation Hub Location Problem (MOCMAHLP) is addressed. The number of hubs is not known in advance and the hub network is not assumed to be fully interconnected. To the best of our knowledge, this is the first attempt that gives a mathematical model for this variation of HLP. To contribute to fill this gap, we propose a mathematical model for the MOCMAHLP with objectives of minimizing transportation cost and fixed costs and minimizing the maximum travel time required for routing the commodities between all node pairs. In most of the HLPs in the literature, the first objective, i.e., the minimization of transportation cost, is used. However,

in real world applications such as systems that transfer perishable goods, rapid transportation and delivery is required, which motivates the second objective. The routes between each source-destination pair should be determined in order to calculate the second objective. Therefore, our proposed mathematical model includes routing of commodities through network, besides determining the location of hubs and the hub-to-hub and spoke-to-hub arcs in the whole network.

The HLP belongs to the class of NP-complete problems [12]. The computational results show that the MOCMAHLP can be solved optimally for specifically small size networks. To solve the problem for a 7-node network using brute-force method that considers every possible solution, we need approximately 3 hours, and if we increase the number of nodes to 8, the required time is more than a day. Therefore, to solve the MOCMAHLP for real size networks, we propose two meta-heuristics based on non-dominated sorting genetic algorithm (NSGA-II [13]) and based on simulated annealing algorithm(AMOSA[35]). NSGA-II algorithm, namely non dominating sorting genetic algorithm, uses elitism principle to select parent solutions and fast sorting algorithm working with complexity of $O(MN^2)$ where M is the number of objectives and N is the population size while many other multi-objective evolutionary algorithms have the complexity of $O(MN^3)$. In NSGA-II algorithm, population is first getting sorted by pareto sets and crowding distance, i.e. starting from best pareto front's smallest crowding distance to firstly biggest crowding distance of the regarding pareto front, lastly the worst pareto front's biggest crowding distance valued solution is paired. Then new offspring is generated by mating the sorted solutions on crowded tournament selection. AMOSA algorithm, namely a simulated annealing based multi-objective optimization algorithm, incorporates the concept of an archive, where the non-dominated solutions seen so far are stored. The algorithm uses delta dominance and current temperature values to select perturbed solution as current solution to continue further iterations. For both algorithms, a novel solution representation and proper genetic operators are designed as part of our solutions. We conducted a set of experiments using the Turkish postal system data set and reduced AP data set [36] considering hyper volume as comparison metric. The results of our experimental study demonstrates that our algorithm can find feasible solutions for a relatively large network containing 81 nodes in less than 10 minutes.

The rest of this thesis is organized as follows. In the next section, we give a review of the related literature that emphasizes similarities, differences and deficiencies of the models presented. In Section 3, meta-heuristic algorithms for the HLP problem and the reason why we have chosen NSGA-II and AMOSA algorithms to solve MOCMAHLP are presented. In Section 4, the proposed mathematical model for the MOCMAHLP is given. In Section 5, we present our NSGA-II and AMOSA approach for the MOCMAHLP. The solution representation and the genetic operators are explained in detail. Section 6 includes our experimental setup and a discussion about computational results. Finally, the last section highlights the conclusion and future work.



2. PROBLEM STATEMENT AND RELATED WORK

In this thesis, we concentrate on multi-objective, capacitated, multiple allocation HLP; and in our mathematical formulation of the given problem, we consider determining the location of hubs, spoke-to-hub and hub-to-hub allocations, and routing of commodities between each pair of nodes. Although most of the models for HLP include only one objective, real-world applications of HLP require more than one objective. In Table 2-1, a complete comparison of models presented in the literature for multi-objective HLP and the model presented in this thesis is given. To the best of our knowledge, there is no study that models this variation of HLP.

Table 2-2 Comparison of the literature on multi-objective HLP with the model in this thesis

Model in literature	Variable Number of Hubs	Capacitated	Multiple Allocation	Hub Network Design
Mirzaei and Bashiri [17]	✓	✓	✓	—
Tavakoli et al. [18]	✓	✓	✓	—
Köksalan and Soylu [19]	—	—	✓	—
Köksalan and Soylu [19]	—	✓	✓	—
Costa et al. [20]	✓	✓	—	—
Ghodratnama et al. [21]	✓	✓	—	—
Kaveh et al. [22]	—	✓	—	✓
Karimi and Setak [23]	✓	—	✓	✓
Our model	✓	✓	✓	✓

In a recent study [17], Mirzaei and Bashiri present a model for multi-objective multiple allocation hub location problem. However, in their model, the hub network is assumed to be fully interconnected and no experimental result is given in the corresponding paper. The multi-objective multiple allocation hub location problem without hub network design is solved also in [18] considering sustainable development paradigm and using multi-objective differential algorithm. Köksalan and Soylu [19], present two models for multiple allocation HLP. Both models assume the fixed number of hubs and fully-interconnected hub networks. In [20]–[22], single allocation HLP is considered. Karimi and Setak [23]

consider the problem similar to our assumptions except the capacities of hubs and arcs in the network.

The common objective used in most of the studies given in Table 2-1 is total cost minimization. Besides this, other objectives can be listed as the following:

- Minimization of maximum travel time [17]
- Minimization of environmental impacts like energy cost, noise, congestion, pollution, greenhouse gas emission [18]
- Minimization of total transportation costs [19], [20], [24]
- Minimization of the maximum delay at each hub [19]
- Minimization of total time for processing the flow collected by the hubs [20]
- Minimization of the sum of waiting times [24]
- Minimization of total gas emissions [20], [24]
- Maximization of benefits of transportation and establishment of the hub facilities [22]
- Minimization of the total transportation time [22]
- Maximization of the total amount of flow delivered in a predefined time [23]

Costa et al. [20] present two different mathematical models for a bi-objective capacitated single allocation HLP. In the first model, it is aimed to minimize the total costs and to minimize the total time for processing the flow collected by the hubs. In the second model, the objectives are to minimize the total costs and to minimize the maximum service time on the hubs. Two bi-criteria uncapacitated, multiple allocation p-HLPs are studied in [19]. The objectives in one of their problems are minimizing the total transportation costs and total traveling costs between hubs and source-destination points. In the other problem, the objectives are minimizing the total transportation costs and minimizing the maximum delay at each hub. Mirzaei and Bashiri [17] consider a multi-objective approach for HLP to minimize the total cost and to minimize the maximum traveling time between nodes in the network. Ghodrathnama et al. [21] formulate a capacitated single allocation HLP with three objectives which are minimization of total costs including the transportation and installation, minimizing the sum of service time in the hubs, and minimization of total

greenhouse emission. Karimi and Setak [23] develop a flow shipment scheduling and incomplete hub location-routing problem using two conflicting objectives which are to maximize the total amount of flow delivered in a predefined time and to minimize the total costs. A hub location-allocation model considering congestion is presented in [24], with the aim of minimizing the total costs and minimizing the sum of waiting times. Tavakoli et al. [18] introduce a multi-objective mathematical model for multiple allocation HLP with multiple capacity levels. The problem minimizes both costs and environmental impacts. Since HLP is NP-complete [12], linear programming can only find optimum solutions for small-scale problems, however, heuristics are used to find optimal or near optimal solutions for large-scale problems in the domain of HLP [14], [25]. In addition, meta-heuristics such as genetic algorithm [9], multi-objective differential evolution [18] and multi-objective evolutionary algorithm [19], [26]–[28] are successfully applied to HLPs. In addition, the non-dominated sorting genetic algorithm (NSGA-II) has been successfully applied to different multi-objective facility location problems [26], [29], [30] as well as HLPs [27], [28]. Therefore, in this thesis, we propose two heuristics; one based on NSGA-II and the other based on AMOSA for the MOCMAHLP.

2.1. MOCMAHLP NETWORK AND ROUTING

To explain the evolution of MOCMAHLP problem, it is necessary to start from the very basic version of the problem, namely P-Hub location problem. In P-Hub location problem, number of hubs is preliminary selected and static as parameter P. Number of possible networks is given in the Equation (2.1). An example P-Hub network with having 5 hubs as P is given in Figure 2-1.

$$\binom{N}{P} \quad (2.1)$$

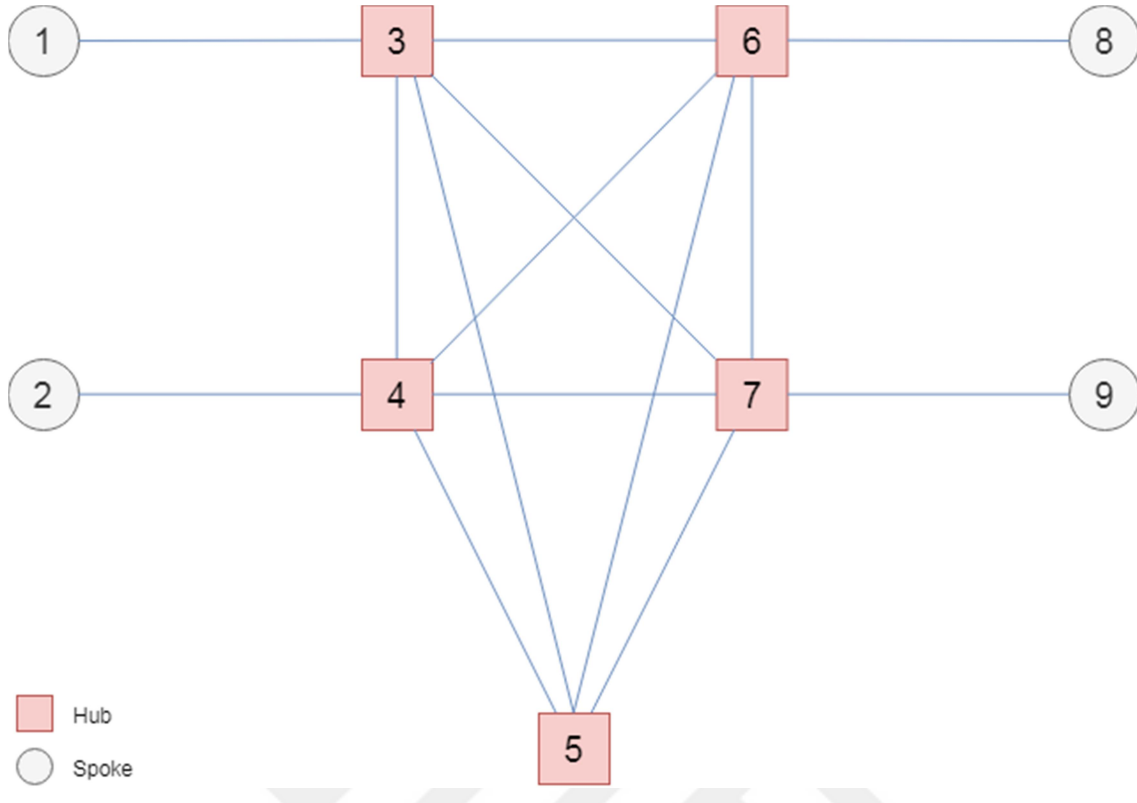


Figure 2-1 An example of P-Hub Location Problem while $P = 5$ for a 9-node network, with nodes 3,4,5,6 and 7 selected as hubs

If the static parameter P is removed from the problem, then problem becomes a hub location problem where the number of hubs can be changed to form the network. In this case, number of possible networks is given in the Equation (2.2).

$$2^N - 1 = \sum_{i=1}^N \binom{N}{i} \quad (2.2)$$

When multi allocation restriction is applied on the sample network of hub location problem given in Figure 2-1, we can have such networks that spokes are not necessarily connected to only one hub, but can be connected to multiple hubs. In the sample network which is given in Figure 2-2, node 9 and node 1 is connected with hubs 7 and 5 and hubs 3 and 4 respectively. Thereby every one of $N - i$ number of spokes can be connected up to i hubs which have $(2^i - 1)$ number of combinations. Adding this restriction to the problem increases number of possible networks given in Equation (2.2) to Equation (2.3).

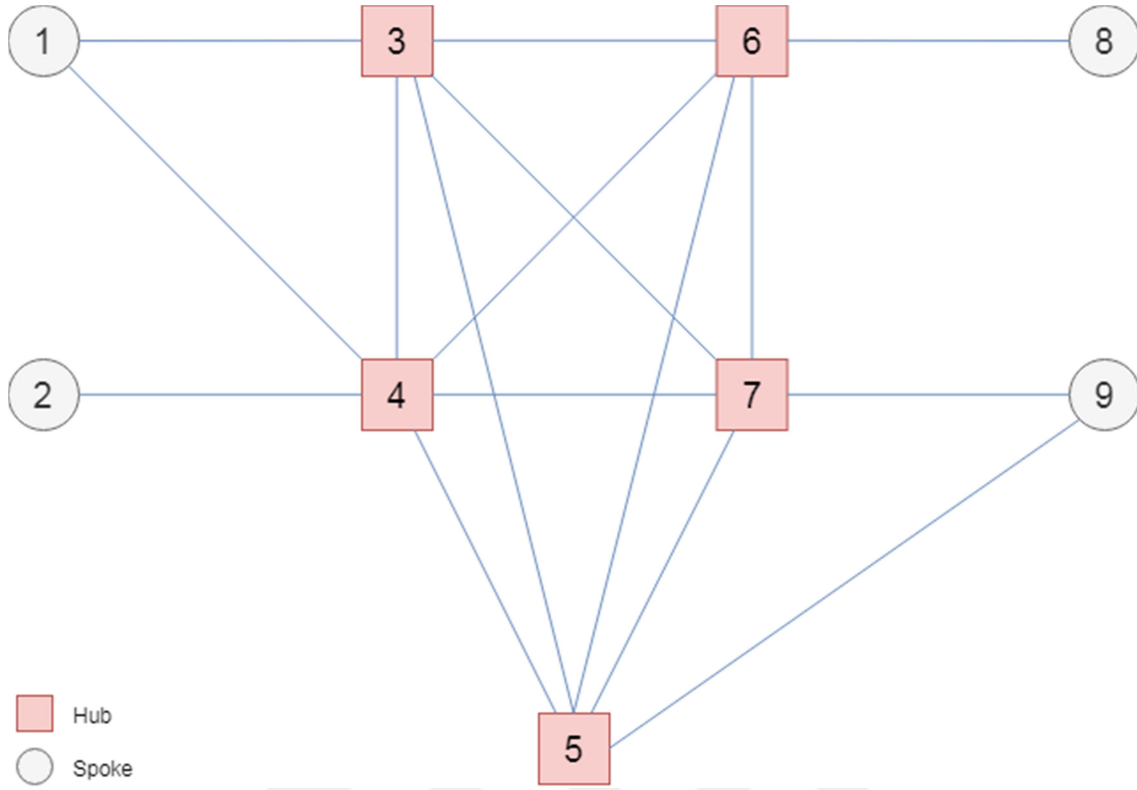


Figure 2-2 An example multi allocation hub location problem with 9 nodes

$$\sum_{i=1}^N \binom{N}{i} (2^i - 1)^{N-i} \quad (2.3)$$

On top of multi allocation hub location problem, if the restriction of fully interconnection is no more required is added, then it brings new possible networks to our problem. For every hub within i number of selected hubs, can be connected $2^{i-1} - 1$ different way to each other considering having i hubs as given in the Equation (2.4). A sample network for not interconnected multi allocation hub location problem is given in Figure 2-3.

$$\sum_{i=1}^N \binom{N}{i} (2^i - 1)^{N-i} (2^{i-1} - 1)^i \quad (2.4)$$

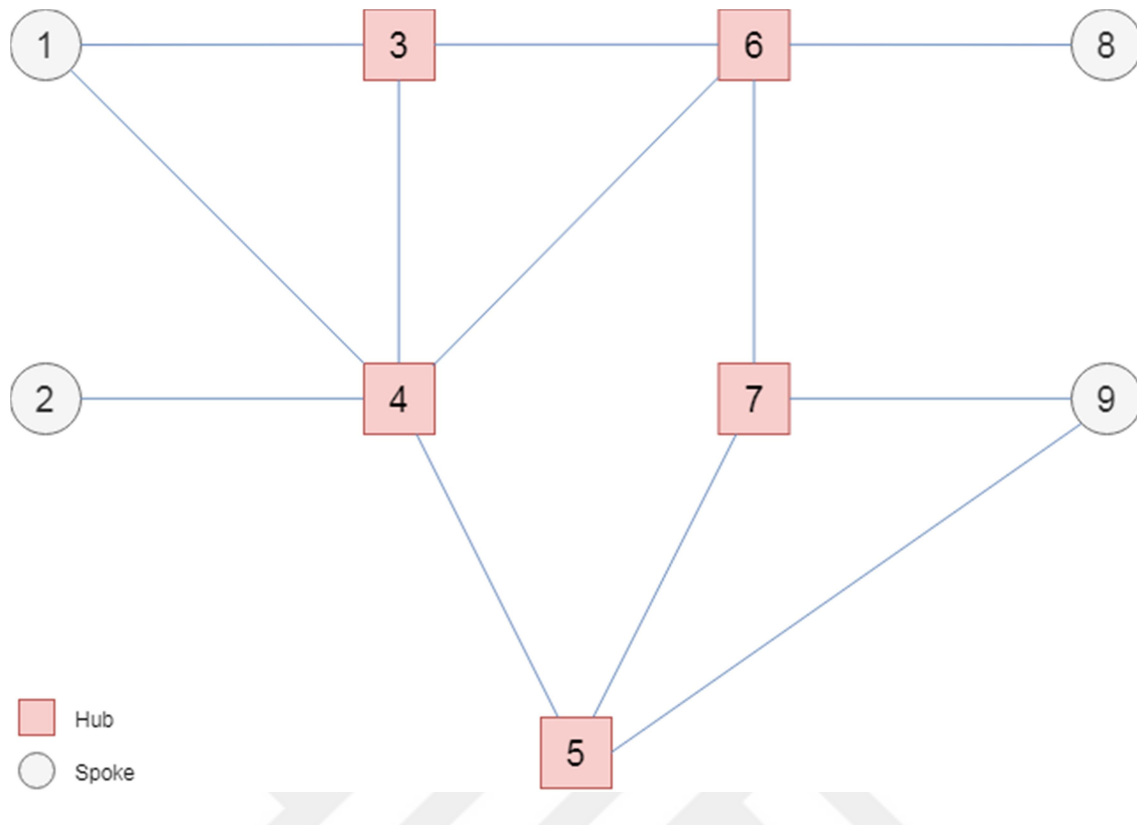


Figure 2-3 A sample network for not interconnected multi allocation hub location problem

Supposing the shortest path of commodity whose source node is 9 and destination node 1 is shown in the Figure 2-4 with green marker. Then we should send all the demand over this route. However, if capacity restrictions on both arcs and hubs are introduced, then a routing strategy is required to follow since after routing of certain amount of demands are allocated on arcs and through hubs may be fulfill the capacity of an arc or a hub. An alternative route to the shortest path on the sample network is shown in Figure 2-4 by blue marker. Another important decision is to distribute the total demand if there are more than one route for the same commodity. The strategy we used for this situation is: whenever capacity is not enough to carry the whole demand of next commodity, then we reserve all possible demand that can be carried through shortest path and send the first package over this route. Then for the remaining demand of the commodity, next shortest path is found and send the remaining amount as second package. If second package also hits the capacity of route, same strategy is used until there is no more available paths which has capacity. In this case

we denote that this network is infeasible and use the number of total demands which could not be transferred in this network as constraint in the problem.

