



YAŞAR UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

PH.D. THESIS

**EVOLUTIONARY ALGORITHMS FOR SOLVING A
MULTI-OBJECTIVE GREEN VEHICLE ROUTING
PROBLEM**

KAZIM ERDOĞDU

THESIS ADVISOR: ASST. PROF. DR. KORHAN KARABULUT

COMPUTER ENGINEERING

PRESENTATION DATE: 11.06.2019

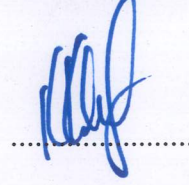
BORNOVA / İZMİR
JUNE 2019

We certify that, as the jury, we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of the Doctor of Philosophy.

Jury Members:

Asst. Prof. Dr. Korhan KARABULUT
Yaşar University

Signature:



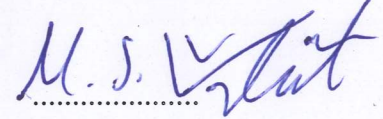
Asst. Prof. Dr. Dindar ÖZ
Yaşar University



Prof. Dr. Aybars UĞUR
Ege University



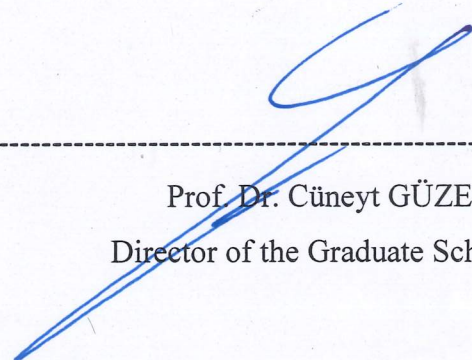
Prof. Dr. Mehmet Süleyman ÜNLÜTÜRK
Yaşar University



Asst. Prof. Dr. Mete EMİNAĞAOĞLU
Dokuz Eylül University



Prof. Dr. Cüneyt GÜZELİŞ
Director of the Graduate School



ABSTRACT

EVOLUTIONARY ALGORITHMS FOR SOLVING A MULTI-OBJECTIVE GREEN VEHICLE ROUTING PROBLEM

Erdođdu, Kazım

Ph.D., Computer Engineering

Advisor: Asst. Prof. Korhan KARABULUT

June 2019

Green Vehicle Routing Problems (GVRPs) increasingly gain prominence due to the environmental issues created by the transportation vehicle fleets. The amount of CO_2 emissions caused by the fossil fuel vehicles can be decreased by reducing the amount of fuel consumption of these vehicles. In this thesis, a Multi-Objective Green Vehicle Routing Problem (MOGVRP) was studied. Two objectives were taken into consideration in the problem: minimizing the total distance and minimizing the total fuel consumption of all vehicle routes. Two state-of-the-art methods NSGA-II and ϵ -MOEA were adapted and applied for the solution of the problem, a multi-objective local search heuristic was proposed, and Path-Relinking heuristic was modified for the multi-objective problem.

Key Words: Green Vehicle Routing Problems, Multi-Objective Problems, Evolutionary Algorithms.

ÖZ

ÇOK AMAÇLI BİR YEŞİL ARAÇ ROTALAMA PROBLEMİNİN ÇÖZÜMÜ İÇİN EVRİMSEL ALGORİTMALAR

Erdoğan, Kazım

Doktora Tezi, Bilgisayar Mühendisliği

Danışman: Dr. Öğr. Üyesi Korhan KARABULUT

Haziran 2019

Yeşil Araç Rotalama Problemleri, nakliye araç filolarının sebep olduğu çevre kirliliklerinden ötürü artan ölçüde önem kazanmaktadır. Fosil yakıtlı araçların yaydıkları CO_2 miktarı bu araçların yakıt tüketimlerinin azaltılmasıyla daha az bir seviyeye indirgenebilir. Bu tez çalışmasında, Çok Amaçlı Yeşil Araç Rotalama Problemi üzerinde çalışılmıştır. Problem içerisinde iki amaç fonksiyonu kullanılmıştır: araçların toplam kat ettiği mesafenin en aza indirgenmesi ve toplam yakıt tüketiminin en aza indirgenmesi. Çözüm için bilinen yöntemlerden olan NSGA-II and ϵ -MOEA uyarlanarak uygulanmış, yeni bir çok amaçlı yerel arama sezgisel yöntemi önerilmiş ve de Path-Relinking sezgisel yöntemi çok amaçlı problem yapısına göre uyarlanmıştır.

Anahtar Kelimeler: Yeşil Araç Rotalama Problemleri, Çok Amaçlı Problemler, Evrimsel Algoritmalar.

ACKNOWLEDGEMENTS

First, I would like to thank my supervisor Asst. Prof. Dr. Korhan KARABULUT for his guidance, helps and patience during this study.

I would also like to express my enduring love to my family, who are always supportive, loving and caring to me in every possible way in my life.

Kazım ERDOĞDU

İzmir, 2019



TEXT OF OATH

I declare and honestly confirm that my study, titled “EVOLUTIONARY ALGORITHMS FOR SOLVING A MULTI-OBJECTIVE GREEN VEHICLE ROUTING PROBLEM” and presented as a PhD Thesis, has been written without applying to any assistance inconsistent with scientific ethics and traditions. I declare, to the best of my knowledge and belief, that all content and ideas drawn directly or indirectly from external sources are indicated in the text and listed in the list of references.

Kazım ERDOĞDU

Signature



June 18, 2019

TABLE OF CONTENTS

ABSTRACT	v
ÖZ	vii
ACKNOWLEDGEMENTS	ix
TEXT OF OATH	xi
TABLE OF CONTENTS	xiii
LIST OF FIGURES	xv
LIST OF TABLES	xix
ABBREVIATIONS	xxi
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 VEHICLE ROUTING PROBLEM	4
2.1. Variants of the VRP	5
CHAPTER 3 MULTI-OBJECTIVE OPTIMIZATION PROBLEMS	7
3.1. Mathematical Model of MOOPs	7
3.2. Pareto-Optimality	8
3.3. Solution Methods for MOOPs.....	10
3.4. Classical Solution Methods for MOOPs	10
3.4.1. Weighted Sum Method	11
3.4.2. ϵ -Constraint Method	13
3.4.3. Weighted Metric Methods	14
3.5. Evolutionary Algorithms.....	16
3.5.4. Non-Dominated Sorting Genetic Algorithm II (NSGA-II).....	18
3.5.5. ϵ -based Multi-Objective Evolutionary Algorithm (ϵ -MOEA)	20
3.6. Performance Metrics	23
3.6.1. Hypervolume (HV)	23
3.6.2. Generational Distance (GD).....	24
3.6.3. Inverse Generational Distance (IGD).....	25
CHAPTER 4 EVOLUTIONARY ALGORITHMS FOR SOLVING A MULTI-OBJECTIVE GREEN VEHICLE ROUTING PROBLEM	27
4.1. Definition of the Problem.....	27
4.2. Literature Review	28
4.3. The Mathematical Model of the MOGVRP	31

4.4. Solution Representation	34
4.5. Constructive Heuristics	37
4.6. Local Search Heuristics	40
4.7. Path-Relinking Heuristic.....	43
4.8. Reproduction Operators	45
4.9. NSGA-II and ϵ -MOEA Used in the Thesis	47
CHAPTER 5 EXPERIMENTAL STUDIES	50
5.1. Results with Giant Tour Solution Representation.....	52
5.2. Results with Split Vehicle Tour Solution Representation	56
5.3. Visualization of the Results	73
CONCLUSIONS AND FUTURE RESEARCH	76
REFERENCES	79
APPENDIX 1 – Pareto Front Results for the Giant Tour Representation	83
APPENDIX 2 – Pareto Front Results for the Split Vehicle Tour Representation	91

LIST OF FIGURES

Figure 1.1 EPA Report on GHG Emissions in the USA in 2016 (EPA, 2018).....	1
Figure 1.2 EEA Report on GHG Emissions in Europe from 1990 to 2015 (EEA, 2018).....	2
Figure 1.3 TÜİK Report on GHG Emissions in Turkey from 1990 to 2012 (TUIK, 2012) ...	2
Figure 1.4 CO ₂ emission distribution according to vehicle types in Turkey in 2012 (TUIK, 2012)	3
Figure 2.1. Vehicle Routing Problem	4
Figure 3.1. Set of Objective Functions (Zitzler, Laumanns, & Bleuler, 2004).....	8
Figure 3.2. Dominated and Non-dominated Solutions (Araujo, Poldi, & Smith, 2014).....	9
Figure 3.3 Weighted Sum Method on a Convex Feasible Solution Space (Deb, 2001)	12
Figure 3.4 Weighted Sum Method on a Non-convex Feasible Solution Space (Deb, 2001)	13
Figure 3.5 ϵ -Constraint Method (Deb, 2001)	14
Figure 3.6 (a) Weighted Metric Methods: (a) when $p = 1$, (b) when $p = 2$ (Deb, 2001)	15
Figure 3.7 Pseudo-code for EAs	17
Figure 3.8 A sample for the fronts of a MOOP	19
Figure 3.9 Crowding distance in NSGA-II (Deb, Mohan, & Mishra, 2003)	19
Figure 3.10 Pseudo-code for NSGA-II	20
Figure 3.11 Hyper-boxes in ϵ -MOEA (Deb et al., 2003)	21
Figure 3.12 Pseudo-code for ϵ -MOEA	23
Figure 3.13 Hypervolume for a minimization MOOP with two objectives.....	24
Figure 3.14 Generational Distance for a MOOP with two objectives (Sheng et al., 2014) ..	25
Figure 3.15 Inverted Generational Distance for a MOOP with two objectives (Ishibuchi, Masuda, & Nojima, 2016).....	26
Figure 4.1 Bellman Split Algorithm: Splitting Procedure (Prins, 2004).....	35
Figure 4.2 Bellman Split Algorithm: Extracting Vehicles Procedure (Prins, 2004).....	35
Figure 4.3 Bellman Split Example.....	36
Figure 4.4 2-Opt example	41

Figure 4.5 Cross Exchange (Taillard et al., 1997).....	42
Figure 4.6 Path Relinking (Zhang, Bai, & Dong, 2010).....	44
Figure 4.7 Pseudocode for adapted PR with MOP structure	45
Figure 4.8 Pseudo-code of the adapted NSGA-II for the MOGVRP	48
Figure 4.9 Pseudo-code of the adapted ϵ -MOEA for the MOGVRP	49
Figure 5.1 Golden instance sets (Xavier et al., 1998).....	51
Figure 5.2 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT1	52
Figure 5.3 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT2	53
Figure 5.4 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT3	53
Figure 5.5 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT4	53
Figure 5.6 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT5	54
Figure 5.7 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT11	54
Figure 5.8 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT12	54
Figure 5.9 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_1	55
Figure 5.10 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_5	55
Figure 5.11 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_9	55
Figure 5.12 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_13	56
Figure 5.13 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_17	56

Figure 5.14 HV values of the DoE for the adapted NSGA-II.....	57
Figure 5.15 GD values of the DoE for the adapted NSGA-II.....	58
Figure 5.16 IGD values of the DoE for the adapted NSGA-II.....	58
Figure 5.17 Interaction of DoE parameters according to HV for the adapted NSGA-II	58
Figure 5.18 Interaction of DoE parameters according to GD for the adapted NSGA-II	59
Figure 5.19 Interaction of DoE parameters according to IGD for the adapted NSGA-II.....	59
Figure 5.20 HV values of the DoE for the adapted ϵ -MOEA.....	60
Figure 5.21 GD values of the DoE for the adapted ϵ -MOEA.....	60
Figure 5.22 IGD values of the DoE for the adapted ϵ -MOEA	60
Figure 5.23 Interaction of DoE parameters according to HV for the adapted ϵ -MOEA	61
Figure 5.24 Interaction of DoE parameters according to GD for the adapted ϵ -MOEA	61
Figure 5.25 Interaction of DoE parameters according to IGD for the adapted ϵ -MOEA.....	61
Figure 5.26 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT1	62
Figure 5.27 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT2	62
Figure 5.28 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT3	63
Figure 5.29 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT4	63
Figure 5.30 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT5	63
Figure 5.31 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT11	64
Figure 5.32 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT12	64
Figure 5.33 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_1	64
Figure 5.34 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_2	65

Figure 5.35 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_3.....	65
Figure 5.36 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_4.....	65
Figure 5.37 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_5.....	66
Figure 5.38 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_6.....	66
Figure 5.39 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_7.....	66
Figure 5.40 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_8.....	67
Figure 5.41 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_9.....	67
Figure 5.42 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_10.....	67
Figure 5.43 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_11.....	68
Figure 5.44 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_12.....	68
Figure 5.45 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_13.....	68
Figure 5.46 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_14.....	69
Figure 5.47 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_15.....	69
Figure 5.48 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_16.....	69
Figure 5.49 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_17.....	70
Figure 5.50 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_18.....	70

Figure 5.51 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_19	70
Figure 5.52 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_20	71
Figure 5.53 MOGVPR GUI with JAVA – Instances.....	74
Figure 5.54 MOGVPR GUI with JAVA – Solutions.....	74
Figure 5.55 MOGVPR GUI with JAVA – Individual Vehicle Tours.....	75

LIST OF TABLES

Table 5-1 Performance Measures of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour on all Christofides and Golden Instances	73
--	----

ABBREVIATIONS

2D	Two dimensional
AHLF	Assigning Heaviest Load First
ALNS	Adaptive Large Neighborhood Heuristic Algorithm
ATCS	Apparent Tardiness Cost with Setups
C&W Savings	Clark and Wright Savings
CH ₄	Methane
CLRSPOA	Cumulative Location-Routing and Speed Optimization Algorithm
CO	Carbon Monoxide
CO ₂	Carbon Dioxide
CVRP	Capacitated Vehicle Routing Problem
CX	Cycle Crossover
DoE	Design of Experiment
EA	Evolutionary Algorithm
EEA	European Environment Agency
ELV	Emptying the Lighter Vehicles
EP	Evolutionary Programming
EPA	Environmental Protection Agency
ES	Evolutionary Strategy
E-TVRP	Emission-based Time-Dependent Vehicle Routing Problem
FCR	Fuel Consumption Rate
FCVRP	Fuel Consumption Rate on VRP
GA	Genetic Algorithm
GD	Generational Distance
GHG	Greenhouse Gas

GLRP	Green Location-Routing Problem
GP	Genetic Programming
GUI	Graphical User Interface
GVRP	Green Vehicle Routing Problem
HV	Hypervolume
HVRP	Heterogeneous Vehicle Routing Problem
IGD	Inverted Generational Distance
MCBWW	Moving Customers Between and Within Vehicles
MDVRP	Multi Depot Vehicle Routing Problem
MOEA	Multi-Objective Evolutionary Algorithms
MOGA	Multi-Objective Genetic Algorithm
MOGVRP	Multi-Objective Green Vehicle Routing Problem
MOOP	Multi-Objective Optimization Problem
MOP	Multi-Objective Problem
MOVSRP	Multi-Objective Vehicle Routing Problem
mTSP	Multiple Travelling Salesman Problem
NO _x	Nitrogen Oxides
NPGA	Niched Pareto Genetic Algorithm
NSGA	Non-dominated Sorting Genetic Algorithm
NSGA-II	Fast and Elitist Non-dominated Sorting Genetic Algorithm II
OX	Ordered Crossover
PAES	Pareto Archived Evolutionary Strategy
PDVRP	Pickup and Delivery Vehicle Routing Problem
PMX	Partially Mapped Crossover
PR	Path Relinking
PRP	Pollution Routing Problem

PRPTW	Pollution Routing Problem with Time Windows
RT	Random Task
RTF	Random Task Flower
SDVRP	Split Delivery Vehicle Routing Problem
SMSA	String-model-based Simulated Annealing
SOP	Single Objective Problem
SOVRP	Single Objective Vehicle Routing Problem
SPEA	Strength Pareto Evolutionary Algorithm
SVRP	Stochastic Vehicle Routing Problem
TDVRP	Time-Dependent Vehicle Routing Problem
TPX	Two-point Crossover
TSP	Travelling Salesman Problem
TÜİK	Türkiye İstatistik Kurumu
VEGA	Vector Evaluated Genetic Algorithm
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows
ϵ -MOEA	ϵ -based Multi-Objective Evolutionary Algorithm

CHAPTER 1

INTRODUCTION

Vehicle Routing Problems (VRPs) are one of the widely occurring problems in distribution and logistics fields. As commerce grows all around the world each day, the transportation of commercial products from the warehouses to the customers plays a more critical role for the commercial and transportation companies. In order to maximize their gain and minimize their costs, they need to arrange the routes of their vehicle fleets in the best possible way. This fact makes VRPs very relevant to real-life optimization problems, especially in the transportation field.

Although there are numerous variants of VRP, Green VRPs (GVRPs) have begun to attract many researchers for environmental reasons. The increase in the number of vehicles running on fossil fuel has been increasing the level of emissions of air pollutants all around the world. The measurements on Greenhouse Gas (GHG) Emissions in different countries demonstrate the seriousness of the problem.

According to the 2016 report of the United States Environmental Protection Agency, 83% of the GHG emissions in transportation was caused by light-, medium-, and heavy-duty trucks. 96.7% of total emitted GHG is CO₂ (Figure 1.1). GHG emissions caused by light-duty trucks was increased by 14.4% and GHG caused by medium- and heavy-duty trucks was increased by 84.9% from 1990 to 2016 (EPA, 2018).