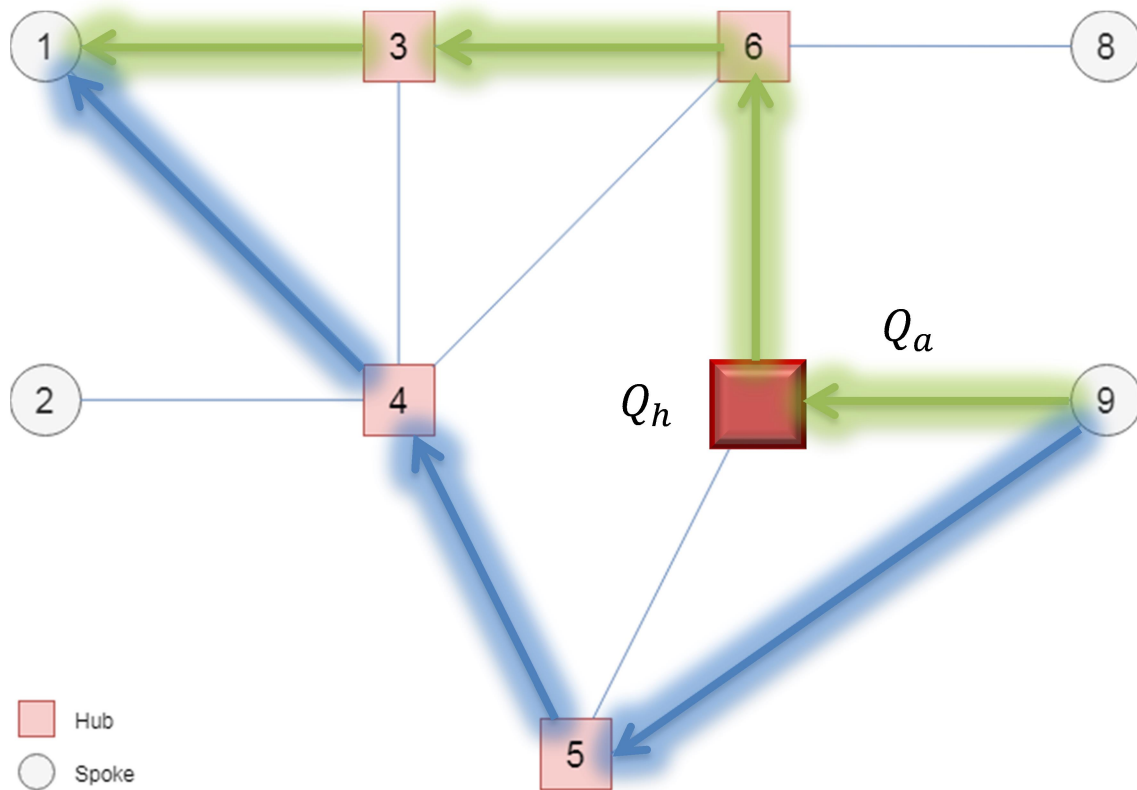


Figure 2-4 A Sample routing for capacitated multi allocation not interconnected hub location problem

3. METAHEURISTICS FOR MULTI-OBJECTIVE HLP

Most commonly used algorithms for multi-objective hub location problems are NSGA-II[13], Archive multi-objective simulated annealing (AMOSA) [35], and multiple artificial bee colony (MOABC) [28]. NSGA-II is an improved version of NSGA, which is a fast sorting and elite multi-objective genetic algorithm. In NSGA-II, the mechanics of general genetic algorithms (i.e. mutation, cross-over) are used. Archive multi-objective simulated annealing (AMOSA) incorporates the concept of an archive, where the non-dominated solutions seen so far are stored. Last of all, multiple artificial bee colony (MOABC) [37], is inspired from foraging behavior of honey bees and population based algorithm which has three kinds of bees; employed bees, onlooker bees, scout bees; each of which are specialized with a mission such as exploitation and exploring.

In this thesis, we have selected NSGA-II and AMOSA algorithms as multi-objective optimization algorithms due to commonly usage in the literature. The provided algorithm is open for application of other multi-objective algorithms for improvement and comparison purposes as indicated in future works section.

3.1. NSGA-II

The general nature of NSGA-II algorithm provides integration into any multi-objective problem by overriding crossover and mutation functions specifically to problem itself. Also the algorithm contains an internal fast sorting algorithm and naturally has both exploration and exploitation. Because of these reasons, it is selected as population based algorithm and applied into many multi-objective hub location problems.

A. Ebrahimi Zade et al.[28] used a modification of NSGA-II algorithm in their bi-objective hub maximal covering problem under uncertain shipments. They added a new dynamic immigration operator into original algorithm and compared the results with the original NSGA-II algorithm. F. Niakan et al.[38] studied hub network design under uncertainty and they developed multi-objective invasive weed optimization (MOIWO) for the problem. Finally they compared the results of MOIWO with NSGA-II and multi-objective simulated annealing (MOSA) algorithm. M. Musavi et al.[39] studied a novel multi-objective sustainable hub location-scheduling problem model by using CAB and AP datasets and used NSGA-II algorithm to solve the presented problem.

3.2. AMOSA

Improved version of multi-objective simulated annealing algorithm by adding archive provides another easy to integrate algorithm by only overriding perturb function. The algorithm is selected as single point search algorithm by overriding perturb function and applied into many multi-objective hub location problems.

F.Parvaresh et al.[40] studied hub network design problem in the presence of disruptions and solved the problem with using two multi-objective meta-heuristic algorithm; one based on tabu search and the other is simulated annealing based SMOSA algorithm. As also mentioned in section 3.1. Niakan et al.[38] used MOSA algorithm to compare with their solution.

4. MATHEMATICAL MODEL

In this section, we present a new model for the Multi-objective Capacitated Multiple Allocation Hub Location Problem (MOCMAHLP). The basic assumptions of the problem are as follows:

- The solution space is discrete and finite
- The set of nodes is known
- The number of hubs is not fixed
- Each node has a fixed cost to be established as a hub
- Each arc in the established network has a fixed cost
- Each node can be assigned to more than one hub
- There is no direct link between spokes, i.e. two spokes can only communicate via at least one hub
- Hub network is not fully interconnected
- Each arc in the network has a flow capacity
- Each hub has a limited capacity

In the MOCMAHLP, there are three decisions to be made: (1) hub location, (2) the hub network design and (3) the routing of the flow. The first decision is to determine the number and location of hubs. The second involves decision on the allocation of spokes to hubs and then the determination of hubs that are interconnected. We consider the multiple allocation pattern, i.e. a spoke can be assigned to more than one hub. The last one is the decision on how to route the flow. Since the problem has both hub and arc capacities, the demand may be routed through separate paths going from its source and destination nodes. Therefore, the demand is divided into separate packages with different amount of flow.

Under these assumptions, we develop a mathematical model to the MOCMAHLP based on the formulation of Capacitated Multiple Allocation Hub Location Problem (CMAHLP) proposed by Rodríguez-Martín and Juan José Salazar-González [6]. To the best of our knowledge, this is the first attempt to model this variation of HLP with the specified assumptions.

First, we introduce the notation used in the model. Given a directed graph $G = (V, E)$, where V is a set of nodes and E is a set of directed edges (or arcs), each arc ($a \in E$) is associated with an ordered pair of nodes ($a = (u, v), u, v \in V$). The demand from node i to node j (d_{ij}) is called commodity and will be denoted by a single index k where $k = (i, j)$. The amount of package p of commodity k is denoted as k_p . We additionally use the following notations:

Table 4-1 Variable notations

N	Number of nodes ($N = V $)
H	Set of possible hub locations
a_u	The initial node of arc $a \in E$
a_v	The terminal node of arc $a \in E$
T_a	Time required to transfer over arc $a \in E$
F_a	Total amount of flow traversing arc $a \in E$
C_a	Cost of delivery on arc $a \in E$
λ_a	Fixed cost of arc $a \in E$
Q_a	Capacity of arc $a \in E$
i_k	The source node of commodity k
j_k	The destination node of commodity k
D_k	Total demand of flow for commodity k
D_{k_p}	Demand of flow for package p of commodity k
f_a^k	The flow of commodity k on arc $a \in E$
κ_h	Fixed cost of hub $h \in H$
Q_h	Capacity of hub $h \in H$
χ	Collection discount factor
α	Discount factor of hub to hub transportation
δ	Distribution discount factor

The decision variables for the MOCMAHLP are given as follows:

$$Z_a = \begin{cases} 1, & \text{if arc } a \text{ is used between nodes } a_i \text{ and } a_j \\ 0, & \text{otherwise} \end{cases} \quad (4.1)$$

$$x_i = \begin{cases} 1, & \text{if node } i \text{ is hub} \\ 0, & \text{otherwise} \end{cases} \quad (4.2)$$

$$x'_i = \begin{cases} 1, & \text{if node } i \text{ is spoke} \\ 0, & \text{otherwise} \end{cases} \quad (4.3)$$

$$A_{k_p}^a = \begin{cases} 1, & \text{if the flow at package } p \text{ of commodity } k \\ & \text{is routed via arc } a \text{ with } Z_a = 1 \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

$$D_{k_p} = n, \text{ such that } 0 \leq n \leq D_k \quad (4.5)$$

$$F_a = \sum_{k_p} D_{k_p} A_{k_p}^a \quad (4.6)$$

$$f_a^k = \sum_p D_{k_p} A_{k_p}^a \quad (4.7)$$

The MOCMAHLP can be formulated as follows:

$$\text{minimize } \sum_a F_a C_a (\chi x'_{a_u} x_{a_v} + \alpha x_{a_u} x_{a_v} + \delta x_{a_u} x'_{a_v}) + \sum_a \lambda_a Z_a + \sum_i^N \kappa_i x_i \quad (4.8)$$

$$\text{minimize } \max_{k_p} \sum_a T_a A_{k_p}^a \quad (4.9)$$

subject to

$$1 \leq \sum_i x_i \leq N \quad (4.20)$$

$$F_a \leq Q_a \quad \forall a \in E \quad (4.11)$$

$$\sum_{a \in V^+(h)} F_a \leq Q_h \quad \forall h \in H \quad (4.12)$$

$$\sum_p D_{kp} = D_k \quad \forall k \quad (4.13)$$

$$A_{kp}^a \leq x_{av} Q_h \quad a_v \neq j_k \quad \forall k \quad (4.14)$$

$$0 \leq f_a^k \leq F_a \quad \forall a \in E \quad (4.15)$$

for each commodity k

$$\sum_{a \in V^+(v)} f_a^k - \sum_{a \in V^-(v)} f_a^k = \begin{cases} D_k, & \text{if } v = i_k \\ -D_k, & \text{if } v = j_k \\ 0, & \text{otherwise} \end{cases} \quad (4.16)$$

The first objective is to minimize the total cost including the cost of collection, transfer and distribution (the first part of the objective function), the fixed arc cost (the second part) and the fixed hub cost (the last part). The second objective is to minimize the maximum travel time required for routing between all node pairs. Constraint (4.10) ensures that there are at least one hub. Constraint (4.11) and Constraint (4.12) are arc and hub capacities, respectively. Constraint (4.13) states that the sum of flow demand for all packages of a commodity is equal to the total flow demand for the commodity. Constraint (4.14) ensures that arc a can be selected in the routing if and only if destination node of arc a is hub and hub has capacity, for every commodity k and package p of the commodity excluding the last node. Constraint (4.15) is total flow of commodities. Finally, constraint (4.16) ensures

that for all commodities, sum of all incoming flows minus outgoing flows has to be 0 except arc a is connected to source or destination node of commodity k . In this case, the total flow has to be D_k in case arc a is connected to source node of commodity and has to be $-D_k$ otherwise.



5. METAHEURISTIC APPROACHES

The multi-objective optimization problems are problems that consist of more than one objective function that must be simultaneously optimized. A key goal in multi-objective optimization research is to find a set of optimal solutions known as Pareto optimal set instead of a single optimal solution. A set of points mapped from the Pareto optimal set is referred to as the Pareto-front.

Evolutionary algorithms have been successfully applied to the multi-objective optimization problems. The non-dominated sorting genetic algorithm (NSGA-II) [13] is one of the most well-known multi-objective evolutionary algorithms. NSGA-II uses genetic operators including binary tournament selection, crossover and mutation to generate offspring population. The individuals for the next generation are selected according to non-dominated and crowding distance sorting procedures.

Simulated annealing algorithms have also been applied to the multi-objective optimization problems. Archive multi-objective simulated annealing algorithm (AMOSA)[35] is one of the most well-known variation of the base algorithm. The algorithm is an improved version of multi-objective simulated annealing (MOSA) which is the multi-objective version of the simulated annealing (SA) algorithm.

In this study, since the problem is NP-complete, we propose a heuristic approach based on NSGA-II and AMOSA. In this section, the details of our algorithm are presented. The pseudo-code of the NSGA-II algorithm is given in Algorithm 1 and the pseudo-code of the AMOSA algorithm is given in Algorithm 2. We propose a new representation for the candidate solution and crossover, mutation and perturb operators are adapted to the solution representation. Besides, a repair function for the infeasible solutions is introduced.

We have developed various operator functions for both algorithms and compared the results. Each operator functions have either different mechanics or using the same mechanics but using different procedure within the same mechanics. Even though the problem has 3 different decision data structures, so called; hub decision, link decision and route decision, only first 2 of them was able to be applied crossover and mutation. The reason behind this behavior is, route decision is calculated by using linear approach developed using A* algorithm and this decision is seriously changed by a little change in

the other decisions. So we have two common operators; one is for hub decision and the other is for link decision. After applying all this operators into the decision sets, routing decision is calculated from scratch for every new child.

5.1. SOLUTION REPRESENTATION

Each candidate solution is represented using two same length arrays. The length of both arrays is the same and equal to the number of nodes in the network. One of the arrays is a binary array, which is used for the hub decision, i.e. value 1 states that the corresponding node is a hub and value 0 states that it is a spoke. An example of hub decision array is illustrated in Figure 5-1. This solution includes five hubs out of nine nodes.



Figure 5-6 An example of hub decision array for a 9-node network

The second array, named link decision array, is just an adjacency list representation of the network design. Each element in the array contains the nodes that are connected to the corresponding node. An example of link decision array that can be generated for hub decision array in Figure 5-1 is illustrated in Figure 5-2. In this example, node 3, which is a hub, is connected to nodes 1, 4 and 6. Also, node 9, which is a spoke, is connected to nodes 5 and 7. According to the solution representation given in Figure 5-1 and Figure 5-2, the resulting network design can be seen in Figure 5-3.

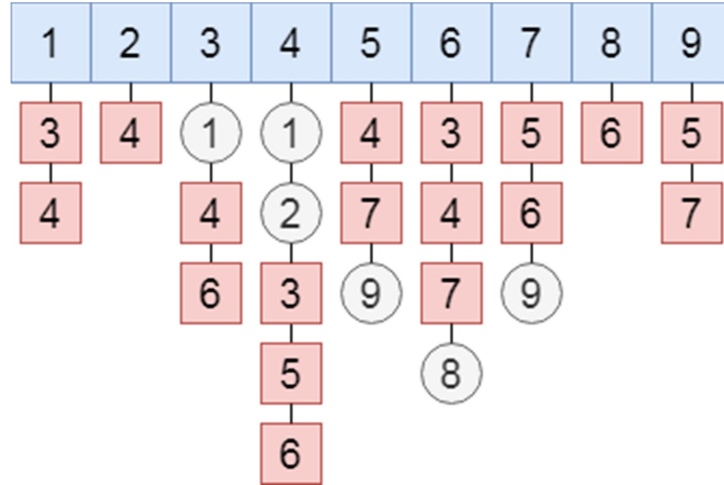


Figure 5-7 An example of link decision array for a 9-node network, with nodes 3,4,5,6 and 7 being hubs

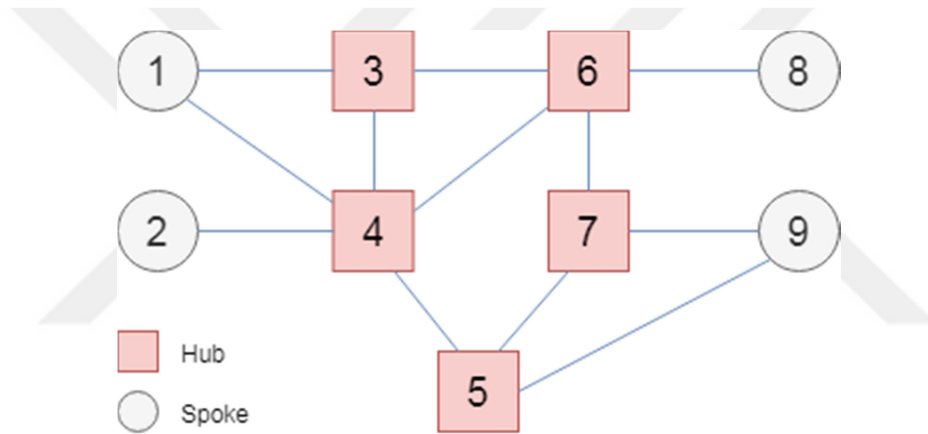


Figure 5-8 The illustration of resulting network for the hub decision and link decision arrays in Figure 5-1 and Figure 5-2

5.2. REPAIR FUNCTION

A repair function is applied to all the infeasible solutions generated as a result of genetic operators. Since the operators are applied to both arrays, a repair phase is also needed for both. In the repair phase for the hub decision array, only the number of hubs in the offspring is taken into account. If the offspring does not include a hub, this offspring is discarded and a random offspring is generated instead. The repair function used for the link decision array considers the five cases listed:

1) A link between two nodes may not be mutually included in the link decision array. In this case, the link is mutually established.

- 2) A hub has no hub connection. In this case, the hub becomes connected to the closest hub.
- 3) There is a connection between two spoke nodes. In this case, the connection is removed.
- 4) A spoke node is not assigned to a hub. In this case, it is assigned to the closest hub.
- 5) The resulting network may be disconnected. In this case, the closest hub nodes of each sub-network become connected.

5.3. ROUTING

To obtain the total amount of flow traversing each arc, the routing of demands for all the commodities must be determined (line 5 of Algorithm 1). Due to the hub and link capacities, the demand of each commodity may be divided into separate packages. Additionally, the packages may be routed through several paths since the multiple allocation pattern and the interconnected hub network are considered.

For routing, we implemented A* algorithm. A* algorithm iteratively check the neighbor nodes for the sum of cost to reach from source to neighbor node and approximate cost to reach from neighbor node to destination node. Meanwhile the algorithm keeps track of visited nodes. If a visited node can be reached later with lower cost than before, than it is considered as the node is not visited. The memory and time complexity of A* algorithm used in the problem is effectively $O(N)$.

Due to capacity constraint, the order of routing allocations are important since some arcs or hubs can be fulfilled just because of staying in the shortest path of some commodity while for some of the next commodities, usage of such hubs or arcs are crucial. To overcome this problem, we calculate routing of commodities in random order.

A simple scenario for routing is given in Figure 5-4. As seen in Figure 5-4, the commodities from node 1 to node 2 and from node 2 to node 8 are not divided into packages and routed through the paths $1 \rightarrow 4 \rightarrow 2$ and $2 \rightarrow 4 \rightarrow 6 \rightarrow 8$, respectively. On the other hand, the commodity from node 1 to node 9 is split into two packages. The first package, which contains three-eighth of the demand, follows the path going through two hubs (node 4 and 5). The remaining part is sent along the path: $1 \rightarrow 3 \rightarrow 6 \rightarrow 7 \rightarrow 9$.

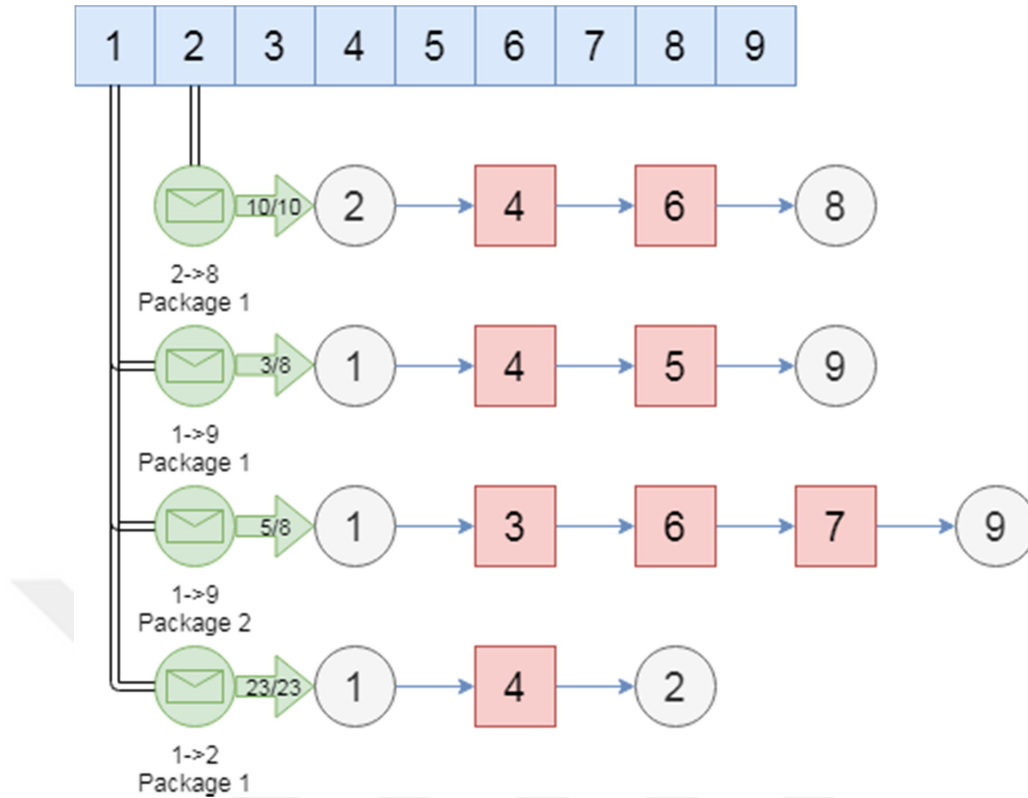


Figure 5-9 Routing example of three source-destination pairs for the given 9-node network in Figure 5-3

A* algorithm which aims to find the shortest path in a graph is employed for the routing of all the commodities. First, A* algorithm searches for an augmenting path in the residual network to route the demand of a commodity, then, if there is such a path, a portion of the flow demand with the residual capacity on this path is routed through the selected path. The residual capacity of a path is equal to the minimum available capacity considering the hub and link capacities along the path. After routing, the residual network is updated with the remaining capacities for each link and hub. If all the demand for the current commodity is not routed, a new path is examined for the remaining demand. This process is repeated for each commodity.

During the process of routing, if there are commodities for which all the flow demand cannot be routed, this solution is regarded as infeasible. The total number of such commodities is specified as constraint violation for the corresponding solution. The constraint violation is taken into account for the dominance.

5.4. NSGA-II BASED APPROACH

Considering the nature of the genetic algorithms, the main mechanics are developed on randomization. Since we have 2 decision variables to apply mutation, all variations of mutation functions take hub mutation probability and link mutation probability in common. The general pseudo-code is given in Algorithm 1.

Algorithm 5-1 The pseudo-code of the proposed NSGA-II approach

```
1: Create an initial population ( $P_0$ ) and evaluate  $P_0$ 
2: Sort the population based on the non-domination
3: while (maximum number of iterations is not reached) do
4:   Generate child population ( $Q_t$ )
5:   Apply A* algorithm for routing
6:   Evaluate the child population  $Q_t$ 
7:   Merge child and parent populations ( $R_t = P_t \cup Q_t$ )
8:   Select the individuals for the next generation
9: end while
```

5.4.1. CROSSOVER OPERATOR

One-point crossover operator is applied to both hub and link decision arrays with a predefined probability. The same crossover point which is randomly selected is considered for both arrays as illustrated in Figure 5-5. Cross over is applied only for NSGA-II algorithm which supports more than one value of arity in solutions.

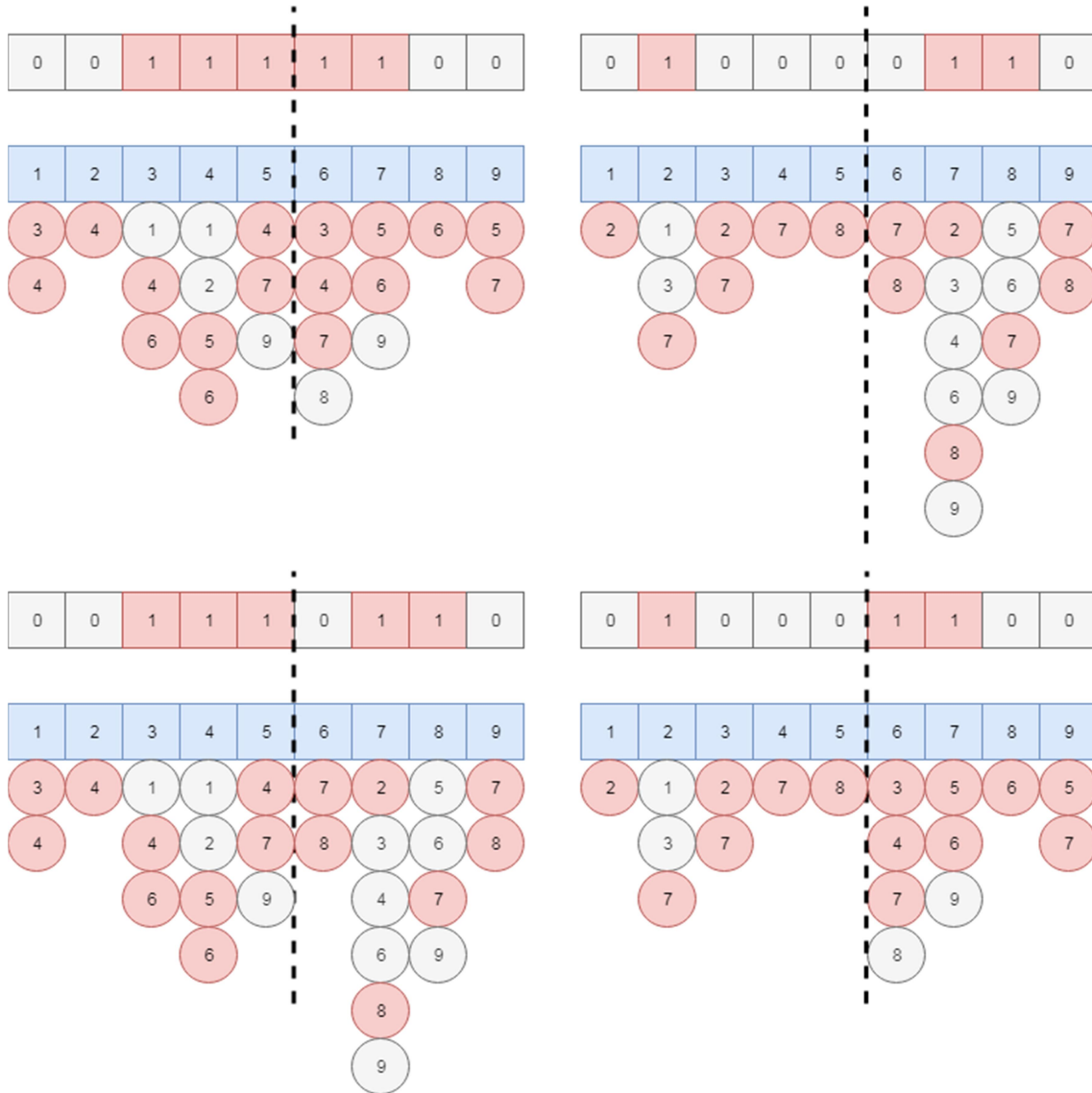


Figure 5-10 Crossover operation on hub decision and link decision. Figures at top illustrate parents and figures at bottom illustrate resulted offspring.

5.4.2. MUTATION OPERATORS

5.4.2.1. ARC MUTATIONS

SWAP ALL ARCS MUTATION

First and most basic of all; swapping all arcs is conducted by switching every single node from hub to spoke and spoke to hub with the given probability in hub decision variable, and selecting one node with the given probability to be swapped the links. Then if the chosen node is hub, select another hub and exchange all the links they have. If the chosen node is

spoke, then select another spoke and exchange all the links they have. In case, there is no hub or spoke to swap the links, i.e. chosen node is hub and network already have had 1 hub only, then skip this swap and continue.

CENTRALITY BASED MUTATION

The idea of centrality based mutation arisen from the need of faster convergence. So that we have added tournament selection for each mutations. Thereby an additional logic for switch hub mutation developed; when a node is selected with the given probability to be switched from hub to spoke, then if exists; we are selecting another hub randomly and comparing their centrality. To switch from hub to node, the hub which is less central than the other is selected to be switched. The same mechanism is applied for spokes; when a spoke is selected with a given probability to be switched to hub, then if exists, another spoke is chosen and the spoke which is more central is selected to be switched to hub. A similar mechanics is applied to link mutation, when a node is selected to be mutated, first it is decided to add or remove a link. Adding or removing is decided with 0.5 probabilities except some conditions. If the decision is to add a link and these condition occur then the decision will be changed to removing; if the chosen node is a spoke, the spoke has to have less links than number of hubs, if it is a hub, the hub has to not to have connected to every nodes in the network. If the decision is to remove a link and this condition occur then the decision will be changed to addition; if the chosen node has no links to be removed. If the adding decision is made and if the chosen node is spoke, than, one of the available hubs which is not connected to this spoke is chosen as hub candidate-A and another hub is chosen as hub candidate-B if available. Finally, between the candidate hubs, one is selected which has less distance value to chosen spoke. If there is only one hub candidate, then no tournament is applied. If the adding decision is made and if the chosen nose is hub, than, the same procedure is applied by choosing candidates between all nodes set instead of hubs set. If the removing decision is made, candidates are getting applied to a tournament and the arc candidate which has bigger distance value is selected as to be removed.

Centrality value is calculated for every node in the network by summing the normalized average demand to/from the node and the normalized average distance to other nodes. The centrality calculation is made the same way for every dataset used while the data set is

loaded from file. The range of centrality value is: $[0, 2]$ in which high values correspond to better centrality.

ONE LINK MUTATION

In the simple mutation case, nodes are being selected to getting applied for simple hub switch mutation with the given probability. Links are selected to be added or removed first with 0.5 probability and same conditions are applied as centrality based mutation except the tournament. In this mutation, only one link is selected and no comparison is made.

5.4.2.2. HUB MUTATION

There are 2 hub mutations designed to work with link mutations, the default strategy used is the simple hub mutation. Only the centrality based hub mutation is used together with centrality based link mutation.

SIMPLE HUB MUTATION

For simple hub mutation, given hub switch ratio is applied in the loop into hub decision set. Whenever the probability hits, then regarding node is switched between hub and switch.

CENTRALITY BASED HUB MUTATION

In addition to simple hub mutation, in CENTRALITY BASED MUTATION, we use additional tournament selection whenever hub mutation probability hits.

-> Algorithm works this way,

-> If we hit hub, then randomly choose another hub. Switch less central hub to spoke.

-> If we hit spoke, then randomly choose another spoke. Switch more central spoke to hub

5.5. AMOSA BASED APPROACH

The nature of simulated annealing algorithms based on exploitation is turned into exploration by decreasing temperature. Considering the nature of the algorithm, similar mechanics are applied. Different than the genetic algorithm mutations, in simulated annealing mutations, only one hub is chosen to be switched and only one arc is chosen to

be added or removed without looping along the number of nodes with selection probability. Given methods of mutations are focused on decision of number of hub mutations and decision of number of arc mutations. Since the results at first mutation shows that using a value more than 1 for hub mutation count for every step strongly decrease the resulting hypervolume, the hub mutation count at other methods are considered as unit mutation. The general pseudocode is given in Algorithm 5-2.

Algorithm 5-2 The pseudo-code of the proposed AMOSA approach

```

1: Create a initial solutions ( $P_0$ ) and apply hill climbing
2: while (minimum temperature is not reached) do
3:   while (maximum number of iterations is not reached) do
4:     set newSolution as perturbed and repaired currentSolution
5:     Apply A* algorithm for routing on newSolution
6:     Evaluate the newSolution
7:     Check dominance status of the newSolution with the Pareto-front
8:     Select newSolution as currentSolution according to AMOSA algorithm
9:     If archive size is greater than soft limit then apply single linkage clustering
10:  end while
11: end while

```

The time complexity of the algorithm should be considered part by part; firstly, the complexity of hill climbing is $O(SL)$ and Single linkage clustering is $O(SL^2 \log SL)$ where SL stands for soft limit. Then comparison of the solution with other solutions is $O(MN)$ where M stands for the current memory size and N stands for number of objectives.

5.5.1. HUB SWITCH NEIGHBOR

Different than the NSGA-II algorithm solution, in AMOSA algorithm, the decision for node switch is not designed to work with possibility; instead, every hub switch operation switches a single random node.

5.5.2. ARC SWITCH NEIGHBOR

5.5.2.1. SWAP ALL ARCS-1

In each mutation operation a hub switch and an all arc swap operation is applied. For hub switch, hub to spoke or spoke to hub switch is decided with 0.5 probabilities first in case of availability, i.e. in case of single hub in the network only spoke to hub switch is allowed without probability. Then, for the arc mutation, same swapping all arcs method is applied as NSGA-II variation given in section 5.4.2.1 ‘Swap all arcs mutation’ with one difference; there is no mutation probability in this case, instead, exactly two nodes are selected and all arcs of them is swapped. The important part is, both of the selected nodes have to be hubs or spokes. An illustration of a sample swap all arcs operation is given in Figure 5-6.

5.5.2.2. SWAP ALL ARCS-2

All test results conducted for the first version of algorithm given in section 5.2.2.2.1. was pointing the requirement of updating the algorithm with less frequent hub mutations. Thereby, hub mutation is allowed only when the temperature is changed and was not applied along number of iterations per temperature. The rest of the algorithm is same as 5.2.2.2.1.

5.5.2.3. ADD/REMOVE K-ARCS

Considering the simple mutation of NSGA-II algorithm, same mechanism is applied in K mutation with the difference of using only one certain mutation, instead of the probability of mutation for each node. Since both of the mutation operations on decision variables are scalable, it can be tested with parameter K for number of arc mutations and parameter P for number of hub mutations.