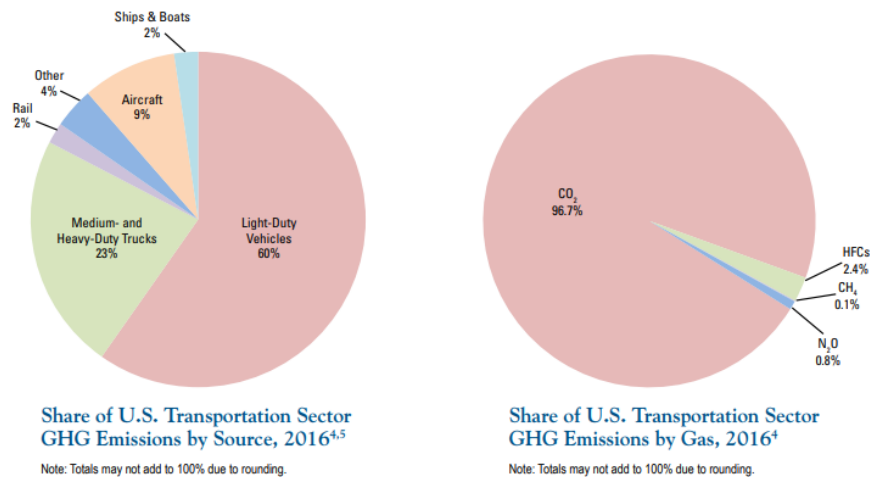
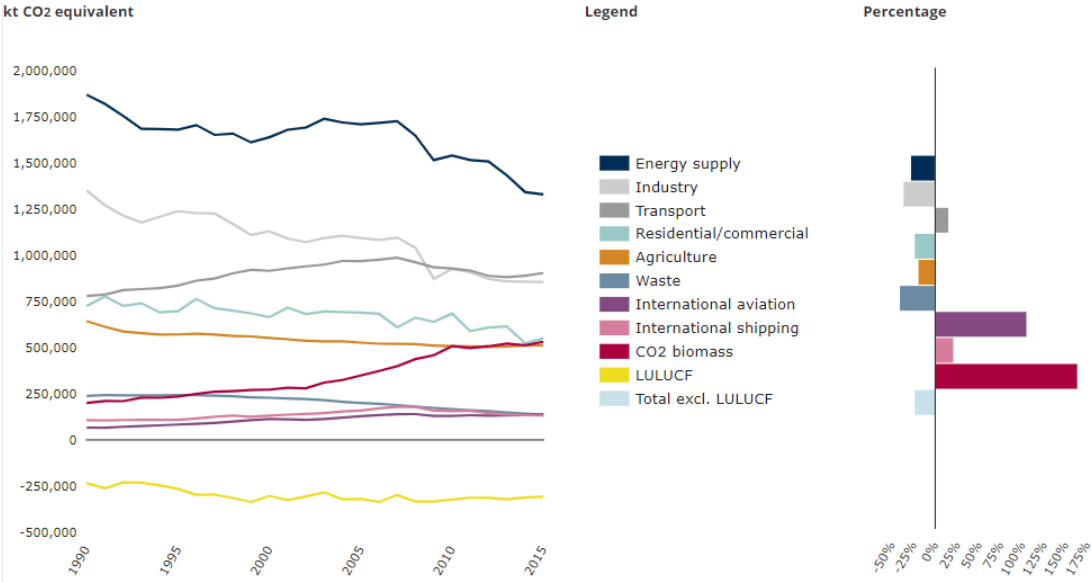


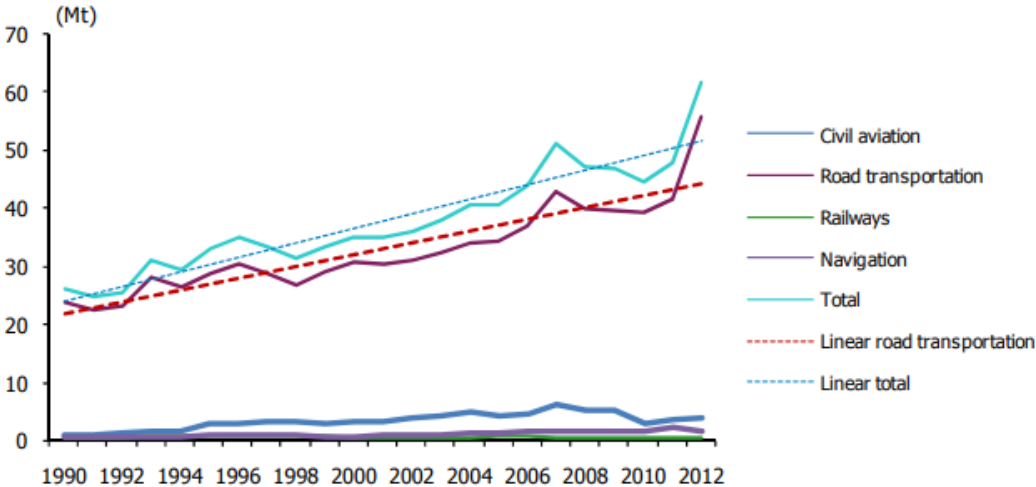
Figure 1.1 EPA Report on GHG Emissions in the USA in 2016 (EPA, 2018)

According to the 2017 report of the European Environment Agency, heavy-duty vehicles are responsible for 27 % of road transport CO₂ emissions. Since 1990, heavy-duty vehicle emissions have increased by 25 % mainly as a result of the increase in road freight traffic (Figure 1.2) (EEA, 2018).



According to TÜİK report on GHG emissions in Turkey 2012, 90.5% is caused by road transportation (Figure 1.3). 39.6% of this road transportation is done by light- and heavy-duty trucks (Figure 1.4). The amount of CO₂ emitted by all road transportation was increased by 129.22% from 1990 to 2012 (TUIK, 2012).

3.6 CO₂ emission trend in modes of transport, 1990 - 2012



3.16 CO₂ equivalent distributions with respect to types of vehicle, 2012

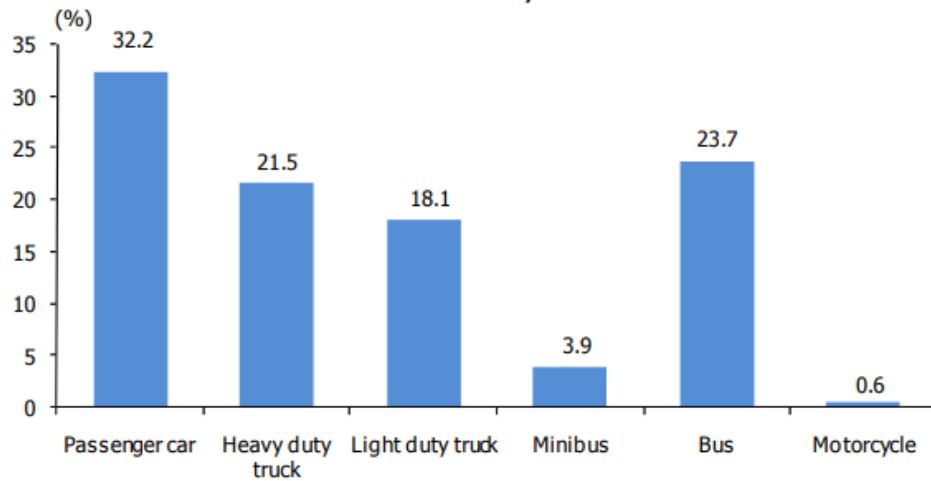


Figure 1.4 CO₂ emission distribution according to vehicle types in Turkey in 2012 (TUIK, 2012)

As it is seen in the statistical data on GHG emissions in different continents, road transportation vehicle (especially the trucks) cause great damages to environment. The GHGs must be reduced. One of the ways for reducing this emission is to arrange the routes of those vehicles in a way to minimize their fuel consumption. And that is the focus of GVRP.

VRPs can be either Single Objective Problems (SOPs) or Multi-Objective Problems (MOPs). SOPs, hence the name, focus on optimizing one objective while MOPs aim to optimize more than one objective at the same time. Most of the time, these objectives fully or to some degree are in conflict with one another. MOPs are more difficult and realistic problems compared to SOPs. Furthermore, while SOPs try to find the single best solution for the problem, MOPs aim to provide multiple and alternative trade-off solutions.

Multi-Objective Green Vehicle Routing Problem (MOGVRP) was studied in this thesis. In Chapter 2, VRPs were defined formally and their variants were discussed. In Chapter 3, MOPs were defined and explained. In addition, solution methods for MOPs were discussed and performance metrics that are used to evaluate the algorithms for solving MOPs were explained. In Chapter 4, the MOGVRP studied in this thesis and the solution methods for this MOGVRP were explained in detail. In Chapter 5, the experimental results were presented and analyzed. And in the last chapter, this thesis study was evaluated, and some future works were mentioned.

CHAPTER 2

VEHICLE ROUTING PROBLEM

VRP is a combinatorial optimization problem which aims to find the minimum cost for a fleet of vehicles in delivering their products to certain customers according to the demands of the customers. In a classical VRP, there is a fleet of capacitated vehicles, a depot, a set of customers and their demands. Each customer is assigned to a vehicle in the fleet until there is no customer left out and while the total load of the demand does not exceed the vehicle capacities. Each vehicle needs to accomplish its tour by leaving the depot loaded with the demands of its customers, delivering these demands to the corresponding customers by visiting them only once, and returning to the depot after all the demands are delivered. The objective of VRP is to minimize the total cost of all vehicle tours. This cost can be the total sum of the distances of vehicle tours, or the total number of vehicles in order to deliver all demands, the total fuel consumption or gas emission of the vehicles, etc. VRP contains the classical Travelling Salesman Problem (TSP). In each vehicle tour, the VRP searches the best TSP solution by starting and ending with depot under certain constraints (Figure 2.1).

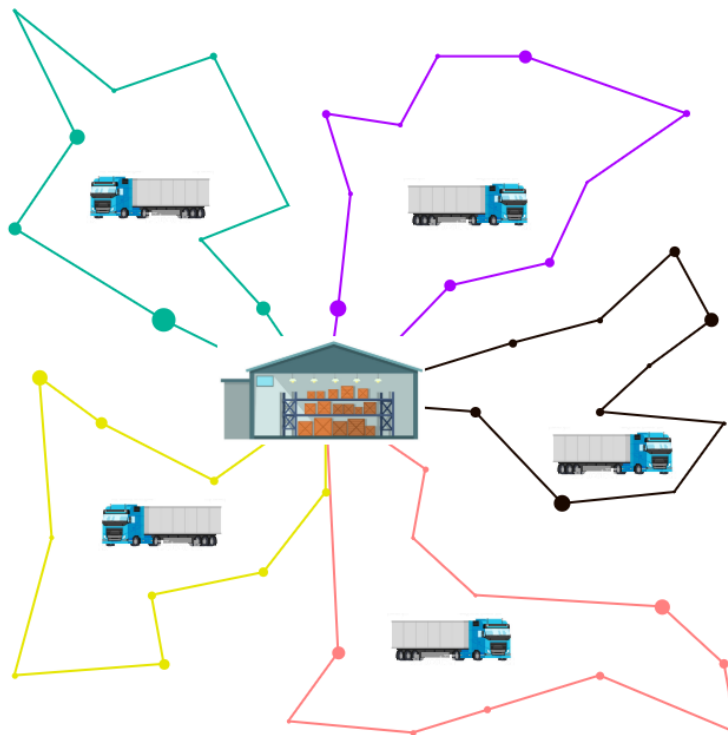


Figure 2.1. Vehicle Routing Problem

2.1. Variants of the VRP

The pioneer studies of VRP began with the paper of George Dantzig and John Ramser in 1959. In their paper, they laid a foundation of mathematical programming and algorithmic approach for VRP in delivering fuel to certain gas stations. Since then, there have been numerous studies in VRP which led the variations of the problem (Carić & Gold, 2008). These variations are related to the capacity of the vehicles, the number of depots, quantity and the quality of the objectives of the problem, dynamic or static locations of the customers, customers' availability time windows, delivering and or picking up the demands of the customers, etc.

In homogeneous VRPs, the vehicles have homogenous capacities wherein heterogeneous VRPs, these capacities are heterogeneous (HVRP). VRPs are generally considered as homogeneous VRPs unless they are defined otherwise. In a classical VRP, there is only one depot. There, however, are VRPs that have multiple depots. The demands of the customers are stored in different depots (MDVRP). Sometimes, the demands are not delivered to the customers all at once. Their demands can be divided into smaller amounts and these smaller demands can be delivered to the customers either by the same or different vehicles. Therefore, a customer can be visited more than once. This type of VRPs is called Split Delivery VRP (SDVRP).

In some VRPs, the demands of the customers must be delivered at certain time intervals. This type of VRPs is called VRP with Time Windows (VRPTW). There are two types of time windows: hard and soft. In hard time windows, all the vehicles must visit their customers within the customers' availability time windows. If the vehicles are early, then they need to wait. If they arrive late, then the customer's demand is not satisfied, therefore the solution is not feasible. In soft time windows, even if the vehicles visit their customers before or after the time interval, it is still considered as a feasible solution. Yet, a penalty is applied to this situation. In Pickup and Delivery VRPs (PDVRPs), while vehicles deliver the demands of the customers from the depot to the customers, they may also pick up goods along the way from the customers and deliver them to the other customers or return them to the depot.

Sometimes the products can be picked up from a single location or from multiple locations before they are delivered to a certain customer. The special case for PDVRPs is called "dial-a-ride problem" in which the goal is to provide transportation to elderly

and handicapped people from one location to another via a fleet of vehicles. In this version of PDVRP, there is no demand of the customers. Vehicle capacities are evaluated in terms of the number of their customers. In certain situations, time windows are also included in the problem.

There are also some VRPs that include some random characteristics. These VRPs are called Stochastic VRPs (SVRPs). In SVRPs, either the demands, customers, or times can be stochastic. The volume or amount of the demands of the customers may be known with a probability, the customers might be present or absent by the time a vehicle visits them, or both the service times at the customers' locations or traveling times of vehicles may be stochastic.

In Dynamic VRPs, the information about the customers, their locations, their demands, travel times, etc. are not known partially or totally prior to the beginning of vehicle tours, and yet they become available over time. Since this information changes over time in this type of problems, the exact information can be deterministically or stochastically obtained in time.

There are also VRPs that focus on the environmental side of the VRP. This type of VRPs is called Green VRP (GVRP). In GVRPs, the main objective is to minimize the level of pollutants that are produced during the routes of the vehicles (Toth & Vigo, 2014).

There are more variants of VRP, and these variants are extended with recent studies in order to model new problems. Most of them are either a subcategory of these variants, or a combination of these variants. Only few of them are totally new variants. However, only the major variants of VRP are mentioned in this thesis. The problem that was studied in this thesis is in the category of GVRP.

CHAPTER 3

MULTI-OBJECTIVE OPTIMIZATION PROBLEMS

Multi-Objective Optimization Problems (MOOPs) are the optimization problems that aim to find the optimum solutions for more than one objective function. Most of the time, these objectives conflict with one another, either partially or fully. In this sense, they are more realistic than Single Objective Problems (SOPs) in which there is only one objective function to be optimized. Most real-world optimization problems are MOOPs by their nature. For example, in a multi-objective version of a Vehicle Routing Problem (VRP), the objective of minimizing the total distance conflict with the objective of minimizing travel cost (i.e. cost of fuel, vehicle number, drivers' cost, taxes, etc.). In this sense, minimizing the total travel distance may lead to minimization of the number of vehicles but it will cause each vehicle to carry more load. This, then, will increase the fuel consumption of each vehicle. Therefore, when dealing with the optimum solutions for the problem, both objectives must be taken into consideration.

3.1. Mathematical Model of MOOPs

The mathematical model of a general MOOP contains a set of decision variables, a set of objective functions and a set of constraints as follows (Zitzler, 1999).

Objective functions:

$$\text{optimize } y = f(x) = (f_1(x), f_2(x), \dots, f_k(x)) \quad (1)$$

Constraints:

$$\text{subject to } e(x) = (e_1(x), e_2(x), \dots, e_m(x)) \geq 0 \quad (2)$$

Inputs:

$$\begin{aligned} \text{where } x &= (x_1, x_2, \dots, x_n) \in X \text{ (decision variables)} \\ y &= (y_1, y_2, \dots, y_k) \in Y \text{ (objective functions)} \end{aligned} \quad (3)$$

In this mathematical model of MOOP, x_i are the decision variables, e_i are the constraints of the problem and $y_i = f(x_i)$ are the objective functions. The function

For the minimizing objective functions,

For $u, v \in X$ and $f \in Y$ where $f = (f_1, f_2, \dots, f_k)$

$u < v$ (u dominates v) $\Leftrightarrow \forall f_i \in f \ f_i(u) < f_i(v)$

$u \preceq v$ (u weakly dominates v) $\Leftrightarrow \forall f_i \in f \ f_i(u) \leq f_i(v)$ (4)

$u \sim v$ (u and v are non – dominated) \Leftrightarrow

$\exists f_i, f_j \in f$ s.t. $f_i(u) < f_i(v)$ and $f_j(v) < f_j(u)$

For the maximizing objective functions,

For $u, v \in X$ and $f \in Y$ where $f = (f_1, f_2, \dots, f_k)$

$u > v$ (u dominates v) $\Leftrightarrow \forall f_i \in f \ f_i(u) > f_i(v)$

$u \succeq v$ (u weakly dominates v) $\Leftrightarrow \forall f_i \in f \ f_i(u) \geq f_i(v)$ (5)

$u \sim v$ (u and v are non – dominated) \Leftrightarrow

$\exists f_i, f_j \in f$ s.t. $f_i(u) > f_i(v)$ and $f_j(v) > f_j(u)$

As a result, if the decision variable vector of x (i.e. a solution) cannot be dominated by other decision variable vectors (solutions), then this solution vector is called a Pareto-optimal solution. The set of all Pareto-optimal solutions are called the Pareto-optimal set or Pareto front. Pareto front of a MOOP contains the globally optimal solutions for the problem. A general graphic for bi-objective minimization problems are shown in Figure 3.2.

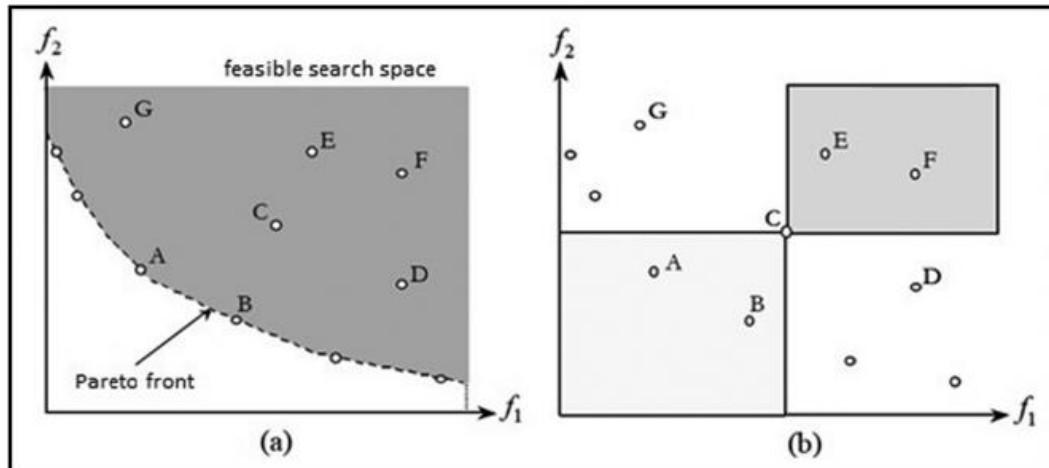


Figure 3.2. Dominated and Non-dominated Solutions (Araujo, Poldi, & Smith, 2014)

In both graphics above, the abscissa and the ordinate consist of the two objectives of the problem (f_1 and f_2). The grey area in (a) represents the feasible solution space. Each solution in this area is a feasible solution. The dashed curve shows the Pareto front. In other words, all the solutions on the Pareto front are non-dominated by any feasible solution, and the solutions on the Pareto front dominate all the feasible solutions that are not on the Pareto front. For example, the solutions A and B are on the Pareto front and are non-dominated solutions (i.e. $A \sim B$). The solutions C, D, E, F, and G are dominated by the solutions A and B (i.e. $A < C, A < D, A < E, A < F, A < G, B < C, B < D, B < E, B < F$ and $B < G$).

The graphics (b) shows the dominance and non-dominance of a single solution. The solutions in the bottom left square of C (A and B) dominate the solution C. The solutions in the top right square of C (E and F) are dominated by the solution C. The solutions in both the top left and bottom right squares of C are non-dominated with the solution C. According to Figure 3.2 (b), $A < C$ and $B < C, C < E$ and $C < F, C \sim D$ and $C \sim G$.

3.3. Solution Methods for MOOPs

Solving MOOPs are different than solving SOPs. It is because, in SOPs, there is only one objective function to be optimized. In each iteration an algorithm for solving SOPs, the algorithm tries to find a better solution than the best-so-far solution. Therefore, the whole process focuses on optimizing a single solution. In MOOPs, however, there is no “the best” solution due to the nature of the conflicting objectives. One solution that provides good results for one of the objectives may not perform the same success for the others. It may even worsen the results of these objectives. Handling all the objective functions simultaneously leads MOOP to progress by taking the whole Pareto-optimal solutions (i.e. Pareto front) into consideration. In order to do that, many different methods have been suggested for the solution of MOOPs. All these methods fall into either of these two categories: classical methods or evolutionary methods.