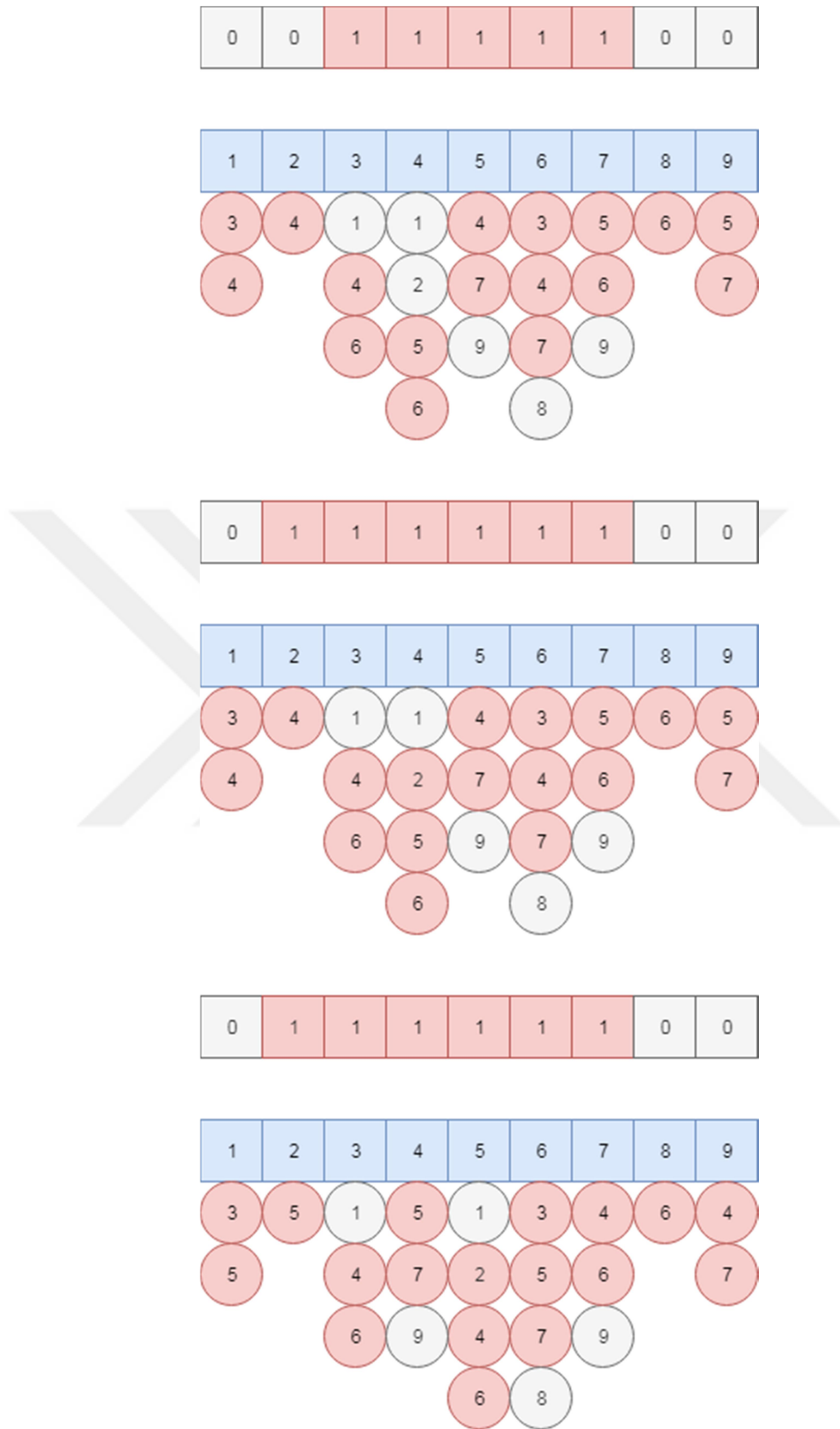


Figure 5-11 Swap All Arcs Mutation - First decision variables shows the initial mapping, second decision variables shows after bit flip operation on only node 2, third decision variables shows after swapping all arcs of nodes 4 and 5 and repairing.

5.5.2.4. ADAPTIVE ADD/REMOVE K-ARCS

After a number of tests are applied, a need of change in K multiplier is arisen. For this method, we applied 3 versions of strategies, the first one , so called adaptive K mutation rewards the algorithm by decreasing K if improvements are recorded in last N hypervolume measurements and increase K if no improvements are recorded in last N hypervolume measurements to punish and help the algorithm to increase exploration. Parameter set needs to be defined for this variation is; hypervolume queue size: N, initial K value, maximum K value. For all K mutations, minimum K amount is considered as 1 due to being unit mutation amount.

5.5.2.5. RANDOM ADD/REMOVE K-ARCS

In this mutation, K mutation count is decided randomly for each mutations between 1 and given parameter maximum K. The advantage of this mutation is, this mutation does not depend on parameters as the previous one since the only parameter required is maximum K parameter.

5.5.2.1. TEMPERATURE BASED ADD/REMOVE K-ARCS

By considering the very fundamental nature of simulated annealing algorithms, in this mutation, K value is changed by temperature starting from maximum K value to 1 by each step of temperature change. The number of temperature iterations is scaled to maximum K value, and each level of K is applied in equal number of evaluations.

6. EXPERIMENTAL STUDY

For the experimental study, we conducted two sets of experiments. In the first set of experiments, we found the optimal solutions for the MOCMAHLP for two different small data sets with a brute-force method that searches the whole solution space. We also used the same data sets to solve the problem using our NSGA-II approach. Since our heuristic approach was able to find optimal solutions for the problem with these data sets, as a second set of experiments, we conducted parameter tuning tests to explore the influence of parameter setting on the performance of our approach. In the parameter tuning tests, we used Turkish postal system data set (will be referred as Turkish data set in the rest of this thesis) introduced in [31–33].

To implement the proposed approach, an open source framework written in Java, MOEA, is used. Tests are conducted within a computer having these hardware configurations: Intel i-7 2600K processor and 8GB ram.

6.1. MONITORING INTERFACE APPLICATION

With using the same framework, a monitoring application is developed for both comparing the results and debugging purposes. First, the main function of the program which runs our heuristic algorithm is designed to serialize all the resulting solutions and save them into a file with statistics and hypervolume values. To show the results in detail, the monitoring application is designed to work with using drag-drop method of files. A general outline for all solutions are printed with pareto's hypervolume, parameters which have been used for computation, objective values of each solution of the regarding final pareto and general information and statistics about each solution is presented with the Figure 6-1.

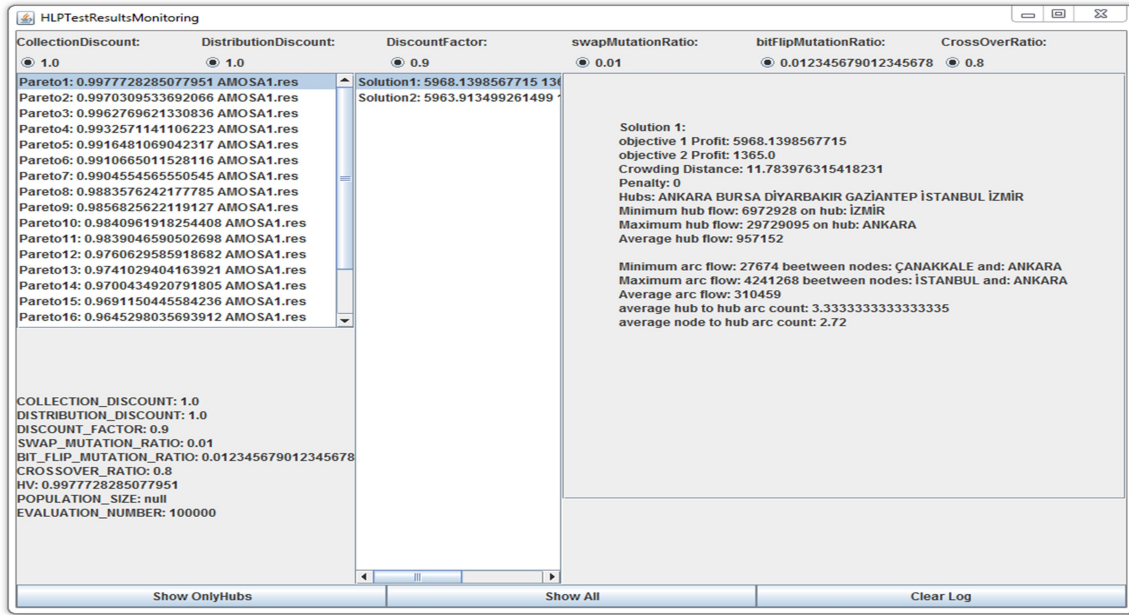


Figure 6-1 Monitoring interface application – Main view of Turkish dataset

The selected solution can be presented with showing only hubs or showing whole network options as shown in Figure 6-2 and Figure 6-3. Then, to see the routing information, left click of mouse on the map selects a source node and right click of the mouse on the map selects a destination node as shown in Figure 6-4 and Figure 6-5. Also the routing information is printed on the right corner of frame with the rule: “Flow amount from selected source to selected destination” / “Total flow amount of regarding hub or arc” – “Capacity of regarding hub or arc”. If there are more than one routing, then both routing information are printed on the map with using different colors and on the right corner of the screen as text with having additional space in between.

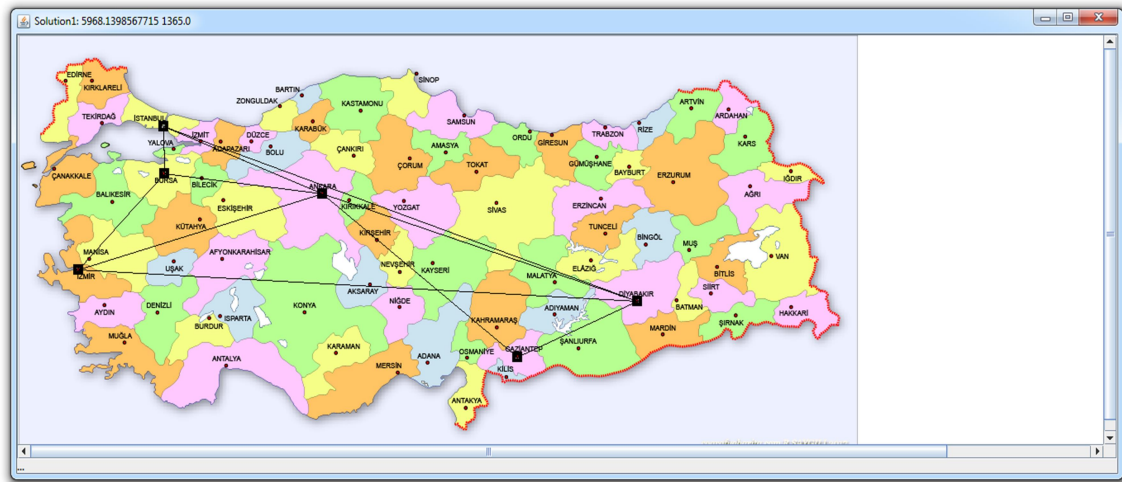


Figure 6-2 Monitoring interface application – Show Only Hubs section

h

show only hubs section given in Figure 6-2, only the hubs and arcs between hubs are drawn onto the map. Hubs are indicated with squares and arcs are indicated with lines. Moreover in show all section given in Figure 6-3, whole network is drawn onto the map. Besides hubs and arcs, spokes are indicated with circles and arcs from hub to spoke and spoke to hub are indicated as same as hub to hub arcs.

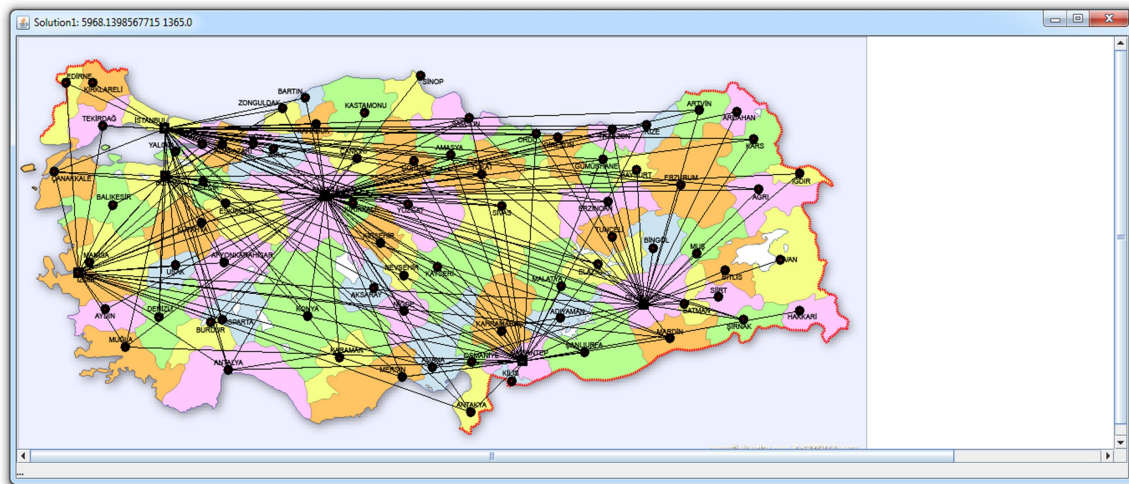


Figure 6-3 Monitoring interface application - Show All section

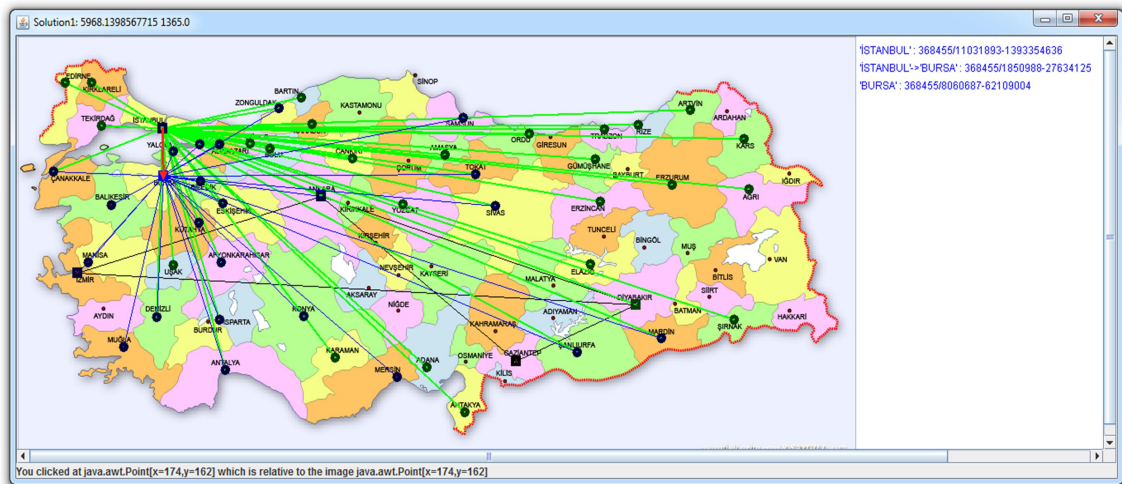


Figure 6-4 Monitoring interface application - Show Only Hubs section - Routing from Istanbul to Bursa

Whenever user clicks on drawn circle or square, related node is selected. If left button is clicked, then it is considered as the node is selected as source. With having the same logic, if right button is clicked, then it is considered as the node is selected as destination node. Whenever a source or a destination is selected, all connected nodes to the selected node are drawn with green arcs in case of source neighborhood and blue arcs in case of destination neighborhood. Additionally, In show only hubs section, choosing source or destination nodes causes neighbor nodes to being drawn. Finally, if both source and destination nodes are selected, the detailed information is printed in the log section on the right side of the panel.

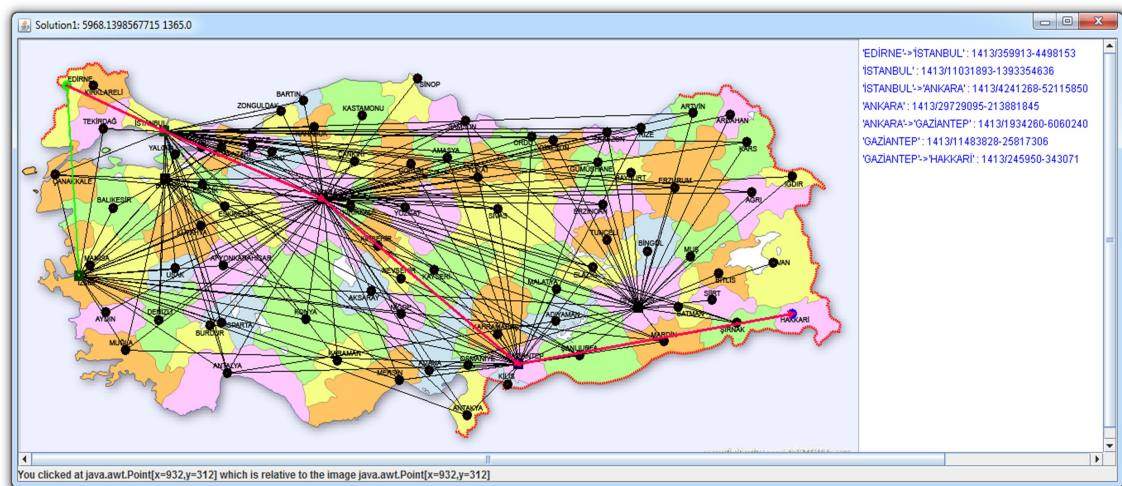


Figure 6-5 Monitoring interface application - Show Only Hubs section - Routing from Edirne to Hakkari

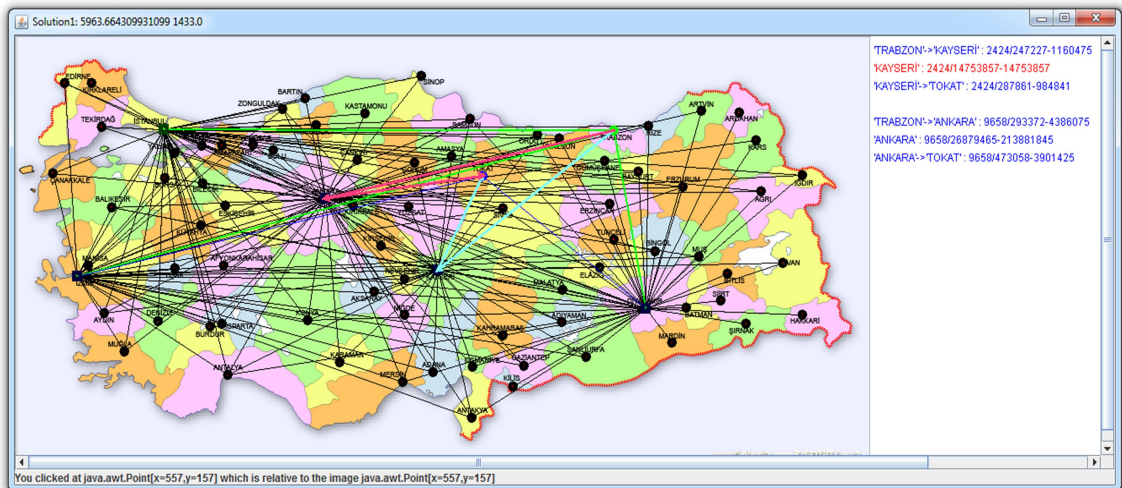


Figure 6-6 Monitoring interface application - Show Only Hubs section - Routing from Trabzon to Tokat - Multiple path case

If there are multiple paths on routing, all paths are drawn on the map with using different colors. Also every path is printed in the logs section. Arcs and/or hubs which caused the routing algorithm to produce multiple paths by reaching full capacity is printed in red color in the logs as can be also seen by values at the hub - Kayseri in Figure 6-6 that, capacity and current amount of the hub is same.

6.2. DATA SET

6.2.1. TURKISH DATA SET

Turkish data set introduced in [31]–[33] is used in the tests. It is available from OR-Library2 . The data set contains 81 nodes which correspond to the 81 cities in Turkey. In the Turkish data set, travel distances, travel times, flow, and fixed link costs between these 81 cities and fixed hub costs for these cities are given. The discount factor for the hub to hub transportation (α) is set to 0.9 as recommended in [33]. The discount factors for collection (χ) and distribution (δ) are set to 1.

Two different subsets of this large data set are generated to get optimal solutions for the problem. These subsets include 5 and 7 nodes, respectively. In the 5-node subset, we included 5 cities (Afyon, Aydın, Denizli, İzmir, Manisa) in the Aegean region of Turkey that have the largest amount of total flow demand within the region. In the 7-node subset, on the other hand, we selected the most populated cities (Ankara, Antalya, İstanbul, İzmir, Samsun, Şanlıurfa, Van) in each geographical region.

We propose a solution approach that routes each commodity considering capacity constraint on hubs and links besides location and allocation of hubs and spokes. However, there is no capacity information for hubs and links in the data set. Therefore, we generated hub and link capacities for the data set using a method explained in the next two paragraphs.

We calculated hub capacities for each city in the data set using the population size of the corresponding city declared by the Turkish Statistical Institute in 2018. To calculate the hub capacities, we propose the following equation given in Eq. (6.1). In the equation, ψ denotes hub capacity multiplier and is used for scaling. To get realistic hub capacities, the ratio of city population (P^c) to total population of the country, is multiplied with total incoming and outgoing flow demand of the city (F^c) as in Eq. (6.1).

$$Q_c = \psi \left(\frac{P_c}{\sum_{i=1}^N P_i} \right) F^c \quad (6.1)$$

$$F^c = \sum_a \begin{cases} F_a, & \text{if } a_u = c \text{ or } a_v = c \\ 0, & \text{otherwise} \end{cases}$$

To calculate link (arc) capacities (Q_a), distance and travel time for the corresponding link are used. We formulated link capacity calculation as in Eq. (6.2). In the equation, ω used for scaling denotes link capacity multiplier. To get realistic link capacities, distance required to travel through the arc (d_a) is divided by time required to travel through the arc (T_a) and is multiplied with flow demand (F_a) between the start and end nodes of arc a .

$$Q_a = \omega \left(\frac{d_a}{T_a} \right) F_a \quad (6.2)$$

The first objective of MOCMAHLP includes summation of three different costs, namely total transportation cost, fixed arc cost, and fixed hub cost, each have different orders of magnitude. In order to bring these values to the same order of magnitude, we defined and tuned some multipliers for these cost values as total transportation cost multiplier, fixed hub cost multiplier and fixed link cost multiplier. As a result of tuning the values for these multipliers, it is experienced that;

- If total transportation cost multiplier (ξ) is increased, more hubs and links appeared in the solutions.
- If fixed hub cost multiplier is increased, fewer hubs appeared in the solutions. Besides, at some point in the execution, the number of hubs decreases to 1 hub.
- If fixed link cost multiplier is increased, fewer links appeared in the solutions. At some point in the execution, each spoke is assigned to only one hub.

Similar to the costs, the capacity values in the data set have different orders of magnitude. Therefore, another set of multipliers are defined for capacities (ψ for hubs, ω for links) as given in Eq. (6.1) and Eq. (6.2). As a result of tuning the values of these multipliers, we conclude that:

- As ψ is decreased, hubs are mostly selected among more populated cities. When ψ is decreased further, then cities which are close to each other are selected as hubs to convey the demand through these cities.
- As ω is decreased, the number of hubs that the spokes are allocated is increasing, thereby total capacity of transportation on links is increasing.

For the first objective, i.e. minimizing the total cost, we need to calculate travel cost of each arc. The cost of traveling through arc a (C_a) is calculated as $C_a = \xi_{da}$, where ξ denotes cost per flow per unit distance, so that by multiplying with the distance of arc gives us cost to move one unit flow through arc. Therefore, as in the first objective, multiplying C_a with total flow on arc gives the true cost of transfer through the arc.

As a result of experiments, the multipliers are set to the following values:

- total transportation cost multiplier (ξ) = 10^{-7} ,
- fixed hub cost multiplier = 0.2,
- fixed link cost multiplier = 1.0,
- $\psi = 50$,
- $\omega = 400$.

These settings are determined empirically as a result of a set of preliminary experiments.

6.2.2. AP DATA SET

A test data set reduced from AP data set is introduced in [36] by Andreas T. Ernst. Even though there are many reduced N noded data sets, 20-noded data set has been chosen due to low complexity and named N20 dataset throughout the thesis. The data sets which have the closest node count to 20 are 10 and 25. The data set which has only 10 nodes is too small to observe the benefits of meta-heuristics and too complex for linear solutions. On the other hand, 25-noded data set is quite large to test and be a step to 81 node network tests.

N20 dataset contains one more data in comparison with Turkish dataset: Hub capacities, two less data in comparison with Turkish dataset: travel time and fixed link costs. To calculate travel time, the distance between nodes (Euclidian distance between nodes' coordinates) is divided by 90 as same method usage in Turkish dataset. To calculate fixed

link costs, Euclidian distance between nodes' coordinates are calculated and thought about direct correlation between distance and fixed link cost; so that fixed link cost is set to distance between nodes. Finally link capacity is calculated as, multiplication of total direct flow between nodes and link capacity multiplier.

6.3. PERFORMANCE METRIC

Since the problem we study is a multi-objective problem, hypervolume indicator [34] which is the most used metric, is selected to evaluate the performance of our algorithm and to compare different test suits. Hypervolume indicates the volume between reference point in the solution space and the Pareto front found by the solution approach. The hypervolume is calculated based on the normalized objective function values. We use the hypervolume calculation provided in MOEA framework for the hypervolume of the Pareto front. Figure 6-7 illustrates the hypervolume indicator.

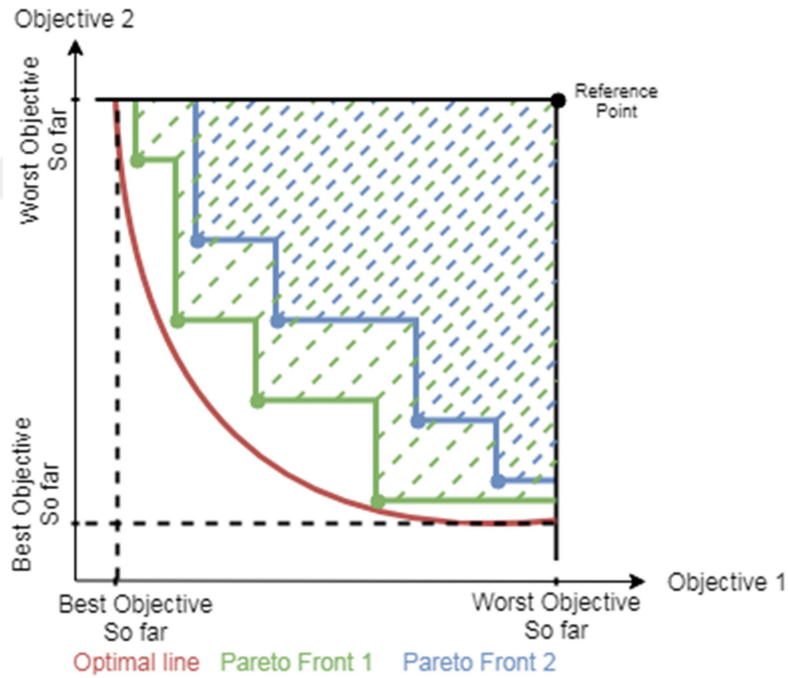


Figure 6-7 Hypervolume indicator

Thus, to obtain a properly chosen reference point, we executed our algorithm 5 times with the selected data set using 200000 evaluations, which is rather a long execution for the problem. At this stage, the algorithm has been converged (see Figure 6-13). The worst and

the best objectives of the problem are obtained for these 5 runs to determine the reference point.

6.4. SOLUTIONS FOR SMALL DATA SETS

To prove and find the best solutions, we implemented linear approach which works on greedy basis. 2 different variation variables used as in type of ‘Long’, one carries hub decision variation and the other carries link decision variable. Since data type of “Long” carries 64 bits of data, hub decision variation variable can support maximum of 64-node network and link decision variation variable can support maximum of 11-node network due following equation:

$$\frac{n(n-1)}{2} < 64$$

Thereby, selected data structure limits linear approach to be used with maximum of 11-node network. But according to complexity of linear approach given in Eq. (6.3), it takes approximately $4.5611 * 10^{28}$ years to calculate the results for 12-node network. Depending on this approximated time estimation, this limitation was fine for the problem.

Every bit in the hub decision variation variable stands for each node, if the bit is set then it is considered that the related node is hub, if else spoke. In the same way, the bits in the link decision variation variable stands for each link, if the bit is set then it is considered that the related link is set and active, if else not set and related nodes are disconnected. For each iteration, regarding network has been created and routing is calculated. Since routing requires a valid network, before calculation of routing, network is controlled and repaired in case of invalid network. Finally, throughout the iteration, each possible solution is evaluated and inserted into pareto set by domination or non-domination status excluding dominated status. Obtained final set after the iteration is the global optimum.

6.4.1. 5-NODE REDUCED EAGEAN AREA DATA SET

In order to examine the correctness of our mathematical model and to observe the performance of our algorithm, first we conducted tests with the small data sets mentioned above. Optimal solutions are found with a brute-force method that evaluates every possible solution and then inserts into global non-dominated Pareto set by comparing with all solutions in the Pareto set.

The maximum number of feasible networks that can be generated for N number of nodes is given in Eq. (6.3). There are 3 parts in the equation; the first part calculates the permutation of i-hubs within division, the second part determines the number of spoke-to-hub connections and the last part determines the number of hub-to-hub connections.

In the first set of tests, to evaluate the effectiveness of our heuristic approach we used the small data sets for which we obtained the optimum values. Unless stated otherwise, we consider the following parameter setting for our heuristic approach in these tests:

- Population size (popsize) is set to 200
- Crossover probability (p_c) is set to 0.8
- Bit-flip mutation probability (p_{bfm}) is set to $1/n$ where n is the number of hubs in the network
- Swap mutation probability (p_{sm}) is set to 0.2
- The maximum number of fitness evaluation count is set to 20000

$$\sum_{i=1}^N \binom{N}{i} (2^i - 1)^{N-i} (2^{i-1} - 1)^i \quad (6.3)$$

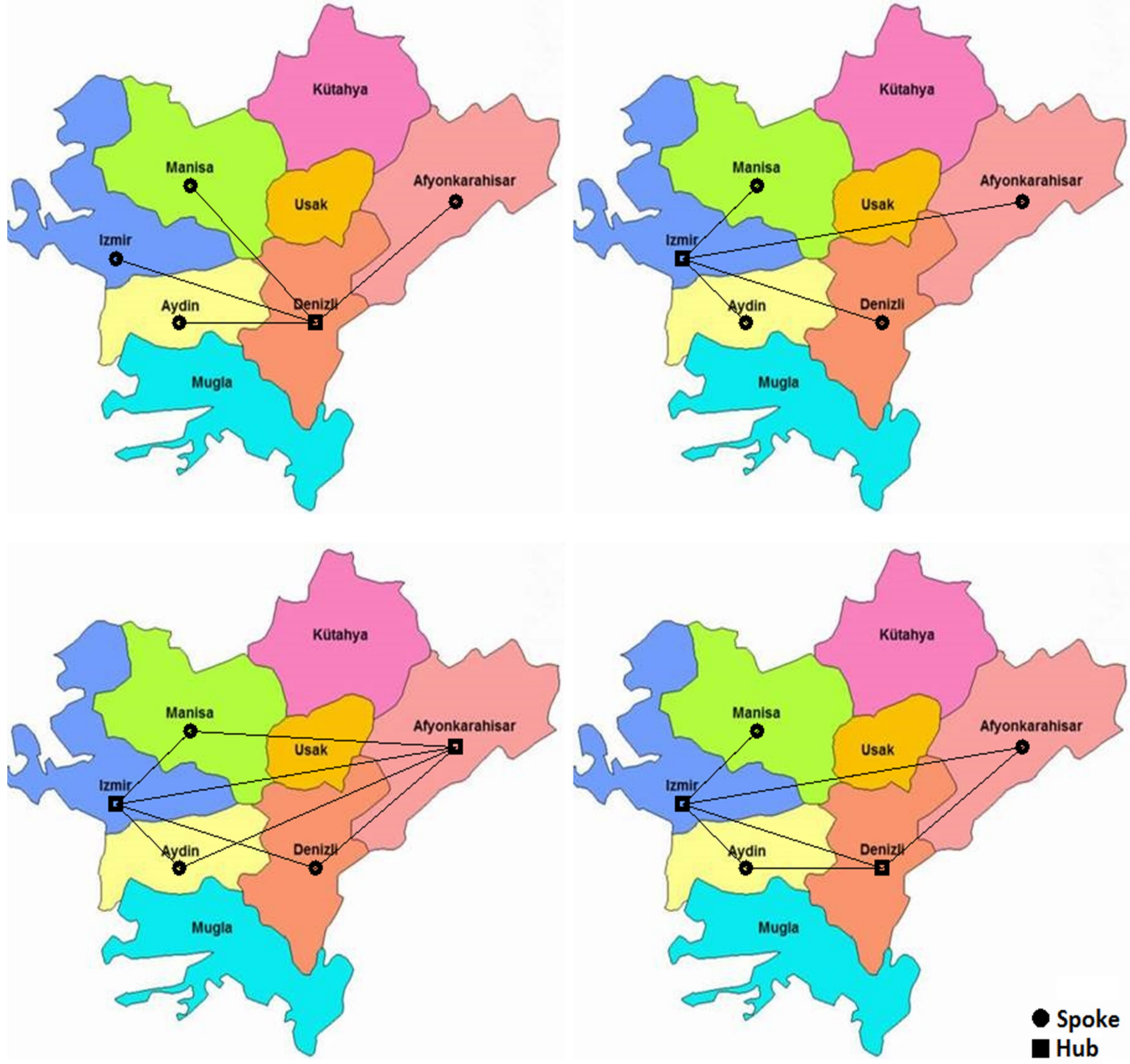


Figure 6-8 Pareto-optimal solutions obtained by brute-force method for 5-node network
(Upper solutions have 1, lower solutions have 2 hubs)

Experimental results show that, the brute-force method finds 4 solutions in non-dominated Pareto set for the 5-node network, with the hubs being Denizli, İzmir, İzmir-Afyon and İzmir-Denizli. Figure 6-8 shows an example optimum solution obtained for 5-node network.

6.4.2. 7-NODE REDUCED TURKISH DATA SET

Optimum pareto front of one of the 7-node data set is given in Figure 6-9. There are 6 solutions in the nondominated pareto set for this network, for which the hubs are selected as, Ankara, İstanbul, İstanbul - Samsun, İstanbul - Şanlıurfa, İstanbul - Şanlıurfa and İstanbul - Van - İzmir.

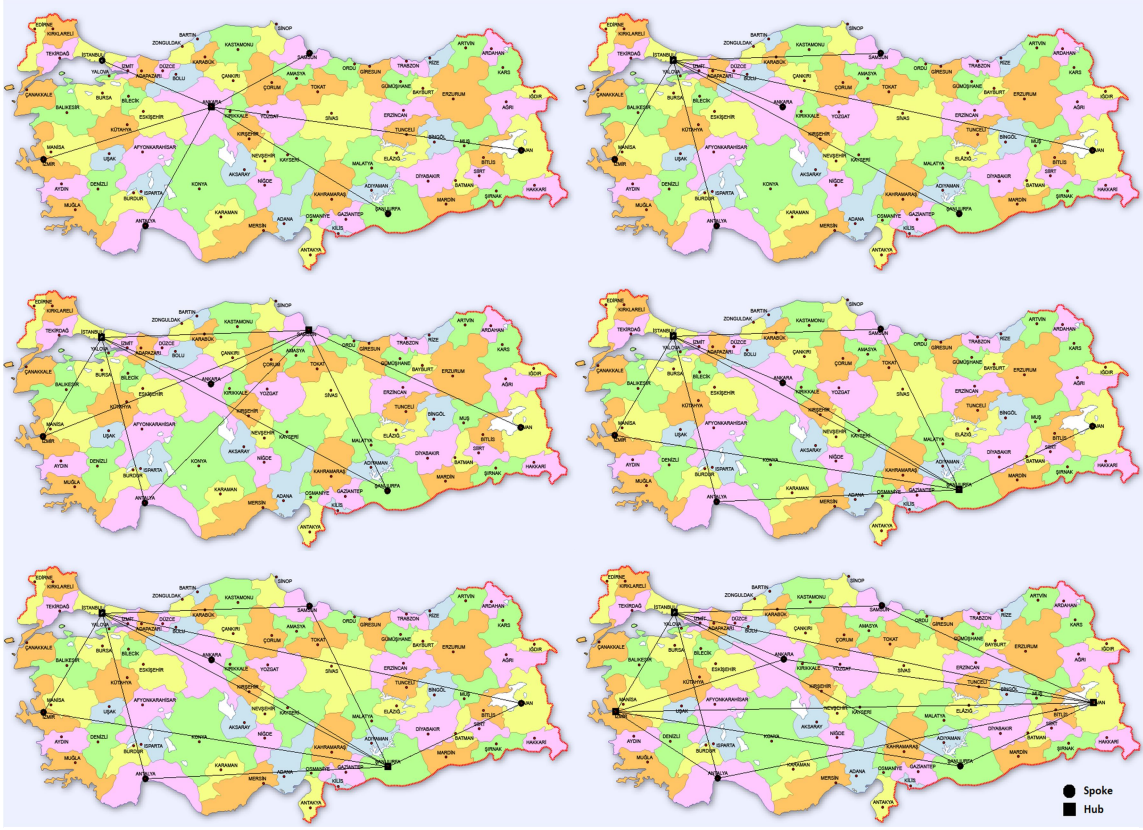


Figure 6-9 Pareto-optimal solutions obtained by brute-force method for 7-node network (From up-left corner to down-right corner the number of hubs at each case is equal to 1, 1, 2, 2, 2 and 3 hubs)

Our AMOSA and NSGA-II based algorithms is applied to these small data sets to explore its effectiveness on the problem while the optimum pareto front is known with following parameter change: Swap mutation probability (p_{sm}) is set to 0.05.