3.4. Classical Solution Methods for MOOPs

Classical methods consist of no-preference methods, posteriori methods, priori methods, and interactive methods. The *no-preference methods* basically generate a few non-dominated solutions to the problem and allow the decision-maker to select from

these results. There is no priori knowledge about the importance of the objective functions. The method, however, uses a heuristic to find a single solution to the problem. The *posteriori methods*, on the other hand, contain some information about the importance of the objectives and the parameters of the algorithm. They use this information to generate a Pareto-optimal solution iteratively. The *priori methods* progress like posteriori methods. They also use information about the importance of objectives and produce one Pareto-optimal solution. The *interactive methods* use priori information about the objectives, as well, but they obtain this information progressively. The following methods are some classical methods which use either priori or progressive information about the importance of objectives (Deb, 2001).

3.4.1. Weighted Sum Method

Weighted sum method is probably one of the easiest, earliest and widely used classical method for the solution of MOOPs. In this method, all the objective functions are weighted with non-negative weight values. Then, the objective function of the problem becomes the linear combination of these weighted objective functions. This way, the MOOP is converted into an SOP by scalarizing all the objective functions into a single objective function. The weighted sum method, then, tries to find a decision vector which optimizes this single objective function. Once it is found, then each objective function of the problem is calculated with this decision vector and these results are considered as the solution for the problem. The mathematical formulation of the weighted sum method is as follows (Deb, 2001):

$$\begin{aligned}
 & \text{optimize } F(x) = \sum_{m=1}^M w_m f_m(x) \\
 & \text{subject to } g_j(x) \geq 0, \quad j = 1, 2, \dots, J \\
 & \text{where } \sum_{m=1}^M w_m = 1
 \end{aligned} \tag{6}$$

The weights for the functions are determined by the decision maker. The weights can either be the same if there is not much priori information about the objective functions. If there is enough priori information about the objective functions, then the weights can be distributed proportionally with regards to the importance of the objective functions. Besides determining the weights, the objective function values should be normalized, as well. It is because if the value range of any objective function differs from the others significantly, then the single objective function obtained by the sum of

these weighted objective functions will favor this objective function. In order to prevent this, the value ranges of the objective functions must be normalized before scalarized with weights.

The Pareto-optimal solution is obtained by iteratively changing the weights of each objective function. At each change of these weights, the algorithm is run for this new single objective function in order to find a new optimum solution depending on these parameters. The results of each objective function that are obtained by different weight combinations are compared with each other by dominance. The non-dominated solutions after this comparison construct the Pareto front of the solution.

Weighted sum method has both advantages and disadvantages (Deb, 2001). Its advantages are that it can easily be applied and used for MOOPs. And, if there is a priori information about the importance of objective functions, the weights can adequately be arranged proportionately. For the MOOPs that have convex Pareto-optimal front, the weighted sum method can find the Pareto-optimal solution (Figure 3.3). On the other hand, the weighted sum method has some serious disadvantages. Weighted sum method cannot find Pareto-optimal solutions in the non-convex feasible solution space (Figure 3.4). Furthermore, changing the weights of each objective function can be done infinitely. Since those weights are Real numbers, then there is no way for the decision maker to try every possible combination of the weights. Even if the decision maker determines a discrete step for changing the weights, there is still a need for an explanation of the size of these steps. As a result, the weighted sum method carries a sort of arbitrariness.

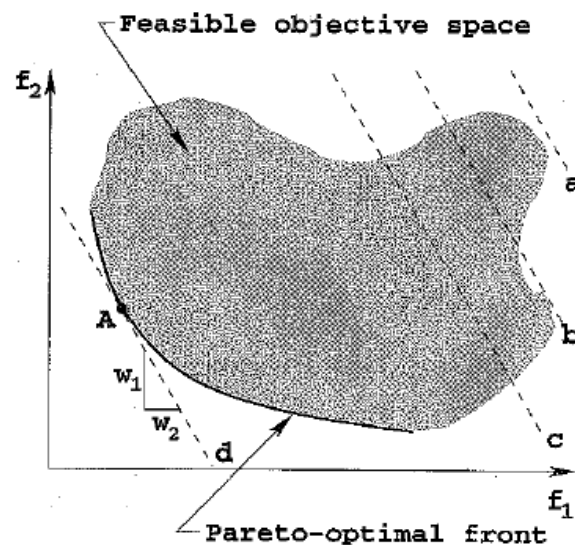


Figure 3.3 Weighted Sum Method on a Convex Feasible Solution Space (Deb, 2001)

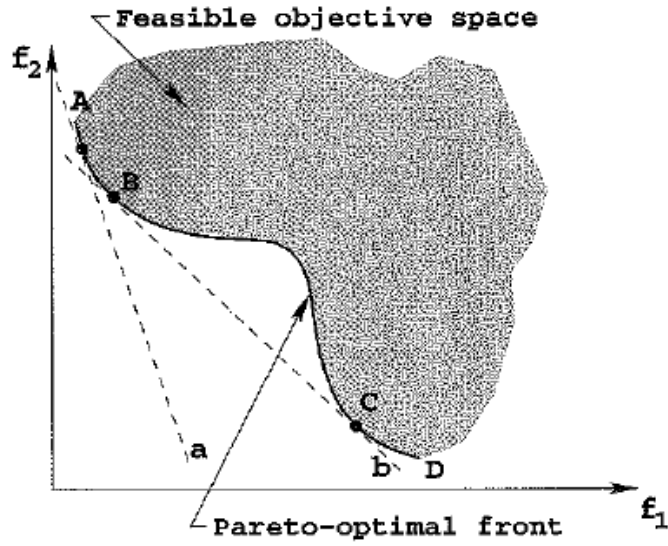


Figure 3.4 Weighted Sum Method on a Non-convex Feasible Solution Space (Deb, 2001)

3.4.2. ϵ -Constraint Method

The ϵ -constraint method was proposed in order to provide a solution to the problems which weighted sum approach encounters with non-convex solution spaces. In this method, the objective functions are not scalarized into a single objective function by weights. One of the objective functions in the MOOP, however, is chosen as the single objective function for the problem. The rest of the objective functions are moved into the constraints set with an upper bound labeled with ϵ_m . The mathematical formulation of this method is as follows (Deb, 2001):

$$\begin{aligned}
 & \text{optimize } f_i(x) \\
 & \text{subject to } f_m(x) \leq \epsilon_m \quad m = 1, 2, \dots, M, \quad m \neq i \\
 & \quad \quad \quad g_j(x) \geq 0, \quad j = 1, 2, \dots, J
 \end{aligned} \tag{7}$$

Each ϵ_m for the respective objective function divides the feasible solution space into regions restricted by these ϵ_m . The union of all these regions become the feasible search space for the MOOP. Then, the goal is to find the optimum value of the function of $f_i(x)$ in this feasible search space. Only one optimum solution for the MOOP is obtained for each ϵ_m value. Optimum solutions for each feasible search space regions are obtained by iteratively changing the ϵ_m values. Among all these solutions, the non-dominated ones generate the Pareto-optimal solution for the MOOP. Figure 3.5

demonstrates an example process of the ϵ -constraint method on a bi-objective optimization problem.

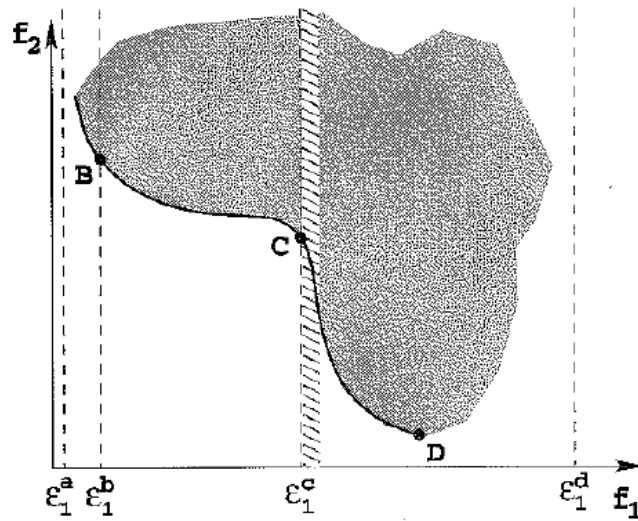


Figure 3.5 ϵ -Constraint Method (Deb, 2001)

The ϵ -constraint method is advantageous in its ability in providing solutions for non-convex solution spaces. In that sense, it works both on convex and non-convex feasible regions. On the other hand, selecting which objective function to be the single objective function of the system, deciding on the ϵ_m and the iteration number or steps all depend on the decision maker. If there is a priori information about the importance of objectives and restrictions on the objective functions, then the method can produce effective Pareto-optimal solutions. If there is no priori information as such, then the method may end up in infeasible solutions or no Pareto-optimal solutions. For these reasons, like the weighted sum method, the ϵ -constraint method is also highly dependent on the priori information about the MOOP.

3.4.3. Weighted Metric Methods

The weighted metric method is the generalized version of the weighted sum method. In weighted metric methods, a distance metric of l_p is used. All the objective functions are combined into this single objective function of l_p as shown in Equation (8). Therefore, l_p becomes the main single objective function to be optimized for the MOOP. l_p is calculated by getting the distance between each objective function and the ideal solution of z^* , scalarizing these distances by non-negative weights, getting the sum of these scalarized distances, and exponentiating it with $1/p$. The mathematical formulation is described as such (Deb, 2001):

$$\begin{aligned} & \text{optimize } l_p = (\sum_{m=1}^M w_m |f_m(x) - z_m^*|^p)^{1/p} \\ & \text{subject to } g_j(x) \geq 0, j = 1, 2, \dots, J \end{aligned} \quad (8)$$

where $\sum_{m=1}^M w_m = 1$

As it can be seen in the formulation, optimizing this single objective of l_p depends on the parameters of the ideal solution z^* , the weights of each objective function and the exponentiation parameter of p . The ideal solution z^* is either the utopian objective vector or the vector of best-known results of each objective function. The weights, as in weighted sum method, can either be distributed evenly or determined according to a priori information about the importance of the objective functions. The sum of these weights is one. The parameter p can be any value in the interval of $[1, \infty)$. These values of p determine the characteristic of the weighted metric method. For example, when $p = 1$ the method turns into the weighted sum method (Figure 3.6(a)). When $p = 2$, l_p calculates the sum of the Euclidean distances between the objective functions and the ideal solution z^* (Figure 3.6(b)).

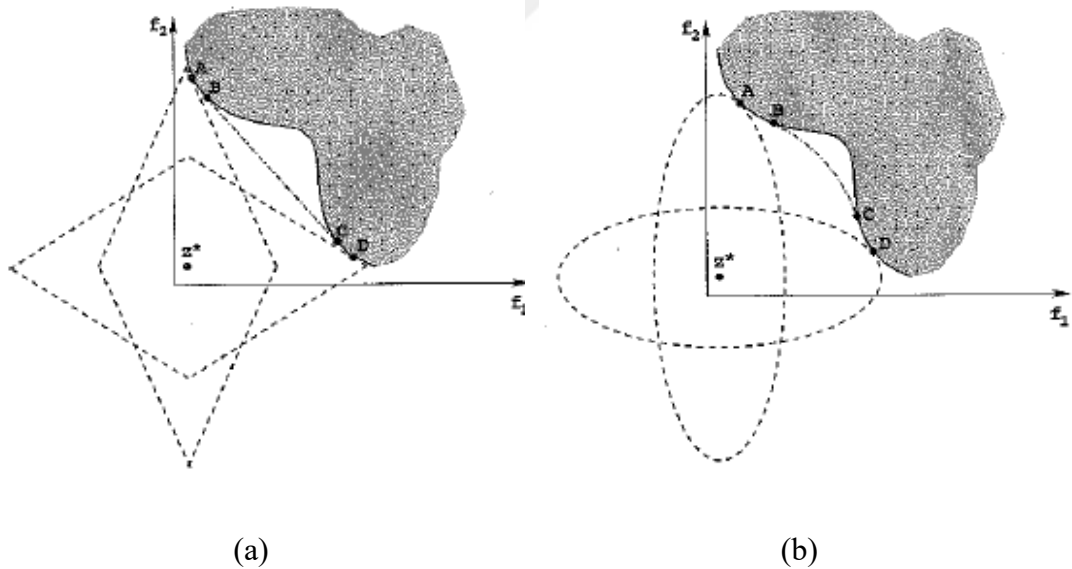


Figure 3.6 (a) Weighted Metric Methods: (a) when $p = 1$, (b) when $p = 2$ (Deb, 2001)

The parameter p can also be infinite. This weighted metric with $p = \infty$ is called *the weighted Tchebycheff* method. This way, the weighted method calculates l_p as following (Deb, 2001):

$$l_{\infty} = \max_{m=1}^M w_m |f_m(x) - z_m^*| \quad (9)$$

Once the parameter p and the vector z^* are determined for the problem, the Pareto-optimal solutions are obtained by iteratively changing the weights of the single objective function of l_p . As the value of p increases, then the obtained solutions converge to the Pareto-optimal solutions set. Theoretically, in the case of $p = \infty$, the real Pareto-optimal set is found.

Weighted metric methods also have some advantages and disadvantages. The primary advantage of the method is that it can be used in both convex and non-convex feasible solution spaces due to the parameter of p . Another advantage of this method is that when the weighted Tchebycheff metric is used, it guarantees a Pareto-optimal solution. This method, however, has certain disadvantages. It is highly dependent on the selection of the parameter p and the weights for the objective functions. It requires a priori information about the importance of objective functions and needs the ideal solution vector z^* or any information about this vector. Furthermore, as the value of p increases, the problem becomes non-differentiable. As a result, gradient-based methods cannot be used in finding the optimum solution (Deb, 2001).

Besides all these methods, there are other classical methods in solving the MOOPs. Most of them, however, are derived methods out these classical methods mentioned here. Since they are out of the scope of this thesis, they are not mentioned here.

3.5. Evolutionary Algorithms

Evolutionary Algorithms (EAs) are metaheuristic optimization algorithms. Unlike the exact methods, they include a stochastic process in finding solutions. EAs mimic the biological evolution processes such as recombination, mutation, and natural selection. In EAs, each solution candidate is considered as an individual. The set of all these individuals constitutes a population. EAs begin with an initial population which is constructed either randomly or by using certain heuristics. Then, the individuals in this initial population are selected using a criterion for producing new individuals. This step is called parent selection. In the reproduction process, recombination and/or mutation mechanisms are used. Optionally, a local search algorithm can be included for improving the quality of each individual. The quality of an individual is evaluated

by fitness functions. Depending on the problem, the goal of the process might either be minimizing or maximizing these fitness functions.

EAs include an iterative process. In each iteration, a new population is generated by reproduction, local search, evaluation, and selection operations. EAs end when they satisfy the termination criterion of the algorithm. This termination criterion can be running for a certain number of iterations, using predefined computational resources or obtaining results within a threshold, etc. The general structure of an EA is described in the following pseudo-code (Deb, 2001).

Initialization

Evaluation

Repeat

Parent Selection

Reproduction (Recombination and/or Mutation)

[Optional: Local Search]

Fitness Evaluation

Survival Selection

Until the Termination Criterion is Satisfied

Figure 3.7 Pseudo-code for EAs

Many EA methodologies have been proposed since the origins of EAs in the 1950s. They, however, can be listed under these categories: Genetic Algorithms (GAs), Evolutionary Strategies (ESs), Evolutionary Programming (EP), and Genetic Programming (GP).

Like in classical methods for MOOPs, there are advantages and disadvantages of EAs. One of the advantages of EAs is that they can explore and exploit the search space of the problem in a reasonable amount of time due to their stochastic process. Although EAs do not guarantee to obtain the optimum solution, they can easily be applied to the problems which cannot be solved by exact methods due to time limitations. EAs can be applied to both SOPs and MOOPs. In the case of MOOPs, since the required solution is a Pareto-optimal set, EAs fit better as a solution method than the classical methods mentioned in the previous section. It is because, in each iteration, an EA produces a population of solutions which gradually contributes to the Pareto-optimal front for the solution. The major disadvantage of EAs is that they do not guarantee the optimum solution. In addition, the parameters used in different parts of EAs (i.e. probability ratios, operator types, etc.) need to be determined by the decision maker.

There has been many EAs proposed for MOOPs such as Multi-Objective Genetic Algorithm (MOGA), Niche Pareto Genetic Algorithm (NPGA), Vector Evaluated Genetic Algorithm (VEGA), Strength Pareto Evolutionary Algorithm (SPEA), Non-dominated Sorting Genetic Algorithm (NSGA), NSGA-II, ϵ -based Multi-Objective Evolutionary Algorithm (ϵ -MOEA), Pareto Archived Evolutionary Strategy (PAES), etc. However, only two of these MOOPs have been studied in this thesis: NSGA-II and ϵ -MOEA. The former algorithm uses only a population in its process while the latter algorithm uses both a population and an archive in its process.

3.5.4. Non-Dominated Sorting Genetic Algorithm II (NSGA-II)

Srinivas and Deb developed Non-Dominated Sorting Genetic Algorithm (NSGA) by implementing the concept of non-dominated sorting on GAs in 1994 (Srinivas & Deb, 1994). NSGA, however, was criticized for having a high time complexity, not having an elitist selection approach, and not providing enough diversity in the Pareto-optimal solution. Therefore, Deb et al. constructed another non-dominated sorting algorithm that responded to these critiques. This new algorithm is called “a Fast and Elitist Non-dominated Sorting Algorithm II” (NSGA-II) (Deb, Pratap, Agarwal, & Meyarivan, 2002).

In both of NSGA and NSGA-II algorithms, ranking the individual solutions play a critical role. Each individual in the population is ranked according to the dominance criterion. All the individuals in the population are compared with each other using this dominance criterion. The non-dominated individuals obtained by this comparison consist of the first front (i.e. Pareto front) and are ranked with 0. Once the first front is found, the individuals in this Pareto are temporarily discounted in order to find the next front. With this temporary discount of the previous front, the rest of the individuals get compared with one another and ranked by the same dominance comparison. This second front is ranked with 1. Then the algorithm continues to find the next ranked front. This ranking process continues until the last front is found and all individuals are assigned a rank. In the end, the whole population gets sorted according to their ranks by finding these fronts (Figure 3.8).

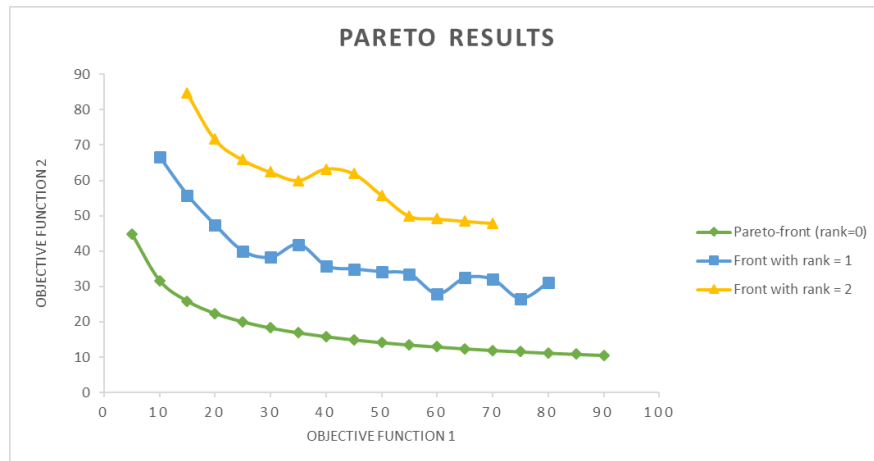


Figure 3.8 A sample for the fronts of a MOOP

In NSGA, the diversity of the solutions is obtained by a shared function. This sharing function finds and evaluates the groups of solutions which have similar or close fitness values for each objective function. As a result, in the selection process, only one of the solutions in the same group is selected in order to protect diversity in the fronts. This sharing function, however, is highly dependent on the decision maker's determination of the sharing values and it increases the overall complexity of the algorithm. Therefore, in NSGA-II, a new method called crowding distance was developed to substitute this sharing function. Crowding distance is the sum of all the distances of each solution to the other solutions. NSGA-II uses this value to eliminate all similar solutions in the same crowding box and preserve diversity throughout the process (Figure 3.9).