Our approach finds exactly the same Pareto set with linear solution for 5-node network on each run (see Figure 6-8). For 7-node network (see Figure 6-9), our approach obtains exactly the same Pareto set with linear solution in %20 of runs (i.e., in 2 out of 10 runs). For tests with the given 7-node network, average hypervolume ratio found with our heuristic to linear hypervolume ratio is %99, 98 and the ratio of worst hypervolume value of our heuristic to linear hypervolume value is %99, 91.

In addition, we generate 500 different sub networks by reducing the Turkish Dataset to test our heuristic approaches and their variation functions against linear solution. For each sub network, we run each neighborhood of each algorithm 200 times and we saved average

hypervolume values along 200 runs. Finally we divide the average hypervolume value of each neighborhood of our heuristics by hypervolume value of linear solution to get the ratio of approximation to linear solution. The results can be seen in Figure 6-10.

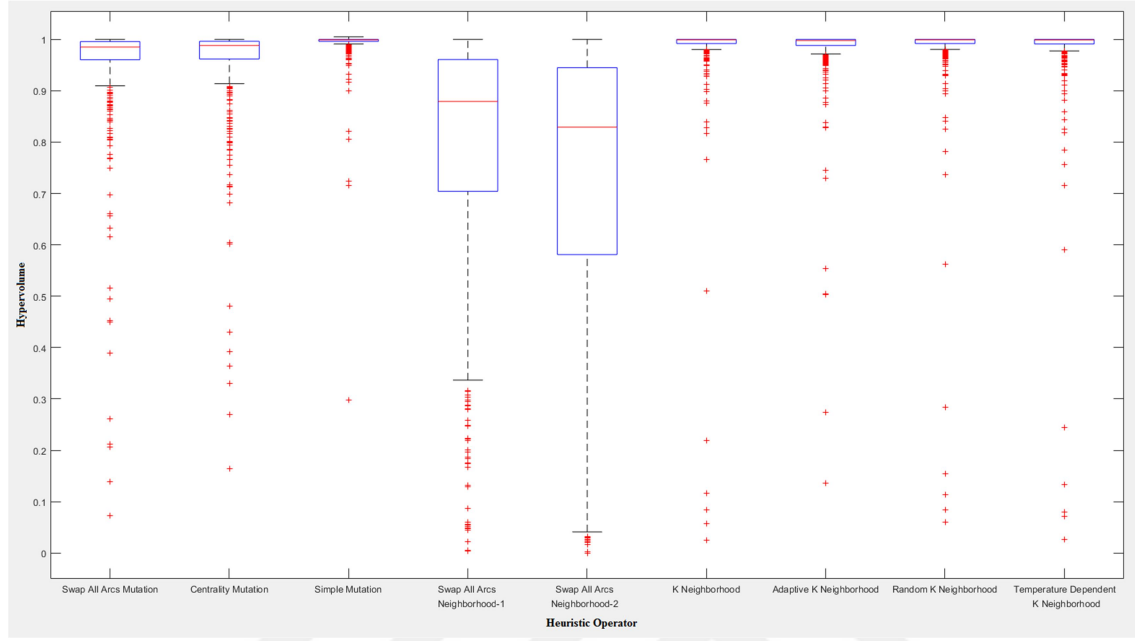


Figure 6-10 Box-plot of ratio of average hypervolume value of 200 runs of regarding algorithm's heuristic operators to hypervolume value of linear solution for 500 different datasets reduced from Turkish dataset. Each run is limited with 20.000 number of evaluations and default parameters are used for every neighborhood.

If we consider the execution times, our heuristic approach obtains the results in less than 5 seconds for 7-node network and it takes even less computation time for 5-node network. On the other hand, the brute-force method finds optimum values for 5-node network in a few seconds, and it takes up to 3 hours to calculate the optimum results for 7- node network. If we increase the network size to 81-nodes using the complete Turkish data set, it gets impossible to find a solution using brute-force method, but our heuristic finds feasible solutions in less than 10 minutes. Table 6-1 summarizes the approximate execution times of brute-force method and our heuristic approach for different sizes of networks. These results show that our heuristic approach is very effective to solve MOCMAHLP.

Table 6-1. Execution times of brute-force method and our heuristic for different network sizes

# of nodes	Brute-Force	Our heuristic
5	3 sec.	2 sec.
7	11013 sec.	3 sec.
81	NA	251 sec.

6.5. EXPERIMENTS ON AP DATA SET

To test NSGA-II and AMOSA algorithms on smaller data set and observe the effects of our variation methods within an easier problem which lasts earlier than the original problem, a 20-nodes dataset generated by using Australia post by Andreas T. Ernst is[36] used. We have kept common parameters the same among the NSGA-II and AMOSA algorithms by setting collection discount as 1, distribution discount as 1, discount factor as 0.9, fixed link cost multiplier as 1, fixed hub cost multiplier as 0.2 and finally total transfer cost multiplier as 10^{-7} . Additionally for NSGA-II algorithm, this configuration is set; bit flip mutation ratio as $1/N$, swap mutation ratio as 0.05, cross over ratio as 0.8 and population size as 200. Additionally for AMOSA algorithm, these configuration is set; soft limit as 100, hard limit as 10, γ as 2, t_{Min} as 10^{-7} , t_{Max} as 200, α as 0.9415 to keep total evaluation number fixed at 20000, number of iterations per temperature as 50, number of hill climbing iterations as 10. The map is obtained randomly from internet for only show case purposes. The only criteria for choosing the map were, if the coordinates given in dataset was staying on a ground or on a city in the map.

An example resulting pareto frontfrom run of AMOSA algorithm is given in Figure 6-11. As can be seen in the figure, pareto front consists of 9 solutions.

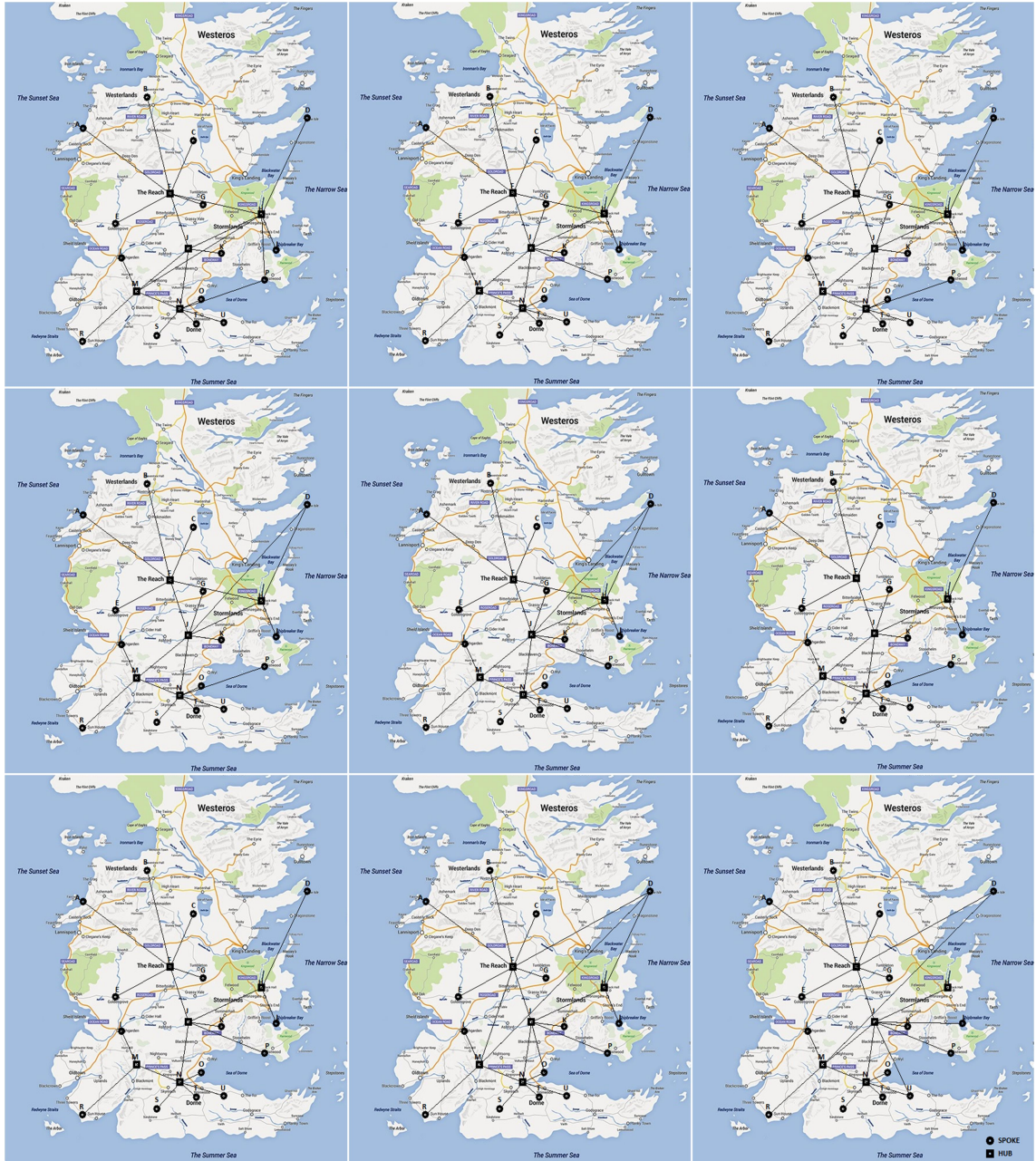


Figure 6-11 Pareto-front solutions obtained by AMOSA algorithm on 20-Node network.

Table 6-2 Average HyperVolume values of 200 runs for 20-Node dataset at certain number of evaluations

Algorithm	Heuristic Operator	Average HyperVolume at number of evaluation			
		5.000	10.000	20.000	50.000
NSGA	Swap All Arcs Mutation	0.8889	0.9153	0.9288	0.9286
	Centrality Based Mutation	0.9122	0.9502	0.9841	0.9944
	Simple Mutation	0.9013	0.9393	0.9751	0.9949
AMOSA	Swap All Arcs Neighbourhood-1	0.9140	0.9369	0.9520	0.9696
	Swap All Arcs Neighborhood -2	0.8106	0.8594	0.8792	0.9094
	K Neighborhood	0.9237	0.9393	0.9501	0.9611
	Adaptive K Neighborhood	0.9358	0.9381	0.9460	0.9554
	Random K Neighborhood	0.9334	0.9418	0.9575	0.9650
	Temperature Dependent K Neighborhood	0.9362	0.9458	0.9605	0.9686

Since the dataset contains 20-nodes, linear solution is not possible to obtain in real time. So our NSGA and AMOSA based meta-heuristic approaches and all implemented operators of them are applied on the dataset. Test results which are given in Table 6-2, shows that centrality mutation based NSGA and temperature dependent K neighborhood based AMOSA algorithms have given the best results except the longest test case: number of evaluation equals 50.000. Nevertheless, they are still very close to maximum average hypervolume value reached by their opponents for the last test case.

Boxplot of the test results for every mutation and neighborhood operations is given in Figure 6-12. Moreover, the statistical comparison is given in Table 6-3. It can be inference that for this problem, best strategy to use is centrality based mutation because it outperforms many other mutations. The detailed analysis is given in Table 6-3.

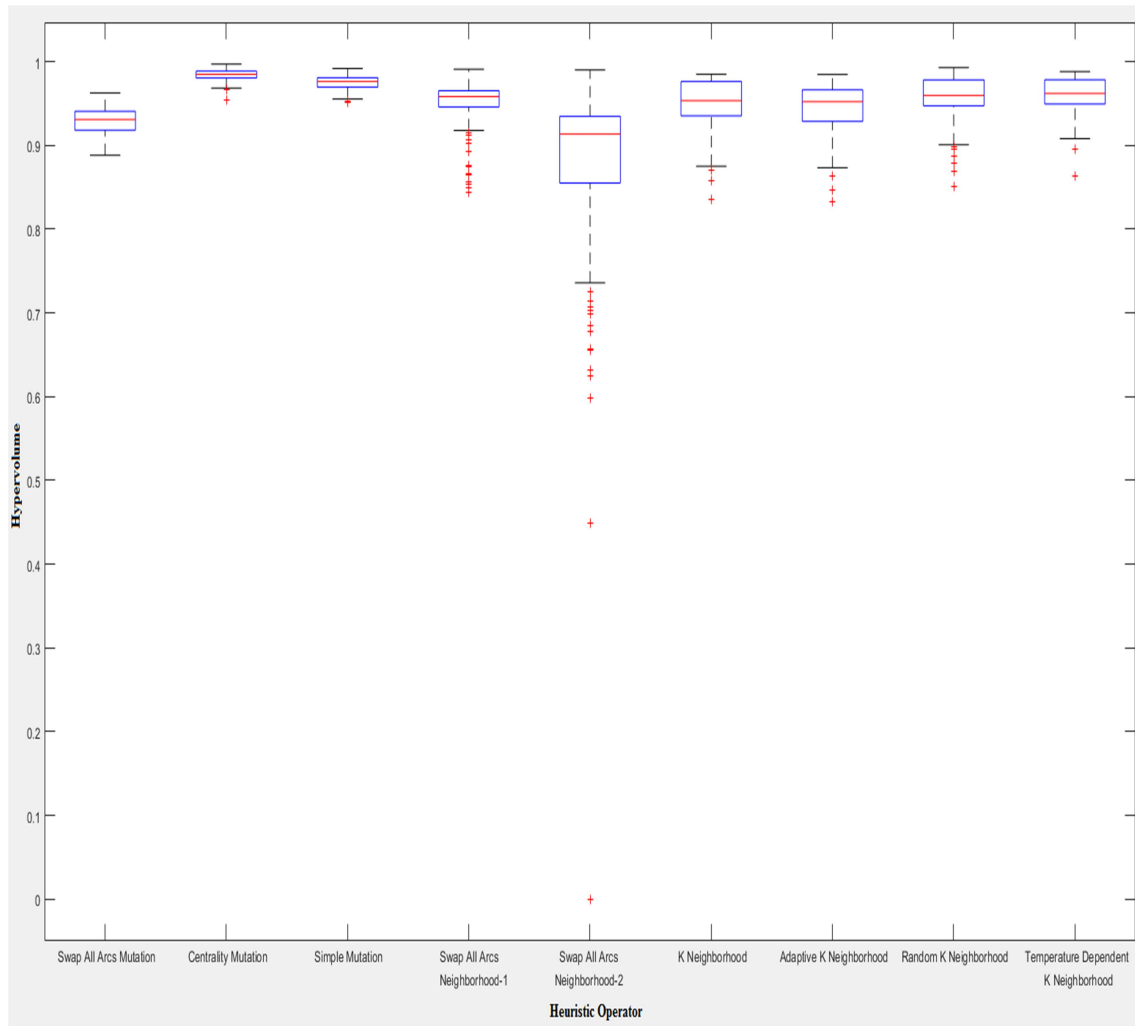


Figure 6-12 Box-plot of hypervolume values of 100 runs for a statistical comparison of different operators of AMOSA and NSGA-II algorithms for N20 dataset.

Table 6-3 Statistical comparison of test results obtained in N20 dataset. s^+ indicates algorithm on row section is statistically better than algorithm at column section. s^- indicates statistically worse. + sign indicates algorithm at row is better than the one at column. – sign indicates the opposite of +

	SAAM	CM	SM	SAAN1	SAAN2	KN	AKN	RKN	TDKN
SAAM		s^-	s^-	s^-	s^+	s^-	s^-	s^-	s^-
CM	s^+		+	s^+	s^+	s^+	s^+	s^+	s^+
SN	s^+	–		s^+	s^+	s^+	s^+	s^+	s^+
SAAN1	s^+	s^-	s^-		s^+	+	+	–	–
SAAN2	s^+	s^-	s^-	s^-		s^-	s^-	s^-	s^-
KN	s^+	s^-	s^-	–	s^+		+	–	–
AKN	s^+	s^-	s^-	–	s^+	–		–	s^-
RKN	s^+	s^-	s^-	+	s^+	+	+		–
TDKN	s^+	s^-	s^-	+	s^+	+	s^+	+	

In Table 6-3, SAAM stands for Swap All Arcs Mutation, CM stands for Centrality Mutation, SM stands for Simple Mutation, SAAN1 stands for Swap All Arcs Neighborhood-1, SAAN2 stands for Swap All Arcs Neighborhood-2, KN stands for K neighborhood, AKN stands for Adaptive K neighborhood, RKN stands for Random K Neighborhood, TDK stands for Temperature Dependent K Neighborhood.

6.6. EXPERIMENTS ON TURKISH DATA SET

In order to explore the influence of the parameter settings on the performance of our approach, a set of experiments are conducted by varying values of population size (popsize), crossover probability (p_c), bit-flip mutation probability (p_{bfm}), and swap mutation probability (p_{sm}). In the experiments, each run is repeated 10 times for the given setting. The termination condition for our heuristic is selected as number of fitness evaluations. To determine this value for the main experiments, we ran tests for our heuristic on Turkish data set for a long number of fitness evaluations. The change of fitness values is plotted in Figure 6-13. As it can be seen from the graph, the fitness values converge to a minimum in approximately 20000 number of evaluations which correspond to 100 generation counts. Therefore, the maximum fitness evaluation count is set to 20000 for each run in the experimental study.