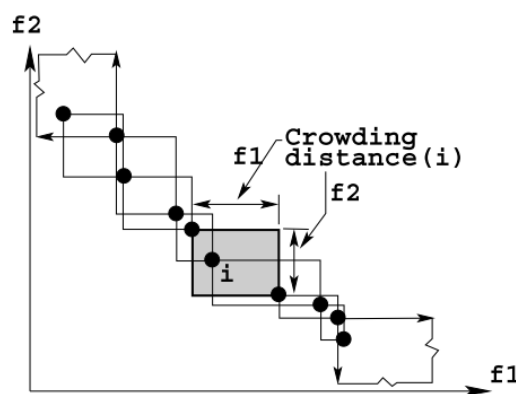


Figure 3.9 Crowding distance in NSGA-II (Deb, Mohan, & Mishra, 2003)

The general structure of NSGA-II is as follows (Deb et al., 2002):

Generating an initial population
Evaluating fitness values
Repeat
 Reproducing offspring by recombination and/or mutation on the parent population
 Evaluating the offspring's fitness values
 Combining the parent and offspring populations
 Finding fronts by fast non-dominated sorting
 Crowding distance sorting
 Truncating the combined population
 Replacing the parent population with the population after truncation
Until *the Termination Criteria is satisfied*
The solution is the Pareto front (i.e. the front with rank 0)

Figure 3.10 Pseudo-code for NSGA-II

In generating the initial population, a random process or a constructive heuristic can be used. The individuals that will be used in recombination and/or mutation are selected according to a non-dominated selection method. In this type of selection, the ranks are used in the evaluation. If one of the candidates in the selection group dominates the other(s) than it is selected. Otherwise, if the candidate individuals in the selection group are non-dominated, then one of them is chosen randomly. Once the individuals from the parent population are selected, then recombination and/or mutation operators are applied to these selected individuals. These operators are the same as the recombination and mutation operators in classical GA. After the fitness values of these newly reproduced offspring are calculated, they are added to the offspring population. Then, the parent and offspring population are combined. The individuals in this new extended population are sorted in a non-dominated manner and are assigned ranks to determine their fronts. Then, their crowding distance is calculated, and the truncating process is applied to the whole population. The truncated population becomes the next parent generation. This whole process repeats until the termination criteria are satisfied. When the algorithm ends, the front ranked with 0 is the Pareto front solution for the MOOP.

3.5.5. ϵ -based Multi-Objective Evolutionary Algorithm (ϵ -MOEA)

ϵ -based Multi-Objective Evolutionary Algorithm (ϵ -MOEA) was developed by Deb, Mohan and Mishra (Deb et al., 2003). ϵ in the name of the algorithm refers to the area of the hyper-boxes in the search space. ϵ -MOEA divides the problem's search space into the grids called hyper-boxes with the lengths of different ϵ 's determined by the

decision maker. Only one solution exists in each of these hyper-boxes. In other words, if any two solutions fall into the same box, they are considered as the same and only one of them is chosen as a solution. For the solutions in different hyper-boxes, the dominance comparison is applied. This process is called ϵ -dominance comparison. This way, the diversity among solutions are preserved by eliminating the near solutions in the solution space (Figure 3.11).

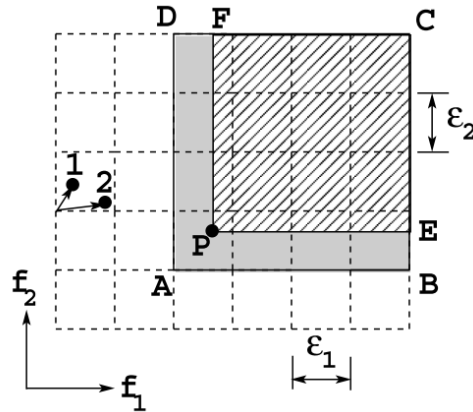


Figure 3.11 Hyper-boxes in ϵ -MOEA (Deb et al., 2003)

ϵ -MOEA uses two evolving populations in its process. The first one is called Evolutionary Algorithm (EA) population $P(t)$ and the second one is called the archive population $E(t)$ (where t is the iteration index). For simplicity, the former will be called population and the latter will be called archive from now on. The algorithm begins with generating an initial population which will be the population of the algorithm. Generating this initial population can be done either by randomization or using certain constructive heuristics. Then the fitness values of each individual are calculated, and every individual is compared with one another by ϵ -dominance comparison. The ϵ -non-dominated individuals are added to the archive. After these initial moves, the iteration phase of the algorithm begins. In each iteration, first, the reproduction takes place. For reproduction, one individual is selected from the population and another one from the archive for mating. After reproduction (i.e. crossover and/or mutation) of these two individuals, the offspring are produced. Then, these offspring are tried to be added to both the population and the archive.

For inclusion of the offspring in the population, the following cases are taken into consideration (Kollat & Reed, 2006). The following items must be understood as if-else statements.

- i. If the offspring dominates any individuals in the population, then it is replaced with one of them at random.
- ii. If the offspring is dominated by any individual in the population, then the offspring is not included in the population.
- iii. If the offspring is non-dominated by all the individuals in the population, then it is replaced with one of the population individuals at random.

For inclusion of the offspring into the archive the following cases are taken into consideration (Kollat & Reed, 2006) (Deb et al., 2003). The following items must be understood as if-else statements.

- i. If the offspring ϵ -dominates any individuals in the archive, then all the ϵ -dominated individuals in the archive are removed and the offspring is included to archive.
- ii. If the offspring is ϵ -dominated by any individuals in the archive, then the offspring is discarded.
- iii. If the offspring is ϵ -non-dominated, then:
 - a. If the offspring is not in any of the ϵ hyper-boxes in the archive then it is included in the archive.
 - b. If the offspring is in any of the ϵ hyper-boxes in the archive, then the normal dominance criteria is applied within this ϵ hyper-box. If the offspring and the individual in the ϵ hyper-box are still non-dominated then one of them is selected at random and the other one is removed.

This way, the size of the population is kept constant, but it is gradually improved by the better offspring in each iteration. The archive, on the other hand, is not limited in size and may enlarge or shrink through each iteration because of the ϵ -dominance comparisons. Because of the ϵ hyper-boxes in the archive, archive preserves the diversity in among its solutions.

ϵ -MOEA algorithm continues the same process at each iteration until the termination criterion is satisfied. Once the algorithm finishes its run, the archive is the Pareto front solution for the MOOP. The general structure of ϵ -MOEA algorithm is as follows (Deb et al., 2003):

Generating an initial population
Evaluating fitness values
Updating the archive by ϵ -dominance
Repeat
 Reproducing offspring by recombination and/or mutation on the population
 Evaluating the offspring's fitness values
 Updating the population by ϵ -dominance
 Updating the archive by ϵ -dominance
Until the Termination Criteria is satisfied
The archive is the solution

Figure 3.12 Pseudo-code for ϵ -MOEA

3.6. Performance Metrics

Evaluating the performance of Multi-Objective Evolutionary Algorithms (MOEAs) is very different than evaluating the performance of classical Evolutionary Algorithms (EAs) for SOPs. In the latter, the best result of the algorithm for its single objective function is compared with the best-so-far solution or the optimum solution if it exists. The statistical results of this comparison (i.e. success ratio, standard deviation, etc.) measure the performance of the algorithm. In MOOPs, however, the comparison must be done according to the non-dominance criterion since there is a set of non-dominated solutions in the result. In literature, many performance metrics have been provided for evaluating the performance of the MOOPs. According to Riquelme et al., there are 54 performance metrics being used in the MOOP literature (Riquelme, Lüken, & Baran, 2015). In this thesis, only three of them are used and therefore these three performance metrics will be explained. These performance metrics are Hypervolume (HV), Generational Distance (GD), and Inverted Generational Distance (IGD). There are certain reasons for picking these three measures. First, they have been in the top five of the most used metrics for MOOPs in the literature. Second, while all of them measures the accuracy of the algorithm, HV and IGD also measure the diversity of the solutions provided by the algorithm. Third, they are all unary metrics, that is, they need just one set of solutions to measure the performance (Riquelme et al., 2015).

3.6.1. Hypervolume (HV)

Hypervolume (HV) performance metric was introduced by Zitzler and Thiele (Zitzler & Thiele, 1998). HV metric is a unary metric which evaluates the performance of an MOEA by calculating the area (or volume) between the Pareto front of the algorithm

and the selected reference point in the solution space. HV calculates the sum of the consecutive areas (or volumes) between each consecutive solution in the Pareto front and the reference point (Figure 3.13).

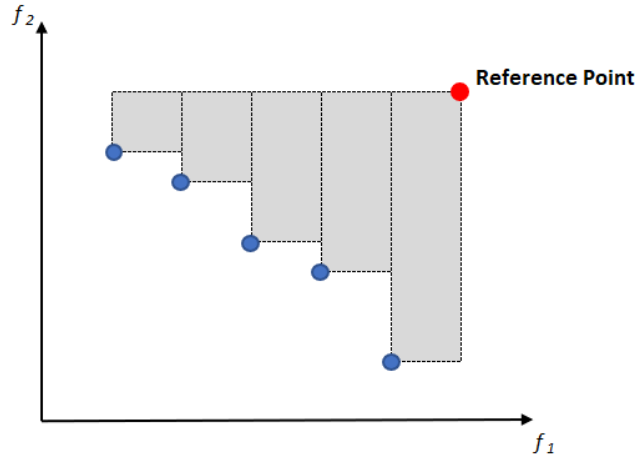


Figure 3.13 Hypervolume for a minimization MOOP with two objectives

The mathematical formula of HV is as follows (Friedrich, Horoba, & Neumann, 2009):

$$I_H = volume \left(\bigcup_{(x_1, \dots, x_n) \in X} (r_1, x_1) \times (r_1, x_1) \times \dots \times (r_n, x_n) \right) \quad (10)$$

where X is the Pareto front and $r = (r_1, \dots, r_n)$ is the reference point.

Since HV is a unary performance metric, each algorithm can be evaluated individually by its HV value. For the maximization problems, the reference point can be selected as the origin of the objective space. For the minimization problems, the reference point is the point in which each coordinate is the optimum solution for the related objective function if these optimum solutions exist and are known. Otherwise, it can be the point in which each coordinate is the best-so-far solution found by running different MOEAs. In both minimization and maximization problems, the algorithms having a greater HV value are said to be more promising than the other ones.

3.6.2. Generational Distance (GD)

Generational Distance (GD) metric was proposed by Veldhuizen and Lamont (Van Veldhuizen & Lamont, 2000). The main idea in GD is to calculate the distance between the Pareto front found by the MOEA and the true Pareto front (Figure 3.14). More specifically, GD calculates an average minimum Euclidean distance between each solution in the found Pareto front and those in true Pareto front.

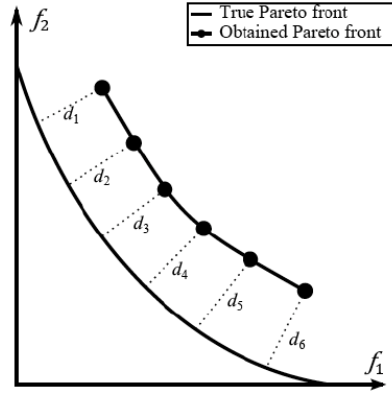


Figure 3.14 Generational Distance for a MOOP with two objectives (Sheng et al., 2014)

The mathematical formulation of GD is as follows (Tian, Zhang, Cheng, & Jin, 2016):

$$GD(P, P^*) = \frac{\sqrt{\sum_{y \in P^*} \min_{x \in P} d(x, y)^2}}{|P|} \quad (11)$$

In this formula, P refers to the Pareto front found by the MOEA and P^* refers to the true Pareto front. x represents the solutions on the found Pareto front while y refers to the solutions on the true Pareto front. The function $d(x, y)$ calculates the Euclidean distance between x and y .

GD measures the convergence of P , therefore, a smaller GD value indicates a better solution. Unlike HV, GD is not able to evaluate the diversity of P .

3.6.3. Inverse Generational Distance (IGD)

Inverse Generational Distance (IGD) was proposed by Coello and Cortes (Coello & Cortés, 2005). They based their method on GD. In IGD, the distance from the true Pareto front to the found Pareto front by the MOEA is measured. This is why it is called “inverted”. In IGD, the average distance of every point on the true Pareto front to the points on the found Pareto front by the MOEA is calculated (Figure 3.15).

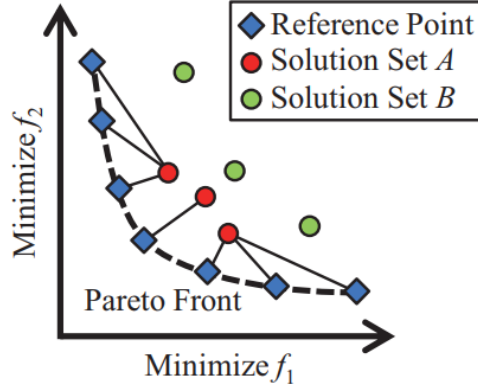


Figure 3.15 Inverted Generational Distance for a MOOP with two objectives (Ishibuchi, Masuda, & Nojima, 2016)

The mathematical formula of IGD is shown below (Tian et al., 2016):

$$IGD(P, P^*) = \frac{\sum_{x \in P^*} \min_{y \in P} d(x, y)}{|P^*|} \quad (12)$$

In this mathematical formula, P refers to the Pareto front found by the MOEA and P^* refers to the true Pareto front. x represents the solutions on the found Pareto front while y refers to the solutions on the true Pareto front. The function $d(x, y)$ calculates the Euclidean distance between x and y . Like GD, the smaller values of IGD indicates that the found Pareto front by the MOEA is closer to the true Pareto front.

IGD, however, has some advantages over GD. First, it measures the convergence of P better than GD. It is because, when P has fewer solutions, GD does not give much information about the distance between P and P^* (Coello & Cortés, 2005). Second, IGD measures both convergence and diversity of P , while GD measures only convergence (Tian et al., 2016).

CHAPTER 4

EVOLUTIONARY ALGORITHMS FOR SOLVING A MULTI-OBJECTIVE GREEN VEHICLE ROUTING PROBLEM

4.1. Definition of the Problem

In this thesis, a Multi-Objective Green Vehicle Routing Problem (MOGVRP) was studied. It is because environmental concerns increase day by day with the increase of the level of pollutants caused by the vehicles, and there has been less amount of studies being done in GVRP compared to the other variants of VRP. In addition, there has been a growing interest in GVRP recently. The reason for this recent interest is twofold. First, along with the growth of number of vehicles around the world, the pollution caused by these vehicles has been increased proportionally. According to the European Environment Agency's 2011 Report, greenhouse gases (GHG) emitted in transportation in 2011 increased by 24% compared to the GHG emission in 2009 (Eglese & Bektaş, 2014). This led the European Parliament and Council to adopt regulations on new mandatory CO_2 emissions for passenger cars and vans in 2009 and 2011. Since then, despite some ups and downs, there has been an increase in CO_2 emissions by vehicles in Europe according to the European Environment Agency's 2019 Report (Pastorello, 2019). As a result, many environmental organizations have begun to advise solutions on urban and highway transportation. Second, the companies that use vehicle fleets for transportation or delivery have concerns about their fuel consumption as well as delivery times and travelled distances due to the incremental cost of fuel (Xiao, Zhao, Kaku, & Xu, 2012). Decreasing the costs of these vehicle fleets will also be of benefit for the transportation companies.

The MOGVRP studied in this thesis has two objectives. One of them is minimizing the total distance of all vehicle routes, while the other one is minimizing the total fuel consumption of all vehicle routes. For the former objective's fitness evaluation, 2D Euclidean distance calculation was used. For the latter objective's fitness evaluation, the fuel consumption formula presented by Xiao et al. (Xiao et al., 2012) was used. It is because this fuel consumption formula is highly referenced by other articles and it only requires the distances between the customers (and the depot) and the amount of the demands of the customers. As a result, it can be applied to any VRP instance. The detailed formulations of these objective functions are explained in Section 4.3.

NSGA-II and ϵ -MOEA were used in solving the MOGVRP studied in this thesis. Along with these algorithms, certain constructive heuristic methods and local search algorithms were included in the solution method. The parameters and the nature of these methods and algorithms are explained more in detail in Section 4.5.

Seven Christofides (Christofides, Mingozzi, & Toth, 1979) and five Golden (Golden, Wasil, Kelly, & Chao, 1998) instances were selected for experimental studies in this thesis. Xiao et al (Xiao et al., 2012) also used the same instances and provided single objective results for both objectives. Therefore, their results were taken as Pareto front results containing two solutions for each instance, and the Pareto-results obtained by the algorithms used in this thesis were compared and evaluated against them via the performance metrics mentioned in the previous chapter.

4.2. Literature Review

The studies done on GVRP mainly focus on minimizing the emission of air pollutants that cause GHG, i.e. CO, CO₂, CH₄ and NO_x (Eglese & Bektaş, 2014)(Demir, Bektaş, & Laporte, 2013)(Toth & Vigo, 2014)(Boulter, Mccrae, & Barlow, 2007). Different studies on GVRP included different components to their emission functions and used different algorithms. Figliozzi (Figliozzi, 2010) focused on Emissions in VRPTW with three objectives: minimizing emissions, minimizing fuel consumption and minimizing total travel distance. He considered the problem as a SOVRP and applied his IRCI (Iterative Route Construction and Improvement) algorithm. He used an emission function that includes the average speed of vehicles and their travel distances. Since the benchmark set he used (Solomon's well-known 56 instances) does not include data on the speed of the vehicles, he generated three different speed levels depending on the urban traffic: uncongested, somewhat congested and congested.

Kuo (Kuo, 2010) studied a Time-Dependent VRP (TDVRP) on Solomon's 100-customer Euclidean problems using three objectives: minimizing total time, minimizing fuel consumption and minimizing total distance. He considered these objectives as a SOVRP in turn and applied a simulated annealing algorithm. He considered the speed and weight of the vehicles and their traveling distances as components of the fuel consumption function. Similar to Figliozzi, Kuo also classified his speed constants in three categories: low, medium and high.

Bektaş and Laporte (Bektaş & Laporte, 2011) studied Pollution-Routing Problem (PRP) which is a type of GVRP. They focused on a single objective which basically calculates the total fuel consumption of vehicles. This fuel consumption is the weighted sum of three functions which are the cost of the vehicle's load, the cost caused by variations in the vehicle's speed and the salaries of the drivers. Since there is no concrete speed data, they used the average of lower and upper bounds of the speed on each arc between the customers. They applied their study on a real-life case (using generated data for 10, 15 and 20 cities in the UK) and analyzed the effects of each component in the fuel consumption function.

Demir et al. (Demir, Bektaş, & Laporte, 2011) reviewed and compared six different fuel consumption models using simulations with 18 scenarios. The fuel consumption models they have considered are instantaneous fuel consumption model, four-mode elemental fuel consumption model, running speed fuel consumption model, and comprehensive modal emission model, MEET (methodology for calculating transportation emissions and energy consumption) and COPERT (computer program to calculate emissions from the road transportation model).

Jabali et al. (Jabali, Woensel, & Kok, 2012) studied an Emission-based Time-Dependent VRP (E-TDVRP) as a SOVRP. They used a cost function containing the hourly cost of a driver, CO_2 emission cost, and fuel cost. Their fuel consumption model also includes various speed parameters of the vehicle such as optimal speed, congestion speed in traffic, the upper limit of speed. They implemented a Tabu search algorithm and used a specific benchmark set used in (Jabali et al., 2012).

Demir et al. (Demir, Bektaş, & Laporte, 2012) studied PRP with Time Windows (PRPTW) as a SOVRP. In their study, they used a more comprehensive fuel consumption model than the fuel consumption models used in other GRVP studies. The parameters in their model that impact fuel consumption are the weight of the vehicle, lower and upper bounds in the speed of the vehicles, rolling and air resistance, air density, engine friction factor, the heating value of typical diesel fuel, fuel and CO_2 emission cost per liter. Most of these parameters were taken into consideration with their estimated values. The authors developed a heuristic algorithm called Adaptive Large Neighborhood Heuristic Algorithm (ALNS) and applied it on nine different instance sets which were generated by the authors.

Demir et al. (Demir et al., 2013) improved their study on PRPTW and applied their enhanced ALNS Algorithm on newly generated instances based on real geographic data. In their study, the authors considered the PRPTW problem as a MOVRP with two objectives: minimizing fuel consumption and minimizing total travel time. As in their former studies, their fuel consumption function included many parameters such as the weight of the vehicle, lower and upper bounds in the speed of the vehicle, rolling and air resistance, air density, engine friction factor, heating value of typical diesel fuel, fuel and CO_2 emission cost per liter. The authors used four different methods for the solution: weighted sum, weighted sum with normalization, ϵ -Constraint, and a hybrid method (hybridization of the adaptive weighted sum and ϵ -Constraint methods). They obtained Pareto fronts (i.e. non-dominated solution sets) for each method and compared their results. Their comparison also includes two performance metrics: Hypervolume and ϵ -indicator.

Xiao et al. (Xiao et al., 2012) focused on Fuel Consumption Rate on VRP (FCVRP) with two objectives: minimizing total fuel consumption rate and minimizing total distance. They solved the FCVRP as a SOVRP for each objective by using a String-model-based Simulated Annealing (SMSA) Algorithm developed by the authors. They also formulated a fuel consumption function which only depends on the weight of the vehicle and the distance traveled by the vehicle. In this way, their fuel consumption function can be used in many instances in literature without generating, estimating or assuming extra data such as speed, engine power, the slope of the roads, etc. The authors applied their SMSA method along with their fuel consumption function on 7 instances of Christofides et al. (Christofides et al., 1979) and 20 instances of Golden et al. (Golden et al., 1998).

Koç and Karaođlan (Koç & Karaođlan, 2016) developed an exact solution approach which is based on the simulated annealing heuristic and branch and cut algorithm to solve a single objective GVRP. In their study, they aimed to provide the optimum solution to the transportation companies in organizing the routes of their vehicles which use alternative energy such as electricity or natural gas.

Dükkancı et al. introduced the Green Location-Routing Problem GLRP which includes locating the depots from which the vehicles will be dispatch and solving the classical TWVRP with multiple depots (Dükkancı, Kara, & Bektaş, 2019). Their problem contains a single objective function which consists of two components. The first

component minimizes the fixed cost of operating depots and the second component minimizes the total fuel consumption cost. Dükkancı et al. proposed two heuristic algorithms: Cumulative Location-Routing and Speed Optimization Algorithm (CLRSOA) and Iterated Local Search algorithm (ILS). They concluded that the “green” aspect in their objective function affect the optimal solutions.

Macrina et al. (Macrina, Laporte, Guerriero, & Pugliese, 2019) studied an energy efficient GVRP in which they tried to reduce the costs caused by electric and conventional vehicles. For the electric vehicles, they also took the recharging operations into account in their objective function. In their study, they propose an energy consumption model that includes real-life parameters such as the acceleration and deceleration aspects of the vehicles while driving.

According to Demir et al. (Demir et al., 2013), GHG emissions are directly proportional to fuel consumption. It means that minimizing fuel consumption also means minimizing GHG emissions. As it is seen in the literature, various fuel consumption functions have been proposed. Most of them are highly dependent on real case data such as vehicle’s type and speed, engine power, air density and resistance, road surface wear, gradient and resistance, fuel type, traffic conditions, etc. (Boulter et al., 2007) (Ardenkani, Hauer, & Jamei, 2001). Although the VRP problem gets more realistic when all these parameters are taken into consideration, it is unlikely to have all of this information for any benchmark instance in the literature. Therefore, if a simple but efficient fuel consumption function, which is less dependent on external data, is selected then the performance measures of different algorithms on the same benchmark sets can be compared and analyzed.

For this reason, in this thesis, the same VRP and fuel consumption formulation in Xiao et al. (Xiao et al., 2012) is used. However, these objectives have been used in the MOGVRP. For this purpose, two well-known Multi-Objective Evolutionary Algorithms, i.e. NSGA-II and ϵ MOEA, have been used for the solutions.

4.3. The Mathematical Model of the MOGVRP

The mathematical model of a GVRP contains the general structure of the capacitated VRP (CVRP). Therefore, the mathematical model for the MOGVRP used in this thesis is developed on the classical CVRP’s mathematical model (Carić & Gold, 2008) and is presented as follows.

Variables:

For $i, j \in \{0, 1, \dots, n\}$ and $k \in \{1, 2, \dots, m\}$

$G(V, A)$	Representation of VRP as a graph
$V = \{v_0, v_1, v_2, \dots, v_n\}$	Set of vertices (v_0 : depot and v_i : customers)
$A = \{(v_i, v_j): i \neq j\}$	Set of arcs
$d_{ij} = (v_i, v_j)$	The Euclidean distance between the vertices v_i and v_j
q_i	The demand of the customer i
$K = \{k_1, k_2, \dots, k_m\}$	The vehicles fleet
Q	The capacity of each vehicle (homogenous VRP)
y_k	The remaining capacity of each vehicle
T_{max}	The upper bound of each vehicle tour

Decision Variables:

$$x_{ij}^k = \begin{cases} 1 & \text{if the vehicle } k \text{ visits the customer } j \text{ directly after the customer } i \\ 0 & \text{otherwise} \end{cases}$$

Objective functions:

$$f_1(X) = \text{minimize } \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} d_{ij} \cdot x_{ij}^k \quad (13)$$

$$f_2(X) = \text{minimize } \sum_{i \in V} \sum_{j \in V} c_0 \cdot \rho_{ij}^k d_{ij} \cdot x_{ij}^k \quad (14)$$

$$\text{where } \rho_{ij}^k = \rho_0 + \frac{\rho^* - \rho_0}{Q} w_{ij}^k \quad (15)$$

Here, c_0 is unit fuel cost, ρ_{ij} is the Fuel Consumption Rate (FCR) of the vehicle from customers i to j . ρ_0 is the empty vehicle FCR and ρ^* is the full-load vehicle FCR. w_{ij}^k is the load of the vehicle k while traveling from customers i to j . In the experimental

studies, the values of the parameters of c_0 , ρ_0 and ρ^* are taken as 1, 1 and 2 respectively since Xiao et al. used the same values for the same parameters (Xiao et al., 2012).

Constraints:

$$\sum_{k \in K} \sum_{j \in V} x_{ij}^k = 1 \quad \forall i \in V \quad (16)$$

$$\sum_{i \in V} x_{ip}^k - \sum_{j \in V} x_{pj}^k = 0 \quad \forall p \in V - \{v_0\}, k \in K \quad (17)$$

$$\sum_{j \in V - \{v_0\}} x_{0j}^k = 1 \quad \forall k \in K \quad (18)$$

$$\sum_{j \in V - \{v_0\}} x_{j0}^k = 1 \quad \forall k \in K \quad (19)$$

$$x_{ij}^k = 1 \Rightarrow y_k - q_i \geq 0 \quad \forall i, j \in V, k \in K \quad (20)$$

$$k = \emptyset \Rightarrow y_k = Q, \forall k \in K \quad (21)$$

$$\frac{\sum_{i \in V} q_i}{Q} \leq \sum_{k \in K} \sum_{j \in V} x_{0j}^k \leq |K| \quad (22)$$

$$\sum_{i \in V} \sum_{j \in V} d_{ij} \cdot x_{ij}^k \leq T_{max} \quad (23)$$

Constraint (16) ensures that each customer is visited only once. Constraint (17) guarantees that each vehicle visits the customers consecutively in its tour. Constraint (18) makes sure that each vehicle leaves the depot once and visits its first customer by leaving from the depot. Constraint (19) ensures that each vehicle returns to the depot only once and at the end of its tour. Constraint (20) guarantees that a vehicle's remaining capacity cannot be below zero. In other words, vehicles cannot contain the customers' demands which exceed the vehicle's remaining capacity. Constraint (21) states that every vehicle's initial capacity is the total capacity of the vehicle. Constraint

(22) guarantees that the number of vehicles cannot be lower than the ratio of total demands of the customers over the capacity of the vehicles and cannot be more than the cardinality of the set of all vehicles. Constraint (23) ensures that each vehicle's total tour length does not exceed the T_{max} defined in the problem which is the limit of the total travel time for a vehicle.

4.4. Solution Representation

In NSGA-II and ϵ -MOEA, the solutions are represented as genomes. In each genome, the genotype of the solution is stored in arrays. Each genotype's fitness values of total distance and total fuel consumption are the phenotype of the genome.

Each customer in a genome is represented by an index number from 1 to n , and the depot by 0. As a result, the vehicle tours in the solutions were represented by permutations of these index numbers in dynamic arrays. Two different types of representations of these permutations were used in solving the MOGVRP. The first one is the giant tour and the second one is the split vehicle tours. It was observed that both representations have pros and cons. For example, giant tour representation allows algorithms to apply any crossover and mutation operators easily. Yet, it does not give control over individual vehicles due to its splitting procedure. Customers cannot directly be moved between or within the vehicles. On the other hand, split vehicle tours representation makes these moves possible. It allows algorithms to interfere with each vehicle directly. However, classical crossover operators do not work with this type of representation and modifying these crossover operators according to this type of representation is quite difficult. Because of the limitations in both representations, both representations were used in the algorithms separately, and their results were compared in Chapter 5. The details of these representations are explained below.

In giant tour representation, all the customer indices were assigned to the giant tour permutation either randomly or by a constructive heuristic. The depot's index was not included in the permutation. Yet, the depot was intrinsically added in evaluating the vehicle tour fitness. The giant tour represents all the vehicle tours in order. In order to find these tours, the well-known Bellman Split algorithm was applied (Prins, 2004). The pseudo-code of the Bellman Split algorithm is presented in Figures 4.1 and 4.2.

```

V0 := 0
for i := 1 to n do Vi := +∞ endfor
for i := 1 to n do
  load := 0; cost := 0; j := i
  repeat
    load := load + qSj
    if i = j then
      cost := c0,Sj + dSj + cSj,0
    else
      cost := cost - cSj-1,0 + cSj-1,Sj + dSj + cSj,0
    endif
    if (load ≤ W) and (cost ≤ L) then
      //here substring Si...Sj corresponds to arc (i-1, j) in H
      if Vi-1 + cost < Vj then
        Vj := Vi-1 + cost
        Pj := i - 1
      endif
      j := j + 1
    endif
  until (j > n) or (load > W) or (cost > L)
endfor.

```

Figure 4.1 Bellman Split Algorithm: Splitting Procedure (Prins, 2004)

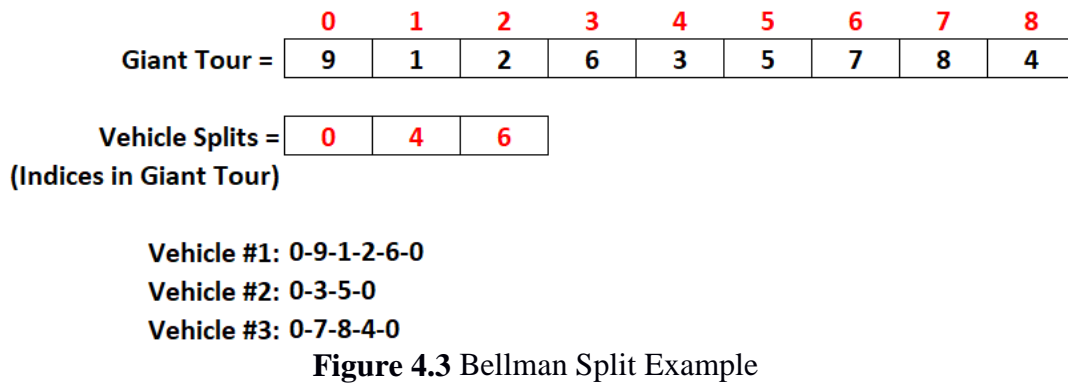
```

for i := 1 to n do trip(i) := ∅ endfor
t := 0
j := n
repeat
  t := t + 1
  i := Pj
  for k := i + 1 to j do enqueue(trip(t), Sk) endfor
  j := i
until i = 0.

```

Figure 4.2 Bellman Split Algorithm: Extracting Vehicles Procedure (Prins, 2004)

Bellman Split algorithm returns the splitting points (i.e. index numbers of the giant tour) for each vehicle on the giant tour and these indices are stored in another dynamic array (Figure 4.3).



Bellman Split algorithm begins with the first customer in the giant tour and hypothetically assigns it in the first vehicle. For the following customers in the giant tour, it decides on whether this customer should belong to the current vehicle or a new vehicle. If the current customer's load is greater than the remaining capacity of the current vehicle or it increases the total distance cost of all vehicles when included to the current vehicle more than being included to a new vehicle, then Bellman Split algorithm hypothetically assigns this customer to a new vehicle, otherwise the customer is considered in the vehicle. Each time Bellman Split decides on starting a new vehicle it holds the index of the last customer in a temporary array by which it later extracts the vehicle split indices on the giant tour.

Vehicle split representation, on the other hand, directly uses separate vehicle arrays. There is no giant tour array and each vehicle is represented by a separate individual dynamic array. This way, each vehicle can be accessed, and their customers can be moved between or within the vehicles directly. As a result, mutation operators can easily be applied to these vehicle tours. However, applying crossover on two genomes cannot be done as directly as in mutation operators. Since there is no one giant tour, selecting different customers from different genomes causes permutation problems. Controlling the repeating or missing customers during and after crossover operation requires checking all the other vehicle permutations in each genome as well. And, this increases the time complexity of the crossover operation. In this thesis, for this split vehicle representation, no crossover operation was used. Instead, Path-relinking heuristic was modified and applied to substitute the crossover operation. After, path-relinking, the mutation operators were applied to the vehicles.

The experimental results with both representations are presented in Chapter 5. It has been observed that split vehicle representation produced more promising results.

4.5. Constructive Heuristics

Twelve constructive heuristics were used in initial population generation for the algorithms in this thesis. Seven of them favor minimizing the distance objective while the other five favor minimizing the fuel consumption objective in making decisions. These constructive heuristics are Clark and Wright Savings (C&W Savings) algorithm, a modified version of the Apparent Tardiness Cost with Setups (ATCS) Rule used in scheduling problems, a modified version of the Somhom Competition Rule used in multiple TSP (mTSP), Insert End All, Assigning Heaviest Load First (AHLF), Nearest Neighbor algorithm, Random Task (RT), and RT-Flower (RTF). The first five of these heuristics were both applied for minimizing the distance and minimizing the fuel consumption objectives with appropriate modifications. The last two of these constructive heuristics were applied only favoring minimizing the distance objective function.

C&W Savings heuristic algorithm (Clarke & Wright, 1964) starts with considering the worst-case situation in the VRP. That is, assigning a vehicle for each customer. Then it iteratively tries to combine the customers into the same vehicle under the VRP constraints and checks how much saving this new route produces regarding distance. Therefore, the algorithm tries to assign customers to vehicles that will produce the best savings. The mathematical formula of the C&W Savings is as follows:

$$s_{ij} = d_{i0} + d_{0j} - d_{ij}, \text{ for all } i, j > 0 \text{ and } i \neq j \quad (24)$$

s_{ij} : distance savings when customer j is added in the vehicle tour

d_{ij} : distance between the customers i and j

ATCS Rule heuristic algorithm (Ow & Morton, 1989) was developed for minimizing the total early and tardy costs of a single machine in scheduling problems. In this thesis, it was adapted for VRP. This adapted algorithm basically tries to add each customer at the end of each vehicle in turn. Then it calculates the estimated distance difference. If this distance is below the average value, then it adds the new customer to the current vehicle, otherwise, it starts a new vehicle and adds this customer to it. The mathematical formula of the ATCS Rule is as follows:

$$h_{ij} = d_{last,j} \cdot e^{\frac{T_i}{\bar{T}}} \quad (25)$$

h_{ij} : priority value when customer j is added to the end of the vehicle's tour

$d_{last,j}$: distance between the last customer in vehicle tour and customer j

T_i : total tour length of the vehicle i

\bar{T} : $\frac{\sum_{i=1}^n T_i}{n}$, where n is the number of vehicles

Somhom Competition Rule is like ATCS rule with a difference in the multiplier. The following is the mathematical formula of Somhom Competition Rule:

$$h_{ij} = d_{last,j} \cdot \left(1 + \frac{T_i - \bar{T}}{\bar{T}}\right) \quad (26)$$

h_{ij} : priority value when customer j is added to the end of the vehicle's tour

$d_{last,j}$: distance between the last customer in vehicle tour and customer j

T_i : total tour length of the vehicle i

$\bar{T} = \frac{\sum_{i=1}^n T_i}{n}$, where n is the number of vehicles

Insert End All constructive heuristic tries to add the new customer to a vehicle by trying to add it to the end of each vehicle and finding the minimum cost in doing that.

The following is the mathematical formula of this heuristic:

$$\Delta = d_{last,j} + d_{j,0} - d_{last,0} \quad (27)$$

Δ : distance difference when customer j is added to the end of the vehicle's tour

$d_{last,j}$: distance between the last customer in vehicle tour and customer j

These four constructive heuristics mentioned above were applied with distance and fuel consumption calculations separately. For fuel consumption calculations, the

distance calculations in the formulas above were replaced with the fuel consumption formula (Eq. 13). This way, it was aimed to generate initial solutions favoring either one of the objective functions in the problem.

Assigning Heaviest Load First (AHLF) is a greedy method that assigns the customers to vehicles with regards to their demands by sorting them in descending order. In this sense, this heuristic favors minimizing the fuel consumption objective. After sorting the customers in descending order according to their demands, each customer is assigned to the vehicles in their order. If there is more than one customer having the same demand, then the nearest one to the previous customer (or depot if it is the first customer) is chosen. If the customer's demand exceeds the capacity of the current customer, then a new vehicle is started.