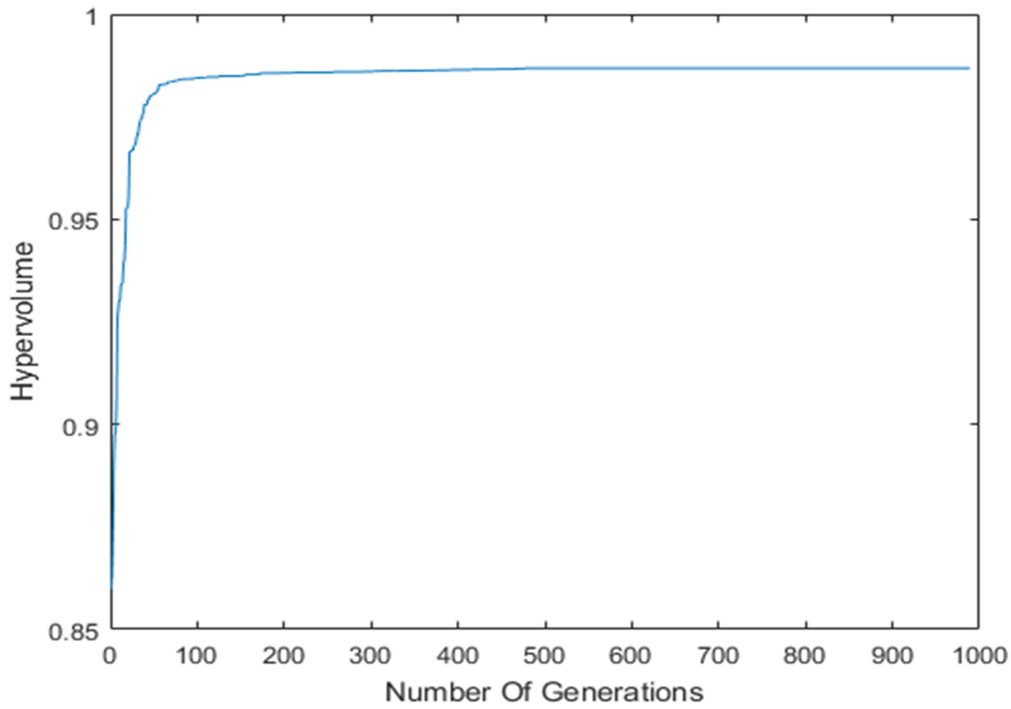


Figure 6-13 Convergence plot generated by the hypervolume values versus generation counts

Table 6-4 Average HyperVolume values of 20 runs for Turkish dataset (Number of evaluation= 20.000)

Algorithm	Heuristic Operator	Average HyperVolume	Maximum HyperVolume
NSGA	Swap All Arcs Mutation	0.9620	0.9693
	Centrality Based Mutation	0.9326	0.9578
	Simple Mutation	0.9593	0.9652
AMOSA	Swap All Arcs Neighborhood -1	0.6754	0.7446
	Swap All Arcs Neighborhood -2	0.3166	0.6892
	K Neighborhood	0.9087	0.9825
	Adaptive K Neighborhood	0.8115	0.9972
	Random K Neighborhood	0.9754	0.9978
	Temperature Dependent K Neighborhood	0.8895	0.9863

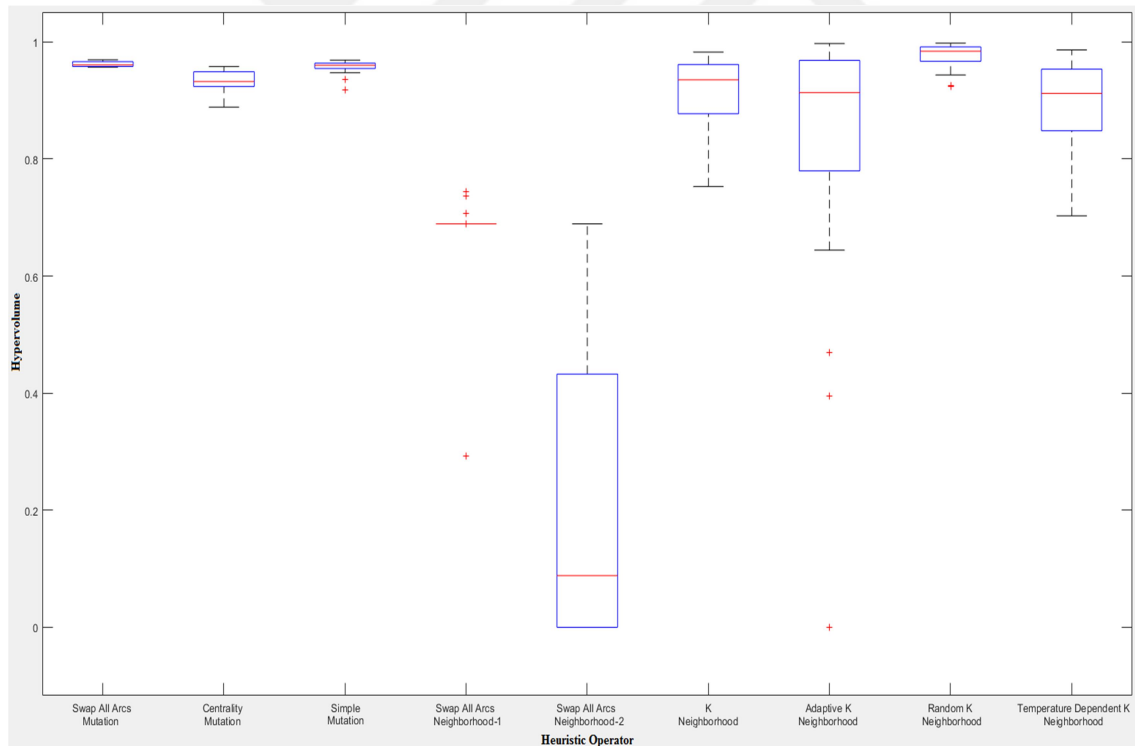


Figure 6-14 Box-plot of hypervolume values of 20 runs for a statistical comparison of different operators of AMOSA and NSGA-II algorithms for Turkish original dataset

The indicated values in Table 6-4 and detailed illustration in box-plot in Figure 6-14 are obtained after parameter tuning if the operator requires. For instance, simple mutation

operator for NSGA-II applies only one arc change in the network and it requires parameter tuning for parameter; arc mutation ratio. Table 6-6 clearly shows that the best arc mutation ratio parameter value for simple mutation operator is 0.12.

Table 6-5 Statistical comparison of test results obtained in original Turkish dataset. s^+ indicates algorithm on row section is statistically better than algorithm at column section. s^- indicates statistically worse. + sign indicates algorithm at row is better than the one at column. – sign indicates the opposite of +

	SAAM	CM	SM	SAAN1	SAAN2	KN	AKN	RKN	TDKN
SAAM		+	+	s^+	s^+	+	s^+	+	+
CM	-		-	s^+	s^+	+	+	+	+
SN	-	+		s^+	s^+	+	+	+	+
SAAN1	s^-	s^-	s^-		s^+	s^-	-	s^-	s^-
SAAN2	s^-	s^-	s^-	s^-		s^-	s^-	s^-	s^-
KN	-	-	-	s^+	s^+		+	+	+
AKN	s^-	-	-	+	s^+	-		-	-
RKN	-	-	-	s^+	s^+	-	+		-
TDKN	-	-	-	s^+	s^+	-	+	+	

The statistical comparison of algorithms on the results of 20 runs on Turkish original data set is given in Table 6-5. The naming is same with the Table 6-3. This test result shows that none of the mutations except Swap All Arcs Neighborhood is dominating each other.

Table 6-6 Simple Mutation – arc mutation ratio parameter tuning (Number of evaluation= 20.000)

Arc Mutation Ratio	Average HyperVolume	Maximum HyperVolume
0.01	0.9410	0.9609
0.05	0.9548	0.9666
0.10	0.9564	0.9688
0.11	0.9558	0.9644
0.12	0.9593	0.9652
0.13	0.9552	0.9655
0.14	0.9556	0.9649
0.16	0.9548	0.9660

Another parameter tuning has been made for Random K-Neighborhood operator on number of iterations per temperature value is given in Table 6-8. The AMOSA parameter α is calculated according to predefined maximum number of evaluations by this equation:

Table 6-7 Variable notations

Variable	Notation
N_{ipt}	Number of iterations per temperature
$N_{maxIter}$	Maximum number of iterations
N_{hci}	Number of hill climbing iterations
N_{sl}	Soft Limit

$$\alpha = (T_{min}/T_{max})^{\left(\frac{N_{ipt}}{N_{maxIter} - (N_{hci} * \gamma * (N_{sl} + 1))}\right)}$$

Since the results of Table 6-8 shows that the best number of iterations per temperature value is 10, which is far by distance to default value of the AMOSA algorithm: 50, especially in average hypervolume value. Thereby, it has been set as the default value of Random K Neighborhood operator.

Table 6-8 Random K Neighborhood – number of iterations per temperature parameter tuning by stabilizing α parameter to keep maximum number of evaluations 20.000 by means of average hypervolume and maximum hypervolume

Number of iterations per temperature	Average HyperVolume	Maximum HyperVolume
9	0.9575	0.9896
10	0.9754	0.9978
11	0.9127	0.9916
13	0.9314	0.9964
15	0.8683	0.9916
20	0.8672	0.9920
25	0.9214	0.9954
30	0.8648	0.9954
50	0.8651	0.9965

In the result tables, the value of tested parameter(s) are given in the first one or two columns and the corresponding average and standard deviation of hypervolume values are given in the last two columns. In the rows of the tables, you can see the performance of each parameter setting. In all tables, the row with the best setting is marked in bold.

In the parameter tuning tests, we experimented with each combination of the following set of values for the parameters:

- Population size: $popsiz \in \{50, 100, 200\}$
- Crossover probability: $p_c \in \{0.5, 0.8, 0.9, 1.0\}$
- Bit-flip mutation probability: $p_{bfm} \in \{1/n, 0.5/n\}$
- Swap mutation probability: $p_{sm} \in \{0.005, 0.01, 0.02, 0.05\}$

In the remaining part, we give results of parameter tuning tests. The results are given in different tables for a better illustration of the effect of parameters. In Table 6-9, experimental results for various population sizes are given. We fixed the crossover probability value to $p_c = 0.9$, bit-flip mutation probability value to $p_{bfm} = \frac{1}{n}$ (n is the number of nodes in the network) and swap mutation probability value to $p_{sm} = 0.01$. Based on the results provided in Table 6-9, it can be seen that, our approach delivers a good performance with $popsiz = 200$.

Table 6-9 Hypervolume results and standard deviation for our approach with various population sizes popsize and fixed parameter settings

$$(p_c = 0.9, p_{bfm} = \frac{1}{n}, p_{sm} = 0.01)$$

popsize	Mean	Std. Deviation
50	0.9894	0.0088
100	0.9839	0.0090
200	0.9927	0.0046

In Table 6-10, the performance of our approach is given for different crossover probabilities, while keeping parameter settings as: (popsize = 200, $p_{bfm} = \frac{1}{n}$, $p_{sm} = 0.01$)

Table 6-10 Hypervolume results and standard deviation for our approach with various crossover probabilities p_c and fixed parameter settings

p_c	Mean	Std. Deviation
0.5	0.9883	0.0114
0.8	0.9948	0.0005
0.9	0.9927	0.0046
1.0	0.9884	0.0090

Figure 6-15 shows the box-plot for the hypervolume values of our approach with different p_c . From Table 6-10 and Figure 6-16, it can be observed that our algorithm performs well when $p_c = 0.8$.

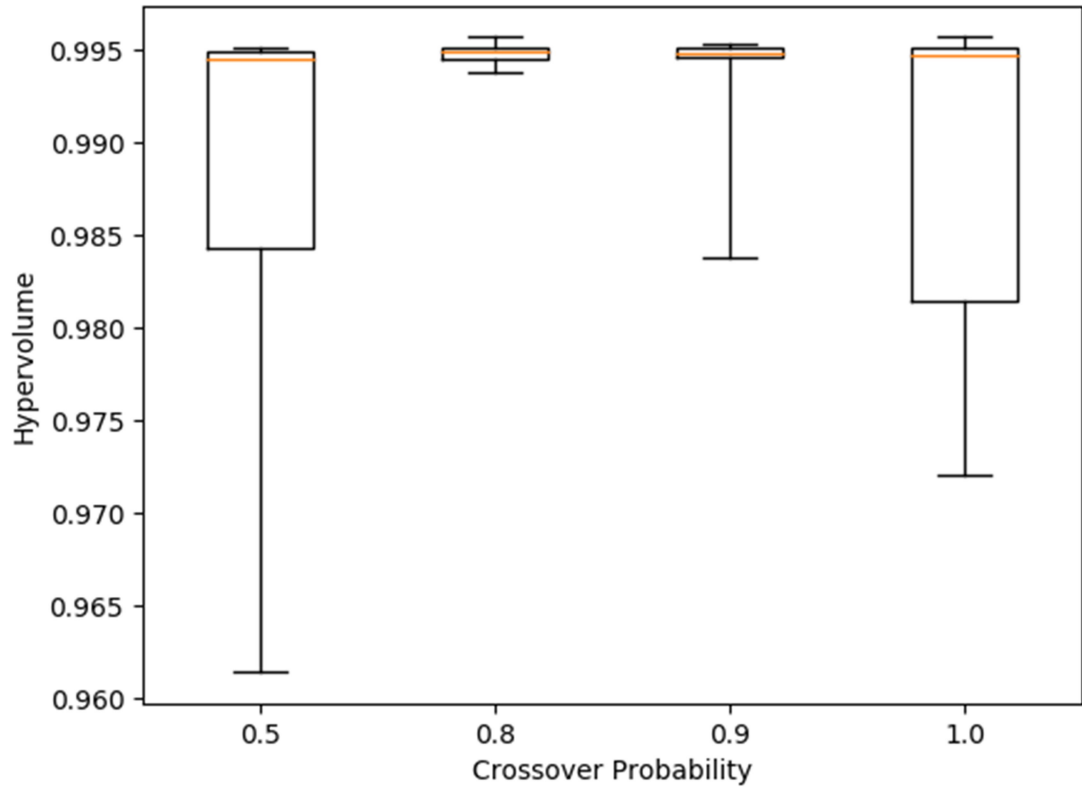


Figure 6-15 Box-plot of hypervolume values for a statistical comparison of different parameter settings for p_c

The effect of the probability for the bit flip p_{bfm} and swap mutations p_{sm} is given in Table 6-11. We can conclude that, the best setting for p_{bfm} is $\frac{1}{n}$ and for p_{sm} , it is 0.01.

Table 6-11 Hypervolume results and standard deviation for our approach with various p_{bfm} and p_{sm} and fixed values of $popsiz = 200$ and $p_c = 0.8$

p_{bfm}	p_{sm}	Mean	Std. Deviation
$\frac{1}{n}$	0.005	0.9887	0.0087
	0.01	0.9948	0.0005
	0.02	0.9906	0.0072
	0.05	0.9939	0.0042
$\frac{0.5}{n}$	0.005	0.9923	0.0054
	0.01	0.9935	0.0057
	0.02	0.9922	0.0059
	0.05	0.9926	0.0055

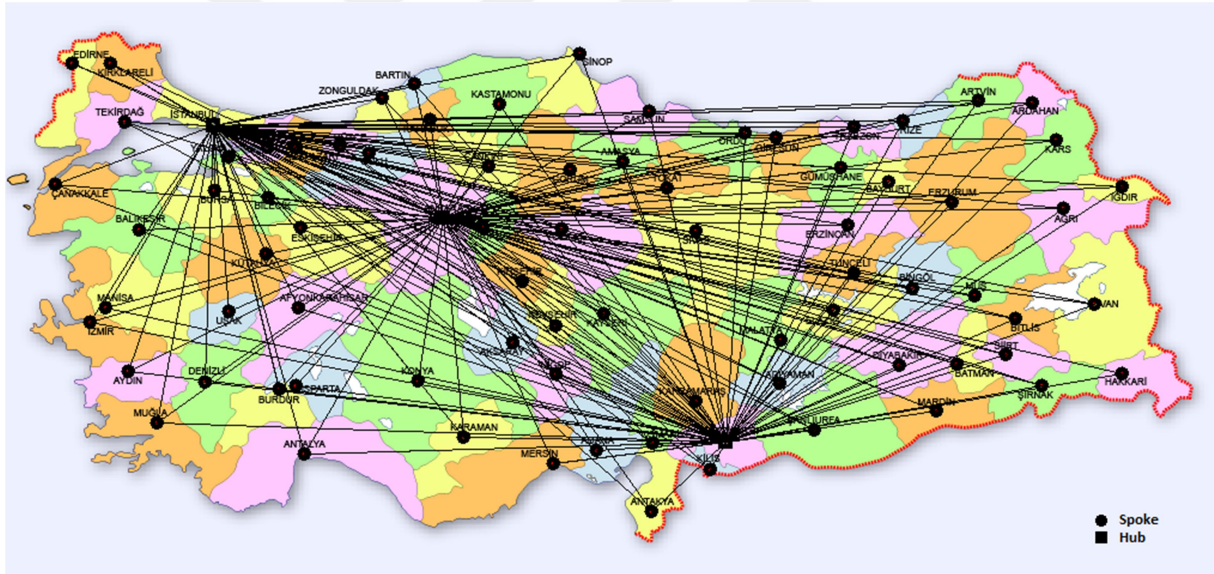


Figure 6-16 The first Pareto optimal solution that represents a hub network with all hub-to-hub and hub-to-spoke links obtained by our approach

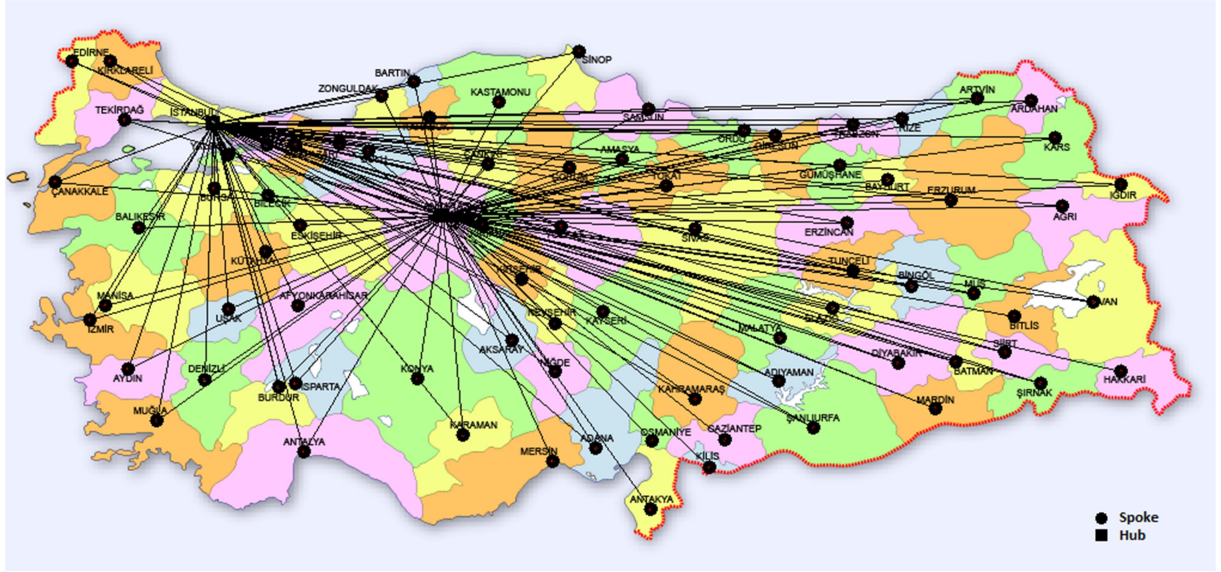


Figure 6-17 The second Pareto optimal solution that represents a hub network with all hub-to-hub and hub-to-spoke links obtained by our approach

After the parameter tuning tests, we conducted main tests on Turkish data set to find solutions for the problem. Figure 6-16 illustrates a Pareto optimal solution that represents a hub network including all hub-to-hub and hub-to-spoke links obtained by our approach with the best parameter settings. This network is chosen from the best hypervolume valued run and according to having the lowest crowding distance in the Pareto set. As seen in the figure, Istanbul, Ankara and Gaziantep which are among the most populated cities in Turkey are selected as hubs. Based on the results, the number of assignment of a spoke to a hub is 2.8 on average. Besides, the average number of hub to spoke connection is 73. On the other hand, Figure 6-17 illustrates a Pareto optimal solution from the same Pareto optimal set. In this solution, Istanbul, which is the most populated city and Ankara which is the capital of Turkey are selected as hubs. Selecting a hub at east part of Turkey, namely Gaziantep, increases the total costs while decreasing the total traveling time according to these solutions. In addition, most cities in the Southeastern Anatolia Region, such as Osmaniye, Kilis, Mardin, etc., are only assigned to Ankara.