The nearest Neighbor algorithm, RT and RTF heuristics (Prins, Labadi, & Reghioui, 2009) are greedy algorithms based on the distance calculations. In the Nearest Neighbor method, the first customer of each vehicle is selected by finding the nearest customers to the depot. Then, the rest of the customers in a vehicle is selected by finding the nearest unassigned customer to the last customer in the vehicle. If there is more than one candidate of nearest neighbors, then the smaller index numbered customer is selected. When the next nearest customer's demand exceeds the remaining capacity of the current vehicle then this customer is not assigned to any vehicle and the whole process is done by starting a new vehicle. The algorithm ends when there are no more customers to assign to a vehicle.

RT and RTF were developed on the Nearest Neighbor method. RT works the same way with Nearest Neighbor except for including a decision making step when there is more than one candidate of nearest neighbors. Unlike, Nearest Neighbor algorithm, RT gives a chance to each candidate by storing them in a dynamic array. RT, then, decides on which one to choose randomly. In order to increase the number of candidates, a threshold may be added in comparing the distances of the neighbors. Hence, RT provides more variety than Nearest Neighbor in generating the initial population.

RTF, on the other hand, adds an intuition into decision making between the nearest customer candidates. It divides the unvisited customers into two sets every time the nearest customers are being searched. The first set (i.e. L_1) contains the unvisited nearest customers that drive the vehicle away from the depot. The second set (i.e. L_2)

contains the rest of the unvisited customers. RTF, then, decides between these two sets in order to find the next customer to assign the current vehicle. If one of these sets are empty, then RTF randomly selects a customer from the other set. Otherwise, if the vehicle's total load is less than or equal to the half of its full capacity, then the next customer is selected from the set L_1 , and from the set L_2 otherwise. RTF repeats this process until all the customers are assigned to vehicles. As a result, RTF first tries to assign the nearest customers which drive the vehicles away from the depot until the vehicle reaches to its half-capacity. Then, it tries to assign the rest of the customers which drive the vehicle closer to the depot.

4.6. Local Search Heuristics

Two different local search algorithms are used for the giant tour representation: 2-Opt with best improvement pivot rule and All Insertion Heuristic. The former is applied to favor the minimizing the total distance objective, while the latter one is applied favoring both objectives. 2-Opt algorithm is a state-of-the-art local search algorithm applied to Travelling Salesman Problems (TSPs) in general. In 2-Opt, every binary combination of the customers in the giant tour is selected and the block of customers between them (including these two customers) is inverted. This way it is desired to fix the unnecessary crossroads in the tours as it is seen in Figure 4.4.

Then, the giant tour was split into vehicles and the total distance cost is calculated. In calculating the total distance, a speed up method is used. This method just calculates the difference of the distance after the change of the customers and adds this difference to the current total distance. If this new vehicle tour permutation decreases the total distance value, then this new permutation replaces the current permutation. Otherwise, the giant tour permutation remained unchanged and the next two edges in the giant tour were selected and the same procedure is applied to them. This process in 2-Opt continues until no further improvements can be made. 2-Opt on giant tour causes customer exchanges between the vehicles due to its split procedure. Because of this reason, it helped both the exploration and exploitation of the algorithms.

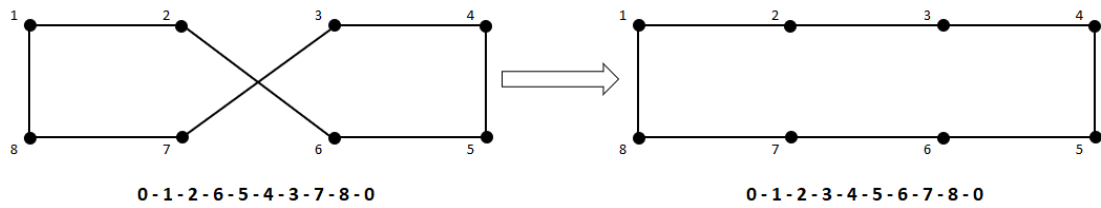


Figure 4.4 2-Opt example

In All Insertion Heuristic (AIH), every customer in the giant tour is removed from its current location and is inserted to every possible location in the giant tour. In each reinsertion, the giant tour is split into vehicles and the fitness values for both objectives are calculated. Then, the current and new permutations are compared according to the dominance comparison. If the new permutation dominates the current one, then the new permutation replaces the current one. If the new permutation is dominated by the current one, then the algorithm continues with the current permutation. If the new permutation and the current permutations are non-dominated, then the new permutation is added to the population as a new solution, but the local search continues with the current permutation. AIH algorithm is run until all the customers in the permutation are tried to be inserted in every possible location in the giant tour.

For the split vehicle tour representation, four different local search algorithms are used: Cross Exchange, 2-Opt with first improvement pivoting rule, Emptying the Lighter Vehicles (ELV), Moving Customers Between and Within Vehicles (MCBWW). The first one is applied to favor minimizing the total distance objective. The other three are applied to favor both objectives.

Cross Exchange method (Taillard, Badeau, Gendreau, Guertin, & Potvin, 1997) searches the solution space by exchanging the blocks of customers between two vehicle tours in a solution. The size of the blocks (i.e. L) on each vehicle tour is bounded above by a distance length determined by the decision maker. The number of customers in each block might be different but the total distance length of these blocks is less than or equal to L . Cross Exchange aims to find the best solution during the search among the neighborhoods of the initial vehicle tours (Figure 4.5).

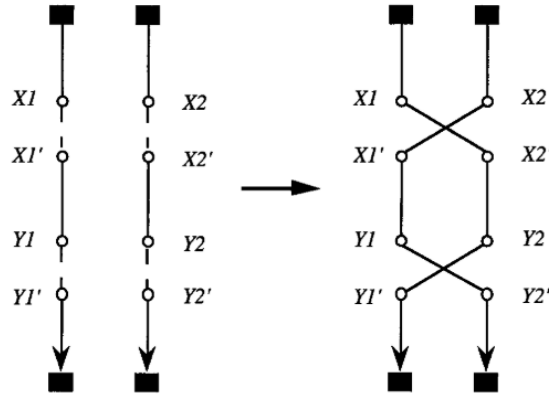


Figure 4.5 Cross Exchange (Taillard et al., 1997)

In the algorithms used in this thesis, Cross Exchange method is adjusted for the MOP structure. That is, in each searching phase, all the produced solutions are stored in a non-dominated archive population which is the outcome of Cross Exchange. The distance length L is selected in a way proportional to the size of the problem instance. Two vehicles are randomly selected for Cross Exchange. Cross Exchange is applied to these vehicles with all the possible L sized block of customers from the first to the end of each vehicle tour.

2-Opt with first improvement pivoting rule is applied to each individual vehicle tour permutation independently. The same procedure in 2-Opt applied on the giant tour is used for individual vehicles in the solution. This way 2-Opt on split vehicle tours provides only exploitation in the algorithms.

ELV aims to improve the fitness values for both objectives by trying to reduce the number of vehicles in the solution. It tries to empty the least loaded vehicles if possible. Since ELV aims to favor both objectives, it used a temporary local archive in which it keeps all the non-dominated results generated during the process. In ELV, first, the original solution is added into this temporary local archive and duplicated in order to find new solutions without changing the original solution. Then all the vehicles in this duplicated copy are sorted according to their remaining capacities in descending order and assigned to a temporary list. In other words, the lighter vehicles (or the vehicles that have more remaining capacities) are placed at the beginning of the list. Starting from the first vehicle in the list, all the customers in this vehicle are tried to be moved to the other vehicles one by one. Then, the distance differences caused by these trials are calculated. After minimum distance difference is found, the customer is added into the vehicle that gave the minimum distance. The solution obtained by this move is

compared with the solutions in the local archive according to dominance criteria. If it dominates any solution in the local archive, then these dominated solutions are removed from the local archive and this new solution is added to the local archive. If it is dominated by any solution in the local archive, then it is not added to the local archive. If it is non-dominated with all the solutions in the local archive, then it is added to the local archive. After this dominance comparison, ELV proceeds with the next customer in the vehicle. If no customer is left in the vehicle, then the current vehicle is deleted and the next vehicle in the list is tried. If the remaining customers in the current vehicle cannot be added to any other vehicle after all tries, then the whole process is repeated with the next vehicle in the temporary list. ELV terminates when all the vehicles in the temporary list are tried. In the end, there might be some vehicles emptied by moving their customers to other vehicles and removed from the solution. Or, even if there is no change in the number of vehicles, the vehicle tours might have been improved by these changes. As a result, ELV returns a temporary local archive that contains a set of non-dominated solutions produced by ELV.

MCBWW also uses a local archive in order to keep all the non-dominated solutions during the process. MCBWW starts with finding the vehicle with the highest distance and then tries to move its customers to every possible location within the same vehicle and in other vehicles. In each remove and insertion move, the generated new solution is compared with the current one according to the dominance criterion. If the current solution dominates the new solution, then the algorithm continues with the next remove and insertion move. Otherwise, if the new solution dominates the current one or they are non-dominated, then the new solution is added to the local archive and the current solution is replaced with the new solution. The algorithm continues until there is no improvement or all the customers in the current vehicle are removed. MCBWW tries to find better and non-dominated solutions by improving the worst vehicle with regards to the distance objective and balance the tour distances of all vehicles.

4.7. Path-Relinking Heuristic

Path Relinking (PR) algorithm (Glover, Laguna, & Marti, 2000) aims to find better solutions by searching among the trajectories of the elite solutions produced during the search of the algorithm. It is applied to two selected solutions named *initial* and *guiding* solutions. It generates a path from the initial to the guiding solution with the

hope of obtaining improved results. First, it finds the common path sequences in both solutions based on the principle that good solutions are likely to share common characteristics (Hof & Schneider, 2019). These paths are commonly shared paths by both the initial and the guiding solutions. It, then, runs through each path in the guiding solution that is not in this common set of paths and transfers the paths in the initial solution like the ones in the guiding solution by removing and inserting the elements in the initial solution. At each of these moves, a new candidate solution is produced and evaluated, and the initial solution is gradually transferred into the guiding solution (Figure 4.6). PR runs until all the uncommon paths in the guiding solution are run through and the related changes are made in the initial solution. The best solution generated during the PR process is carried on to the next generation of solutions.

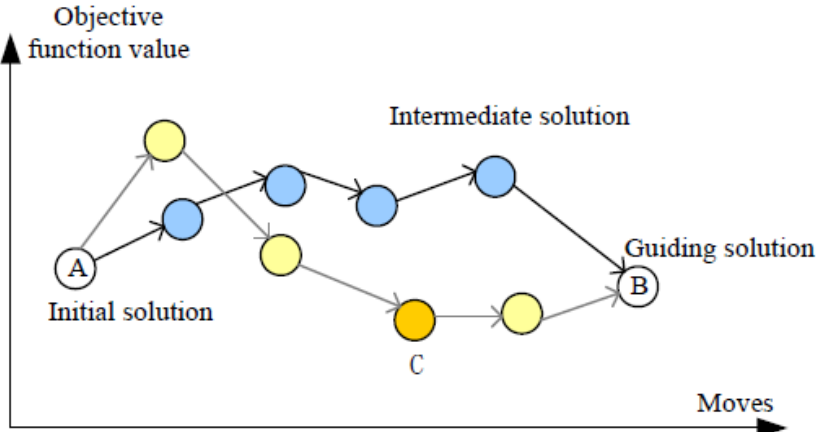


Figure 4.6 Path Relinking (Zhang, Bai, & Dong, 2010)

PR algorithm is adapted for the MOP structure in this thesis. That is, for each move in PR, the generated intermediate solutions are evaluated according to the dominance criteria and stored in a temporary non-dominance archive through the whole process of PR. As a result, PR produces a set of non-dominated solutions. PR is used with split vehicle representation. Since no crossover was used in NSGA-II and ϵ -MOEA, PR functioned like a crossover operator and produced children for the population in the MOGVRP in this thesis.

Select the initial and guiding solutions
Initialize the temporary local archive
Find all edges in the initial solution that are different than the edges in the guiding solution and assign them into the uncommon_edges list
Sort the uncommon_edges list according to the length of the edges in decreasing order
Repeat
 Select the next edge from the uncommon_edges list
 Find the endpoints (customers) of the edge in the initial solution
 Remove and insert the first endpoint before and after the second endpoint in the initial solution
 Remove and insert the second endpoint before and after the first endpoint in the initial solution
 Evaluate all these recently produced children
 Add the non-dominated children in the temporary local archive
 Discard the dominated children
 Replace the initial solution with one of these recently produced and non-dominated children
Until every edge in uncommon_edges is applied
 Return the temporary local archive

Figure 4.7 Pseudocode for adapted PR with MOP structure

4.8. Reproduction Operators

Different reproduction procedures are applied for two different solution representations. In the giant tour representation, Partially Mapped Crossover (PMX), Ordered Crossover (OX), Cycle Crossover (CX), Two-point Crossover (TPX), and mutations by swapping, insertion, and inverting are used.

In PMX (Goldberg & Lingle, 1985), two random crossover points are selected on two parent giant tours and the customers between these crossover points are copied to two new offspring. Then, a mapping is made between these copied sub-tours. The missing customers in each offspring are selected from the other parent depending on this mapping.

As in PMX, two random crossover points are selected and the related sub-tours are transferred to the offspring in OX (Davis, 1985). The rest of the customers in each offspring, however, are selected in order from the other parent. If the customer selected from the other parent already exists in the offspring, then it is skipped and the next customer in the parent permutation is checked. This way the repetitions of the customers are prevented. If the last customer in the parent tour is reached and the offspring is not completed yet, then the checking and adding the customers from the parent tour continues with the beginning of the parent tour.

In CX (Oliver, Smith, & Holland, 1987), no crossover points are selected. A cycle between the two parents is found first. Finding the cycle begins with the first customers in each parent tour. The first customers in both parent tours are added to the cycle respectively. Then, the customer in the second parent tour is searched in the first parent tour. When it is found, its index number in the first parent tour is used to find the next customer in the second parent tour. The customer with that index number in the second tour is added to the cycle. The same process is repeated with this recently added customer. Once its index number is found in the second parent, this index number is used to find the next customer in the first parent tour. The customer located at this index in the first parent tour is added to the cycle. This process goes back and forth between the two parents until the cycle is closed. Then, the customers in the first parent tour are checked with this cycle. If the customer is in the cycle, then it is directly transferred to the offspring with the same location in the tour. Otherwise, that customer is transferred to the offspring with its location in the second parent tour. The same process is applied on the second parent tour in order to produce the second offspring.

In TPX (Ishibuchi & Murata, 1998), two crossover points are selected first. Then, the customers from the beginning to the first crossover point and from the second crossover point to the end in each parent tour are transferred to two new offspring with the same locations. The rest of the customers are selected from the other parent tour and located in between the two crossover points in each offspring. If the customer to be added already exists in the offspring, then it is skipped. This process continues until all the customers are completed in both offspring.

In swap mutation move, any two customers in the giant tour are randomly selected and swapped. In insertion mutation move, a randomly selected customer is removed and inserted to a different location in the giant tour. In invert mutation move, two customers are randomly selected and all the customers between them are inverted (Sivanandam & Deepa, 2010).

In split vehicle tour representation, no crossover operator was used due to the structure of the representation. None of the crossover operators that were used for the giant tour is applicable to split vehicle representation. Therefore, only some of the mutation operators are used on the split vehicle representation.

The very same mutation operators used in the giant tour representation are also used in the split vehicle representation. Yet, they are applied to the individual vehicle tours since there is no giant tour in this representation. In addition, mutations by swapping and inserting are applied both within the individual vehicle tours and between randomly selected two vehicles.

When swapping is applied on the same vehicle, two randomly selected customers are swapped in the same vehicle. When it was applied to two different vehicles, then two randomly selected customers from two different vehicles were swapped. This swap operator is also applied as swapping a block of customers between two vehicles. In this type of swap, the same number of customers in each block are chosen and only the blocks that did not violate the T-max and capacity constraints are swapped, i.e., only feasible swap moves are made.

When insertion mutation is applied within an individual vehicle, a randomly selected customer is removed and reinserted in a different location in the same vehicle. When it is applied to two different vehicles, a randomly selected customer is removed from a randomly selected vehicle and is inserted in a random location in the another randomly selected vehicle. As in swapping mutation, insertion mutation is applied with removing a block of customers from one vehicle and inserting this block in the second vehicle. T-max and capacity constraints are also checked in order to prevent infeasible solutions.

The inverting mutation is applied in the same way in giant tour representation except it is only applied on the individual vehicle tours.

4.9. NSGA-II and ϵ -MOEA Used in the Thesis

NSGA-II and ϵ -MOEA are utilized for the solution of the MOGVRP studied in this thesis. They both use the constructive heuristics, local search moves and reproduction operators with the related solution representations as they are mentioned in the previous subsections. The general structures of both algorithms are given below. The differences in the algorithms for two different solution representations (i.e. giant tour and split vehicle representations) are explained in the following paragraphs.

Pseudo-code for the adapted NSGA-II used in this thesis is given below:

Generate individuals in the initial population by using constructive heuristics and by generating random permutations
Calculate the fitness values of the initial population and rank each individual with 0
Apply local search on the initial population

Repeat

Select individuals for reproduction
Reproduce offspring
Apply local search on the offspring
Evaluate the offspring
Add the offspring to the population by using non-dominance criteria
Find fronts of the combined population by fast non-dominated sorting
Sort the combined population by crowding distance
Truncate the population

Until *all the iterations are done*

Return the Pareto front

Figure 4.8 Pseudo-code of the adapted NSGA-II for the MOGVRP

Each component of this algorithm was explained more in detail in the previous subsections. Hence the unmentioned details will be given here. Selecting individuals for reproduction is done by a modified binary tournament selection method. In classical binary tournament selection, two individuals randomly selected from the population are compared and the better individual with regards to the fitness value is selected for recombination. In this study, this tournament comparison is done according to the non-dominance criteria of the selected individuals. If one individual dominates the other one, then the dominating one is selected. If they were non-dominated, then one of them was randomly selected.

Reproducing offspring is done differently in different solution representations. In the giant tour representation, the crossover and mutation operators for the giant tour mentioned in Section 4.8 are used. In split vehicle representation, Path Relinking (Section 4.7) and the mutation operators for split vehicle representation mentioned in Section 4.8 are used.

The common and different constructive heuristics and local searches are also applied as mentioned in sections 4.5 and 4.6.

Pseudo-code for the adapted ϵ -MOEA used in this thesis is given below:

Generate some individuals in the initial population by using constructive heuristics and the rest randomly
Create an empty non-dominated archive
Calculate the fitness values of the initial population
Evaluate the initial population
Update the archive using the initial population by ϵ -dominance
Apply local search on the initial population
Evaluate the new individuals generated by local search
Update the archive with the generated members by the local search by ϵ -dominance
Repeat
 Select individuals for reproduction
 Reproduce offspring
 Apply local search on the offspring
 Evaluate the offspring
 Add offspring to the population by non-dominance criteria
 Add offspring to the archive by ϵ -dominance
Until *all the iterations are done*
Return the archive

Figure 4.9 Pseudo-code of the adapted ϵ -MOEA for the MOGVPR

Each component of this algorithm is explained in more detail in the previous subsections, as well. However, selecting parents for reproduction is done differently than NSGA-II. That is, one individual is selected from the population and the other one is selected from the archive for reproduction in ϵ -MOEA. Hence binary tournament selection is not used. Reproducing offspring, applying constructive heuristics and local search is done in the same way as it is in NSGA-II. The major difference in ϵ -MOEA is that it uses an archive which holds the non-dominated solutions throughout the algorithm. And, population and archive are both updated after each reproduction and local search process.

CHAPTER 5

EXPERIMENTAL STUDIES

All the algorithms and methods used in this study are coded in C++ programming language with Microsoft Visual Studio Enterprise 2017, version 15.9.11. These algorithms and methods are coded accordingly for each solution representation and their results were obtained separately. First, the giant tour representation version was programmed, and its results were produced. Since these obtained results were not promising with regards to producing Pareto fronts, the solution representation was switched to split vehicle representation. It was observed that the results obtained with the split vehicle representation were more promising than the results of giant tour representation. All these results are presented below.

Seven instances from Christofides (Christofides et al., 1979) and twenty instances from Golden (Golden et al., 1998) instances were used in the experimental studies. In Christofides instances (i.e. CMT1, CMT2, CMT3, CMT4, CMT5, CMT11, and CMT12), the depot and the customers are randomly scattered in the solution space. Each instance is different in size with regards to the number of customers and has a T-max value for the vehicle tours. The number of customers in these instances are 50, 75, 100, 150, 199, 120 and 100 respectively.

In Golden instances, on the other hand, the customers and the depot are laid out in the solution space within an order. These twenty instances can be categorized into five sets (Figure 5.1). In each set, the distribution of the customers is the same, but the number of total customers is different. These sets are {Golden_1, Golden_2, Golden_3, Golden_4}, {Golden_5, Golden_6, Golden_7, Golden_8}, {Golden_9, Golden_10, Golden_11, Golden_12}, {Golden_13, Golden_14, Golden_15, Golden_16}, and {Golden_17, Golden_18, Golden_19, Golden_20} (Figure 5.1). The number of customers in each Golden instance is {240, 320, 400, 480}, {200, 280, 360, 440}, {255, 323, 399, 483}, {252, 320, 396, 480}, and {240, 300, 360, 420}.

In the experimental studies with the giant tour representation, only the first instances of each Golden instance set were used. In the experimental studies with the split vehicle tour representation, all the Golden instances were used.

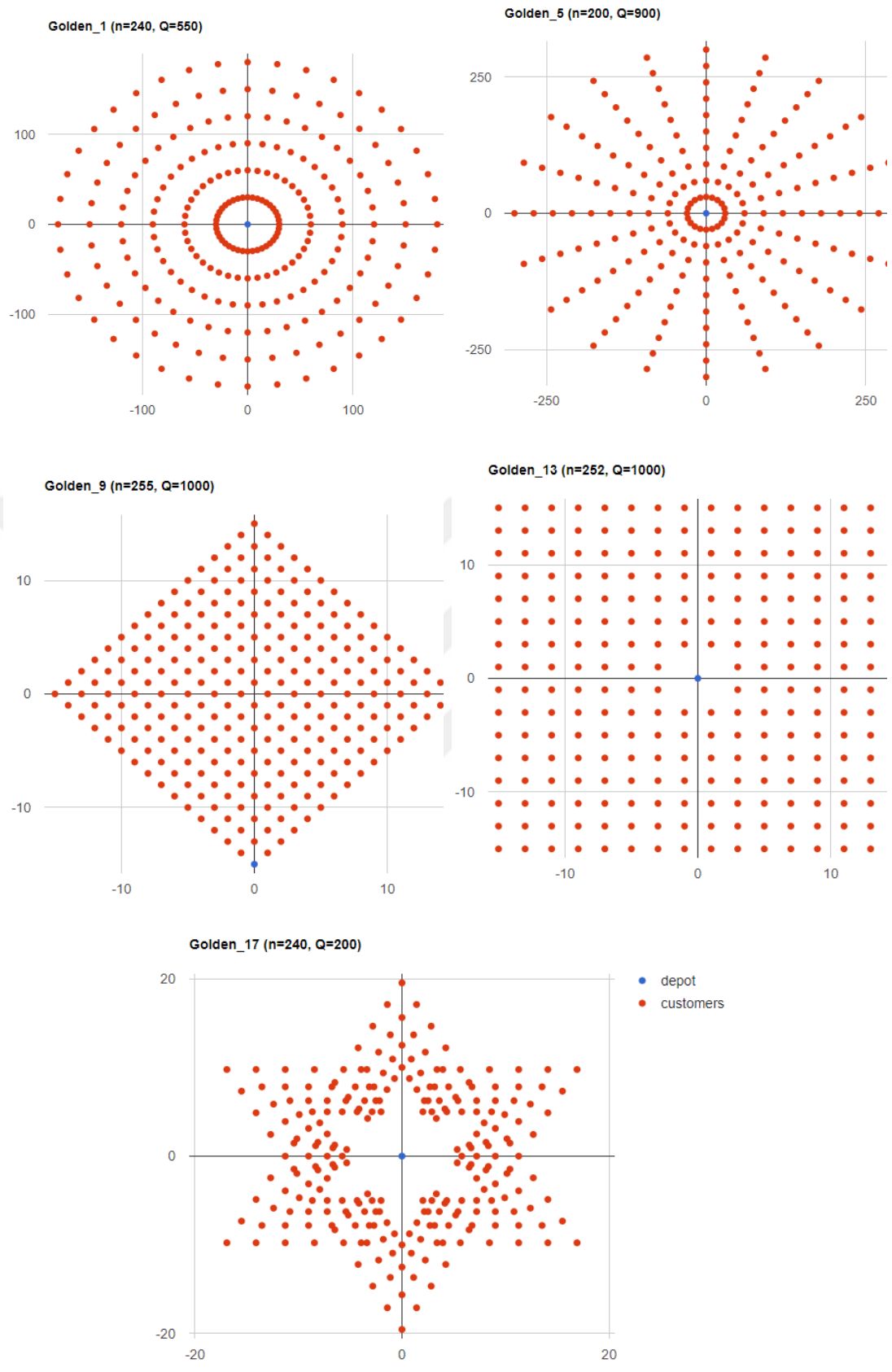


Figure 5.1 Golden instance sets (Xavier et al., 1998)

The results that are obtained in the experimental studies are compared with the results obtained by Xiao et al. (2012). In their article, Xiao et al. obtained their results for two objectives as SOP results. In this thesis, the same two objectives are used as it is in the study of Xiao et al. (2012), and the best results of Xiao et al. are taken into consideration as their Pareto front.

5.1. Results with Giant Tour Solution Representation

After various combinations were tried with the giant tour in the experimental studies, those parameters observed to be producing better Pareto fronts are: population size = 500, crossover rate = 70% and mutation rate = 50%. Termination criteria for the algorithms is defined with the number of iterations, which was 10000. Each algorithm is run ten times on each instance and the best Pareto results obtained by all these ten runs are considered as the results of the algorithms.

The following figures show the Pareto fronts obtained by the adapted NSGA-II and the adapted ϵ -MOEA using the giant tour representation. Their numerical results are laid out as tables in Appendix 1. In the following figures, the blue diamonds are the results obtained by Xiao et al. (2012), the red squares contain the Pareto front obtained by the adapted NSGA-II and the green triangles contain the Pareto front obtained by the adapted ϵ -MOEA used in this thesis.