A sample solution inside the resulting pareto optimal solutions of AMOSA algorithm with having the best resulting configuration is given in Figure 6-18. It can be concluded from the given figure that, decisions of AMOSA algorithm have larger amount of hub cardinality with having better arc allocations. The cities which has low population and low demand has

fewer arc connections. On the other hand, cities which has high population and high demand is either decided as being hub or having larger amount of arc connections.

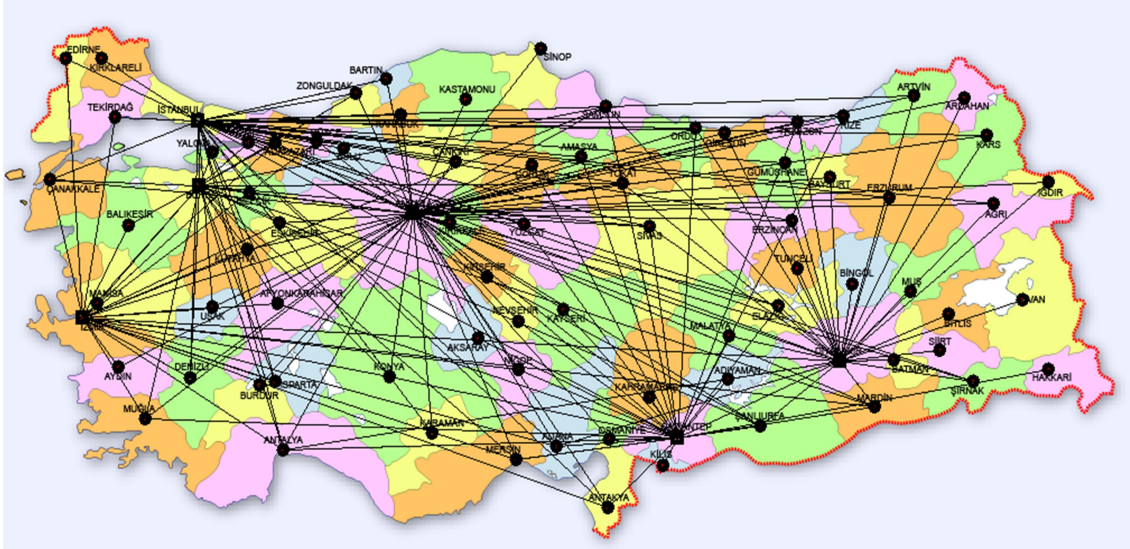


Figure 6-18 Resulting pareto optimal solution of AMOSA algorithm with Random K neighborhood having the hypervolume value in Table 6-4

In Table 6-12, we list some routing scenarios for some source destination pairs with respect to the first Pareto solution given in Figure 6-16 and the second Pareto solution given in Figure 6-17. The routes for each pair may be different. For instance, the commodity from Ordu to Antakya is routed through Gaziantep for the first solution and Ankara for the second solution. While the travel time for the first path (Ordu → Gaziantep → Antakya) is 15 hours, it is 21 hours for the second path. On the other hand, the commodity from Hakkari to Edirne is routed through the same path and the traveling time between these cities is approximately 34 hours. Considering the solutions, we can conclude that routing of commodities in distant cities such as Hakkari to Edirne and Ordu to Antakya has very high impact on second objective.

Table 6-12 Routing scenarios for some source-destination pairs in the given Pareto optimal solutions

Source - Destination	Hubs(Figure 6-15)	Hubs(Figure 6-16)
Edirne - Hakkari	İstanbul → Gaziantep	İstanbul → Ankara
Hakkari - Edirne	Ankara → İstanbul	Ankara → İstanbul
Ardahan - İzmir	Ankara	Ankara
Tekirdag - Kilis	İstanbul → Gaziantep	İstanbul → Ankara
Kırklareli - Gaziantep	İstanbul	İstanbul → Ankara
Tekirdag - Tokat	İstanbul	İstanbul
Ordu - İzmir	Ankara	Ankara
Bilecik - Adıyaman	Ankara → Gaziantep	Ankara
Adıyaman - Bilecik	Ankara	Ankara
Ardahan - Kars	Gaziantep	Ankara
Ordu - Antakya	Gaziantep	Ankara
Mardin - Çanakkale	Gaziantep → Ankara	Ankara
Çanakkale - Mardin	Gaziantep	Ankara

7. CONCLUSION AND FUTURE WORK

This study presents a mathematical model for a multi-objective, capacitated, multiple allocation, and incomplete HLP (MOCMAHLP). In the problem, the aim is to minimize the total cost including total transportation costs and fixed costs and to minimize the maximum travel time required for routing the demands of all commodities. We develop a multi-objective approaches based on NSGA-II and AMOSA for decisions on the number of hubs, the location of hubs, the allocations of spokes to hubs, the hub network design, and the routing of all demands. A new solution representation and the appropriate crossover and mutation operator are proposed. The performance of our approach is tested on AP data set and Turkish data set. We explore the influence of the parameters, namely population size, and crossover and mutation probabilities, on the performance of the proposed NSGA-II based approach and number of arc mutations on the performance of the proposed AMOSA based approach. The results reveal that our approach is not very sensitive to these parameters.

There are two main contributions of this study:

- To our knowledge, this is the first study that proposes a mathematical model for MOCMAHLP. In MOCMAHLP, the hub network is not fully interconnected and hubs and arcs in the network have limited capacity.
 - The model includes location-allocation of hubs and spokes and routing of commodities between each pair of nodes in the network.
 - Two objectives, i.e., the minimization of transportation cost and the minimization of maximum travel time of routing the commodities, are considered.
- We develop a multi-objective approaches for the problem. Our approaches are capable of finding good solutions on large instances of the problem.

As future work, we will build a new model for the multi-period MOCMAHLP and design new techniques as an extension of our approach to handle the change in the environments.

8. CV

Personal Information

Name, Surname	İbrahim DEMİR
Address	Katip Nazmiler Cad. 28/B370 Merdivenköy/Kadıköy/İstanbul
Phone	0530 610 90 05
E-mail	ibrahim.demir@mail.com
Date of Birth	07.07.1989

Education

2007 – 2012 B.S. Electrical&Electronics Engineering at Anadolu University
GPA: 3.02/4.00 Eskisehir, Turkey

2009 – 2012 B.S. Computer Engineering at Anadolu University
GPA: 3.04/4.00 Eskisehir, Turkey

2016 – 2019 M.S. Computer Engineering at Marmara University
GPA: 3.20/4.00 Istanbul, Turkey

Work Experience

R&D Software Engineer in Paranavision (HQ:Turkey)

(July 2011-Aug 2012/ ~1 Year)

-Ankara/TURKEY

- Developing innovator image processing projects by using C++ and C# programming languages. Used technologies are: OpenCV, GStreamer, OpenEmbedded, CMake cross compiler, Beagleboard.

- Integrating Open source Developments into projects like opencv and gstreamer in embedded linux environment.

Software Engineer in Alcatel-Lucent(HQ:France&USA)

(March 2013-March2014/ 1 Year)

-Istanbul/TURKEY

- Development of delivery auditing software modules for ALU & Non-ALU IMS equipmentsfor ALU AT&T GSAT Project aligned with the Gold Standards of AT&Tby using Java.
- Development and maintenance of Turk Telekom Number Portability and MassIVR projects in Java EE.

Software Engineerin Wincor-Nixdorf(HQ:Germany)

(March 2014-Jan 2018 /~4Years)

-Istanbul/TURKEY

- Preparation of Pre-sales activities and POC for Ziraatbank end-to-end project tender which won by Diebold-Nixdorf later.
- Development of phases 0 and 1 of Ziraatbank Proflex4 project with developing complete SOP and palm vein reader with their transactions in Java and C++ languages.
- Development and Consultancy for PS Banking Department to support Banks having various software levels such as XFS(eXtended Financial Services), Protopas(Diebold-Nixdorf's XFS Middleware Framework) and Procash NDC(Diebold-Nixdorf's terminal solution).

Customer Solutions Manager in Keba (HQ:Austria)

(Feb 2018-Feb2019 / ~ 1 Year)

-Istanbul/TURKEY

- Managed the relations with customers and provide solutions with our technical team.

Software Engineer in KG Technology

(Mar 2018-Now / ~ 5Months)

-Istanbul/TURKEY

- Development of credit card store project with using Spring boot and Spring batch at Aksigorta.
- Development of guvencell insurance project with using Spring boot in microservices architecture at Turkcell.

9. REFERENCES

- [1] A. T. Ernst and M. Krishnamoorthy, "Efficient algorithms for the uncapacitated single allocation p-hub median problem," *Location Science*, vol. 4, no. 3, pp. 139–154, oct 1996.
- [2] A.T. Ernst and M.Krishnamoorthy, "An exact solution approach based on shortest-paths for p-hub median problems," *INFORMS Journal on Computing*, 1998.
- [3] M. E. O'Kelly, "A quadratic integer program for the location of interacting hub facilities," *European Journal of Operational Research*, vol. 32, no. 3, pp. 393–404, dec 1987.
- [4] A. T. Ernst and M. Krishnamoorthy, "Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem," *European Journal of Operational Research*, 1998.
- [5] V. Sridhar and J. S. Park, "Benders-and-cut algorithm for fixed-charge capacitated network design problem," *European Journal of Operational Research*, 2000.
- [6] I. Rodríguez-Martín and J. J. Salazar-González, "Solving a capacitated hub location problem," *European Journal of Operational Research*, vol. 184, no. 2, pp. 468–479, 2008.
- [7] J. F. Campbell, "Location and allocation for distribution systems with transshipments and transportation economies of scale," *Annals of Operations Research*, 1992.
- [8] D. Skorin-Kapov, J. Skorin-Kapov, and M. O'Kelly, "Tight linear programming relaxations of uncapacitated p-hub median problems," *European Journal of Operational Research*, 1996.
- [9] H. Topcuoglu, F. Corut, M. Ermis, and G. Yilmaz, "Solving the uncapacitated hub location problem using genetic algorithms," *Computers & Operations Research*, vol. 32, no. 4, pp. 967–984, apr 2005.
- [10] J. F. Campbell, A. T. Ernst, and M. Krishnamoorthy, "Hub arc location problems: Part I - Introduction and results," *Management Science*, vol. 51, no. 10, pp. 1540–1555, 2005.
- [11] M. E. O'Kelly and H. J. Miller, "The hub network design problem. A review and synthesis," *Journal of Transport Geography*, 1994.

- [12] B. Kara and B. Tansel, "On the allocation phase of the p-hub location problem." Department of Industrial Engineering, Bilkent University, Tech. Rep., 1998.
- [13] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [14] S. Alumur and B. Y. Kara, "Network hub location problems: The state of the art," *European Journal of Operational Research*, vol. 190, no. 1, pp. 1–21, Oct 2008.
- [15] I. Contreras, "Hub Location Problems," in *Location Science*. Cham: Springer International Publishing, pp. 311–344, 2015.
- [16] A. Zabihi and M. Gharakhani, "A literature survey of HUB location problems and methods with emphasis on the marine transportations," *Uncertain Supply Chain Management*, pp. 91–116, 2018.
- [17] M. Mirzaei and M. Bashiri, "Multiple objective multiple allocation hub location problem," in *40th International Conference on Computers and Industrial Engineering: Soft Computing Techniques for Advanced Manufacturing and Service Systems, CIE40 2010*, 2010.
- [18] L. Tavakoli, N. Manavizadeh, and M. Rabbani, "Multi-objective multiple allocation hub location problem with multiple capacity levels considering sustainable development paradigm," *International Journal of Applied Decision Sciences*, 2014.
- [19] M. Köksalan and B. Soylu, "Bicriteria p-hub location problems and evolutionary algorithms," *INFORMS Journal on Computing*, 2010.
- [20] M. da Graça Costa, M. E. Captivo, and J. Clímaco, "Capacitated single allocation hub location problem-A bi-criteria approach," *Computers and Operations Research*, vol. 35, no. 11, pp. 3671–3695, 2008.
- [21] A. Ghodratinama, R. Tavakkoli-Moghaddam, and A. Azaron, "Robust and fuzzy goal programming optimization approaches for a novel multiobjective hub location-allocation problem: A supply chain overview," *Applied Soft Computing Journal*, vol. 37, pp. 255–276, 2015.

- [22] F. Kaveh, R. Tavakkoli-Moghaddam, A. Jamili, and M. Eghbali, "Design of a Bi-Objective Capacitated Single-Allocation Incomplete Hub Network Considering an Elastic Demand," *International Journal of Industrial Engineering & Production Research*, vol. 27, no. 4, pp. 373–385, Dec 2016.
- [23] H. Karimi and M. Setak, "A bi-objective incomplete hub location-routing problem with flow shipment scheduling," *Applied Mathematical Modelling*, vol. 57, pp. 406–431, 2018.
- [24] A. Ghodratinama, H. R. Arbabi, and A. Azaron, "A bi-objective hub location-allocation model considering congestion," *Operational Research*, pp. 1–40, 2018.
- [25] R. Z. Farahani, M. Hekmatfar, A. B. Arabani, and E. Nikbakhsh, "Hub location problems: A review of models, classification, solution techniques, and applications," *Computers and Industrial Engineering*, vol. 64, no. 4, pp. 1096–1109, 2013.
- [26] Q. Yuchi, N. Wang, S. Li, Z. Yang, and B. Jiang, "A Bi-objective Reverse Logistics Network Design under the Emission Trading Scheme," *IEEE Access*, pp. 1–1, 2019.
- [27] M. Eghbali, M. Abedzadeh, and M. Setak, "Multi-objective reliable hub covering location considering customer convenience using NSGA-II," *International Journal of System Assurance Engineering and Management*, vol. 5, no. 3, pp. 450–460, 2014.
- [28] A. Ebrahimi Zade, A. Sadegheih, and M. M. Lotfi, "A modified NSGAII solution for a new multi-objective hub maximal covering problem under uncertain shipments," *Journal of Industrial Engineering International*, vol. 10, no. 4, pp. 185–197, 2014.
- [29] R. Bhattacharya and S. Bandyopadhyay, "Solving conflicting bi-objective facility location problem by NSGA II evolutionary algorithm," *The International Journal of Advanced Manufacturing Technology*, vol. 51, no. 1, pp. 397–414, 2010.
- [30] R. Z. Farahani, M. SteadieSeifi, and N. Asgari, "Multiple criteria facility location problems: A survey," *Applied Mathematical Modelling*, vol. 34, no. 7, pp. 1689–1709, 2010.
- [31] S. A. Alumur, B. Y. Kara, and O. E. Karasan, "The design of single allocation incomplete hub networks," *Transportation Research Part B: Methodological*, vol. 43, no. 10, pp. 936–951, 2009.

- [32] S. Çetiner, C. Sepil, and H. Süral, "Hubbing and routing in postal delivery systems," *Annals of Operations Research*, vol. 181, no. 1, pp. 109–124, Dec 2010.
- [33] P. Z. Tan and B. Y. Kara, "A hub covering model for cargo delivery systems," *Netw.*, vol. 49, no. 1, pp. 28–39, Jan. 2007.
- [34] E. Zitzler, D. Brockhoff, and L. Thiele, "The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration" Tech. Rep, 2007.
- [35] S. Bandyopadhyay, S. Saha, U. Maulik, K. Deb, "A simulated annealing-based multiobjective optimization algorithm: AMOSA", *IEEE Trans. Evol. Comput.*, vol. 12, no. 3, pp. 269-283, 2008.
- [36] <http://users.monash.edu/~andreas/Downloads>
- [37] R. Akbari, R. Hedayatzadeh, K. Ziarati, B. Hassanizadeh, "A multi-objective artificial bee colony algorithm", *Swarm Evol Comput* 2:39–52, 2012.
- [38] F. Niakan, B. Vahdani, M. Mohammadi, "A multi-objective optimization model for hub network design under uncertainty: An inexact rough-interval fuzzy approach", *Engineering Optimization*, 47(12), pp.1670-1688, 2015.
- [39] M. Musavi, A. Bozorgi-Amiri, "A multi-objective sustainable hub location-scheduling problem for perishable food supply chain", *Comput. Ind. Eng.* 113, 1–13. <https://doi.org/10.1016/j.cie.2017.07.039>, 2017.
- [40] F. Parvaresh, S.M. Moattar Hussein, S.A. Hashemi Golpayegany, B. Karimi, "Hub network design problem in the presence of disruptions", *Journal of Intelligent Manufacturing*, vol. 25, pp. 755, 2014.