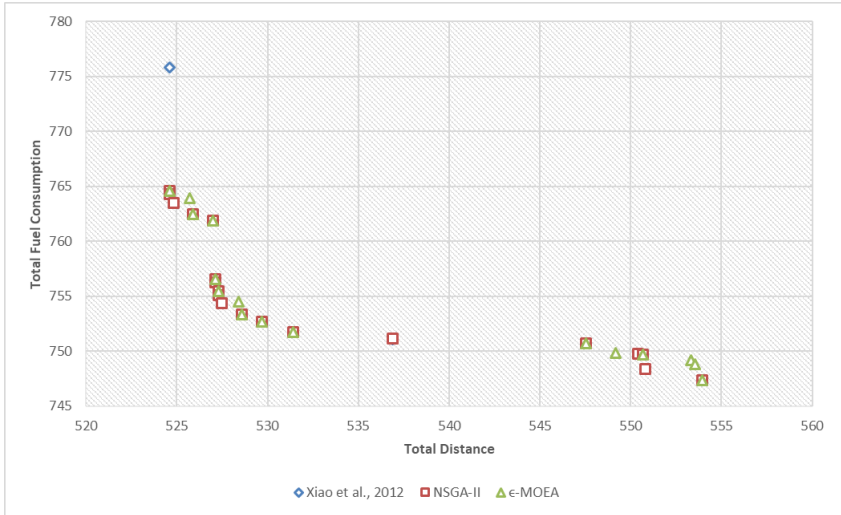


Figure 5.2 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT1

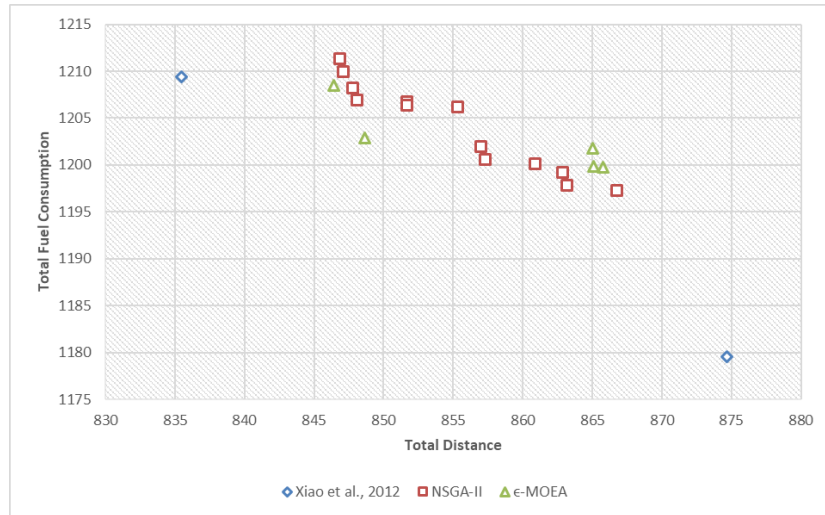


Figure 5.3 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT2

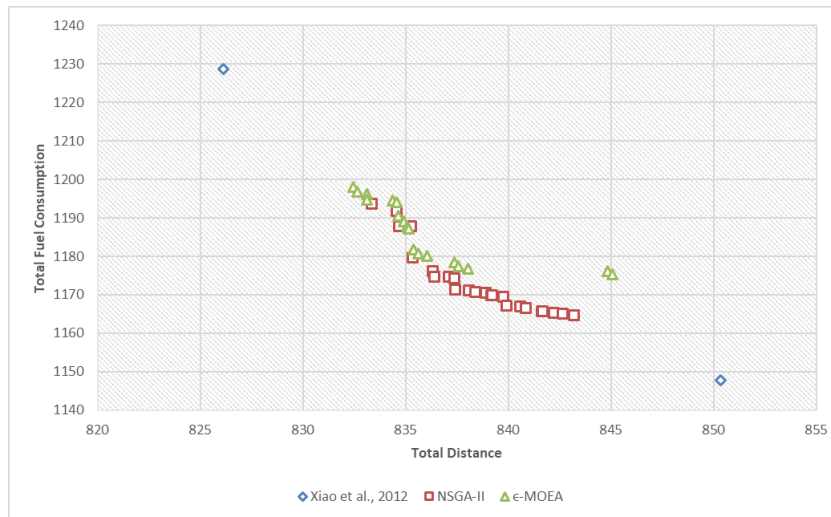


Figure 5.4 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT3

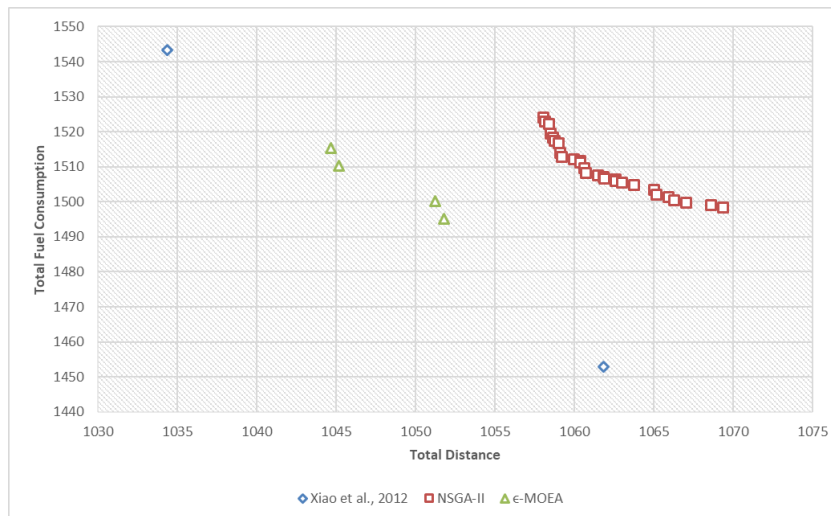


Figure 5.5 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT4

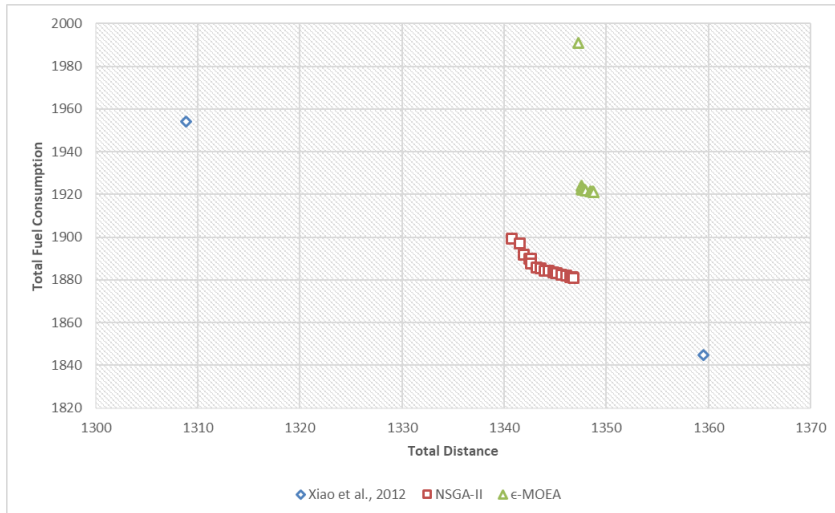


Figure 5.6 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT5

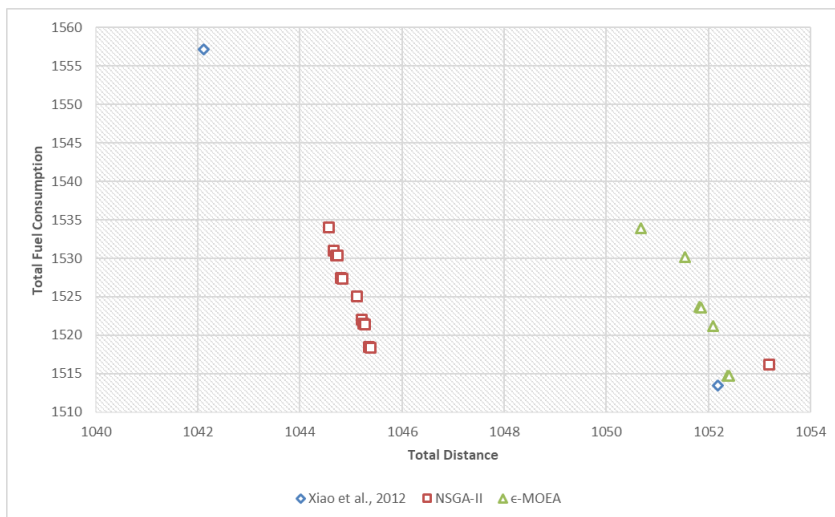


Figure 5.7 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT11

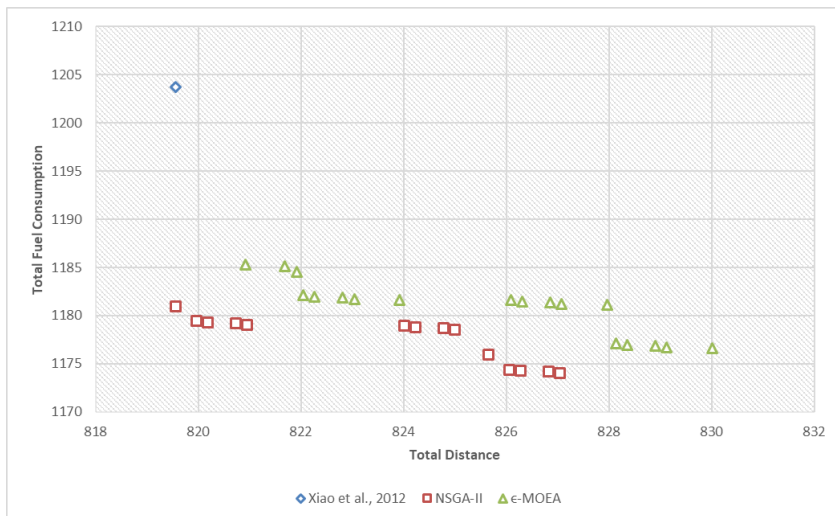


Figure 5.8 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT12

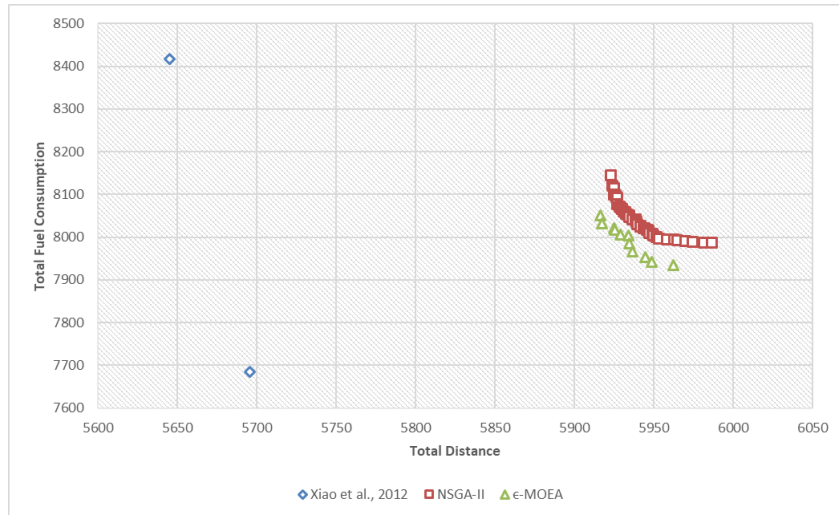


Figure 5.9 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_1

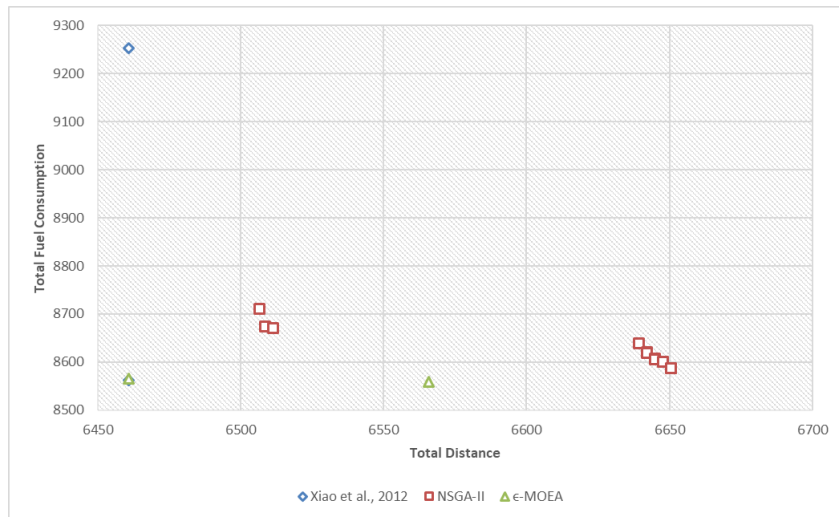


Figure 5.10 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_5

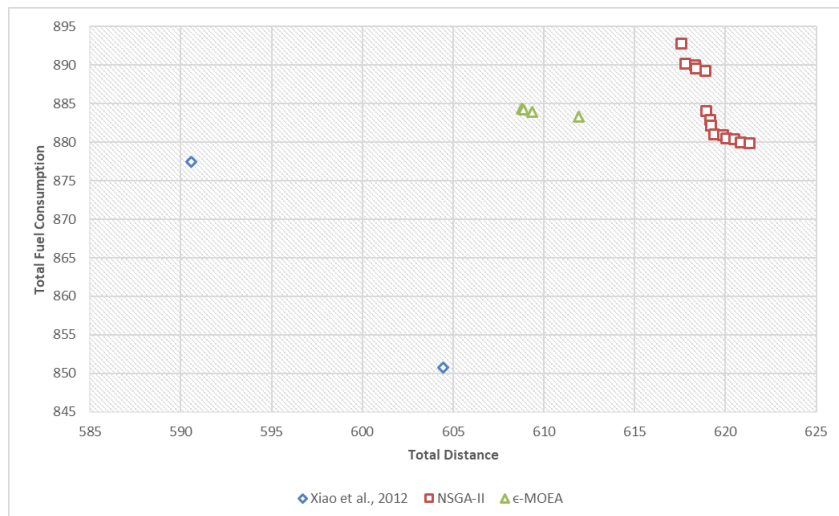


Figure 5.11 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_9

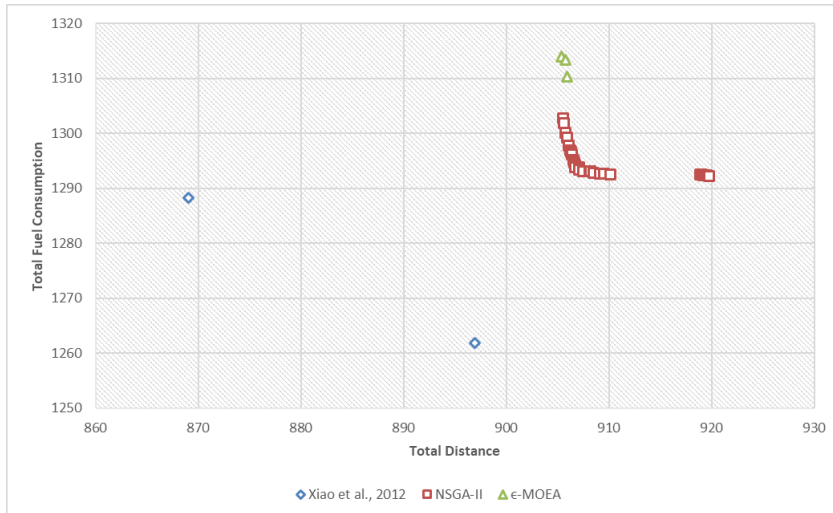


Figure 5.12 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_13

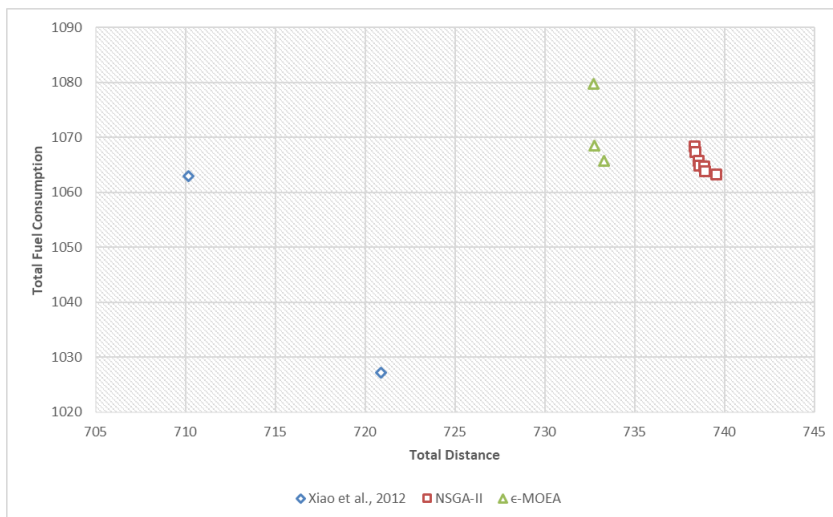


Figure 5.13 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_17

As it is seen in the figures, although promising Pareto fronts were generally obtained for the Christofides instances, the Pareto results obtained for Golden instances were dominated by the results of Xiao et al. (2012). Therefore, the solution representation was switched to split vehicle representation. Their results are explained in the next subsection.

5.2. Results with Split Vehicle Tour Solution Representation

As in giant tour representation, various experimental studies are done with split vehicle representation with different parameters for the methods used in the algorithms in the thesis. Three of these parameters are observed to be significant and affect the results of the algorithms directly. These are the population size (i.e. popSize), the divisor that

adjusts the length of the block depending on the number of the customers in cross exchange (i.e. $divN$), and the second divisor that adjusts the size of the block which is moved to another vehicle in one of the mutation operators (i.e. $blockSizeDivisor$).

A full factorial design of experiment (DoE) approach is applied with different ranges for each of these three parameters for the adapted NSGA-II and the adapted ϵ -MOEA. Since there are infinitely large different possibilities for each of these parameters, the following values are chosen for DoE after some initial experiments: $popSize = 400, 500, 600$, $divN = 2, 3$ and $blockSizeDivisor = 6, 7$. These parameters generated 12 different combinations (i.e. $3 \times 2 \times 2$) and each combination is run 10 times with 1000 iterations in each run on a different instance other than the Christofides and Golden instances. Li_22 instance (Li, Golden, & Wasil, 2005) was chosen for the DoE runs. The reason for not choosing from the Christofides or Golden instances is to prevent the algorithms to adjust their parameters specific to the instances that are used for measuring performance. Three performance metrics of these 12 combinations obtained by the adapted NSGAII and the adapted ϵ -MOEA are evaluated using Minitab software. The main effects of these performance metrics and their interactions with one another are presented in the following figures.

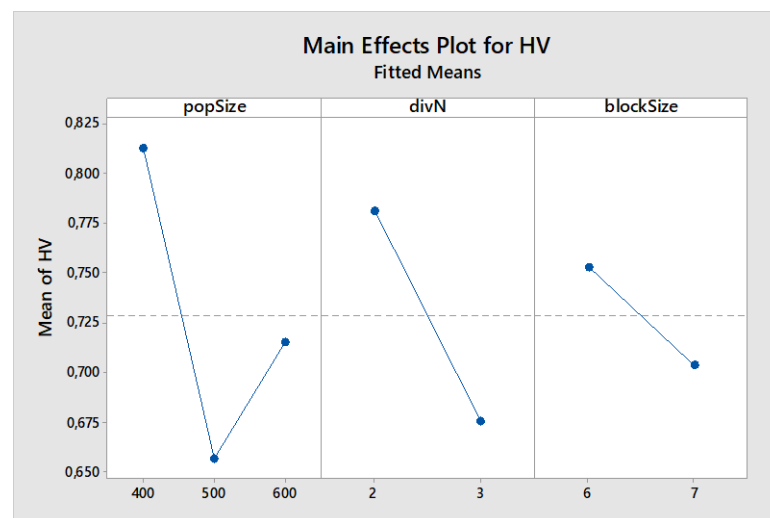


Figure 5.14 HV values of the DoE for the adapted NSGA-II

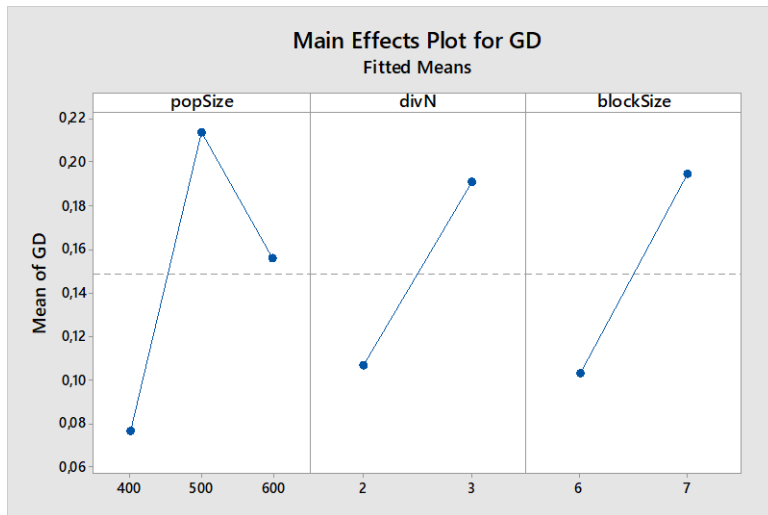


Figure 5.15 GD values of the DoE for the adapted NSGA-II

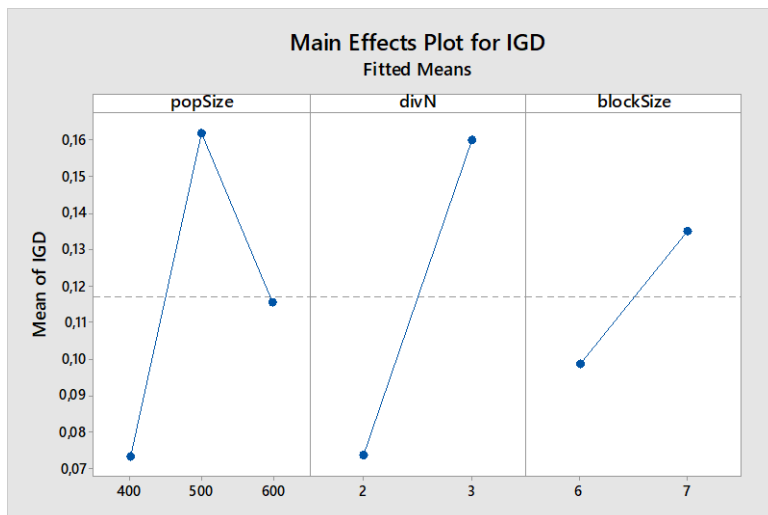


Figure 5.16 IGD values of the DoE for the adapted NSGA-II

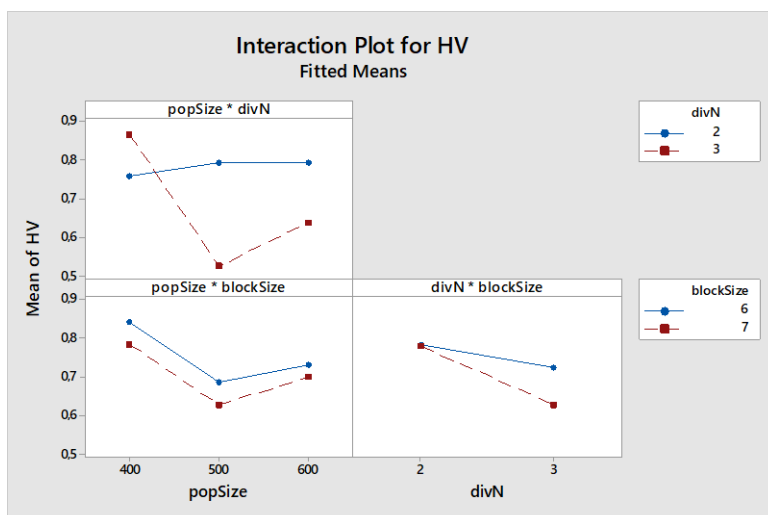


Figure 5.17 Interaction of DoE parameters according to HV for the adapted NSGA-II

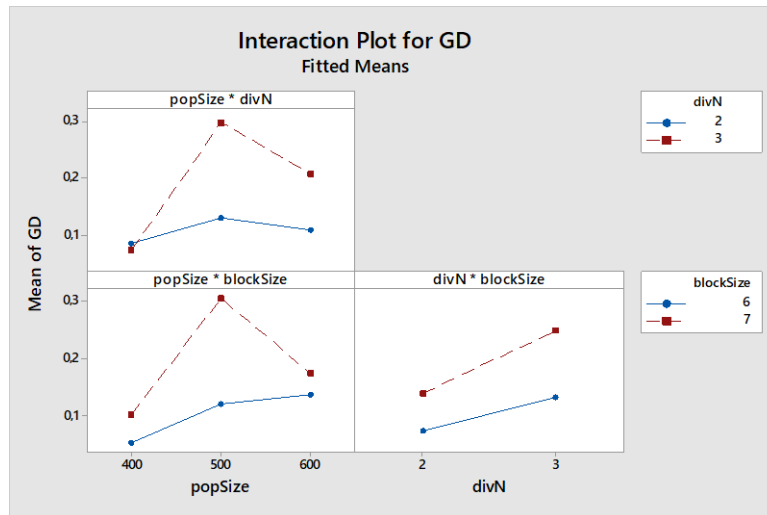


Figure 5.18 Interaction of DoE parameters according to GD for the adapted NSGA-II

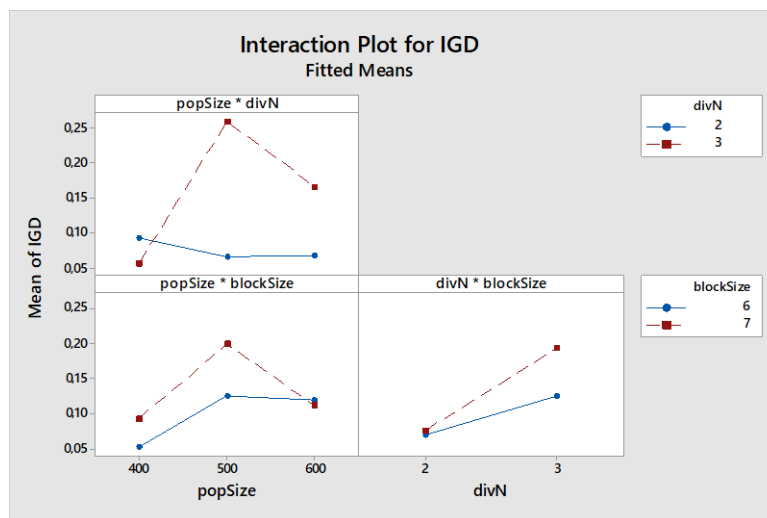


Figure 5.19 Interaction of DoE parameters according to IGD for the adapted NSGA-II

DoE results show that when each parameter was individually evaluated, the best combination for the adapted NSGA-II is obtained as popSize = 400, divN = 2 and blockSizeDivisor = 6. In the interaction plots, it was seen that popSize = 400, blockSizeDivisor = 6 and divN = 3, popSize = 400, and blockSizeDivisor = 6 and divN = 2 give the best results for two by two interactions. Although the interaction of popSize and divN favored the values of 400 and 3 respectively, the other interactions confirm the individual evaluation results. Therefore, popSize = 400, divN = 2 and blockSizeDivisor = 6 are used for the adapted NSGA-II for the final execution of the experimental results.

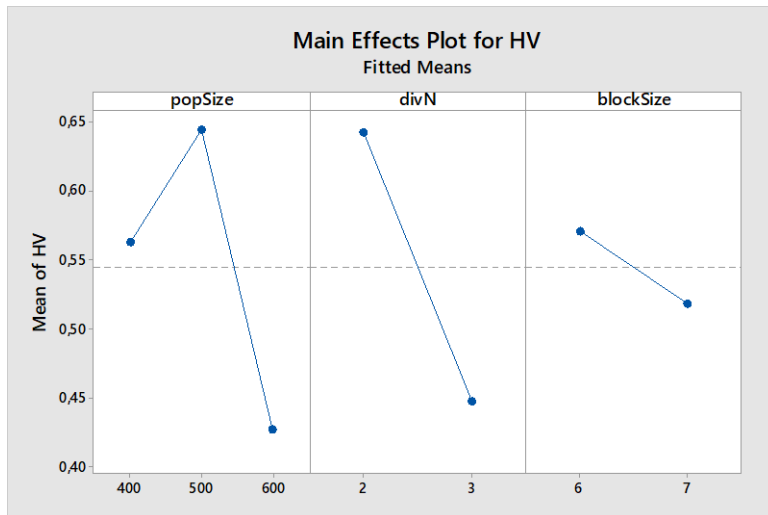


Figure 5.20 HV values of the DoE for the adapted ϵ -MOEA

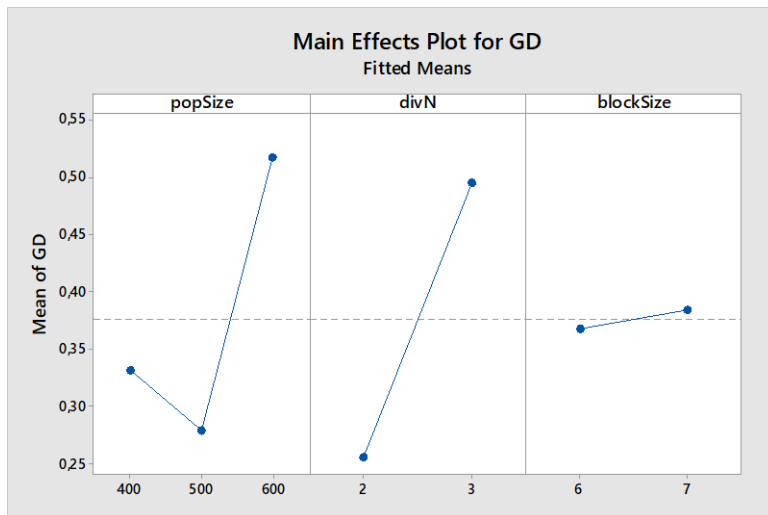


Figure 5.21 GD values of the DoE for the adapted ϵ -MOEA

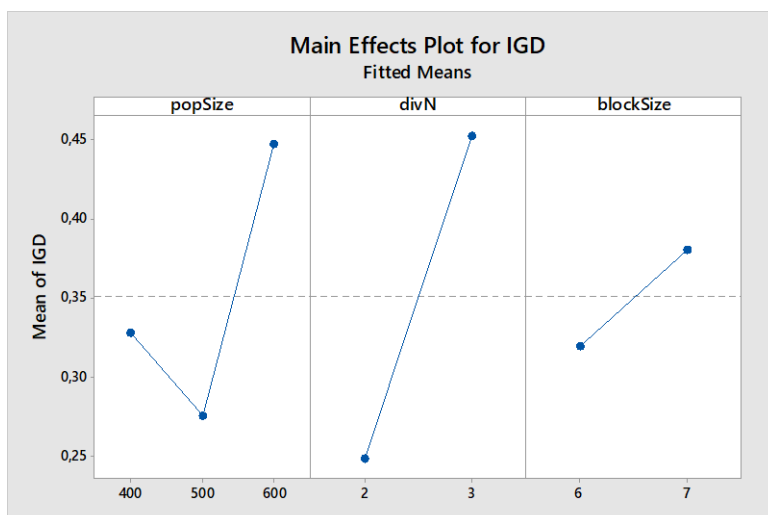


Figure 5.22 IGD values of the DoE for the adapted ϵ -MOEA

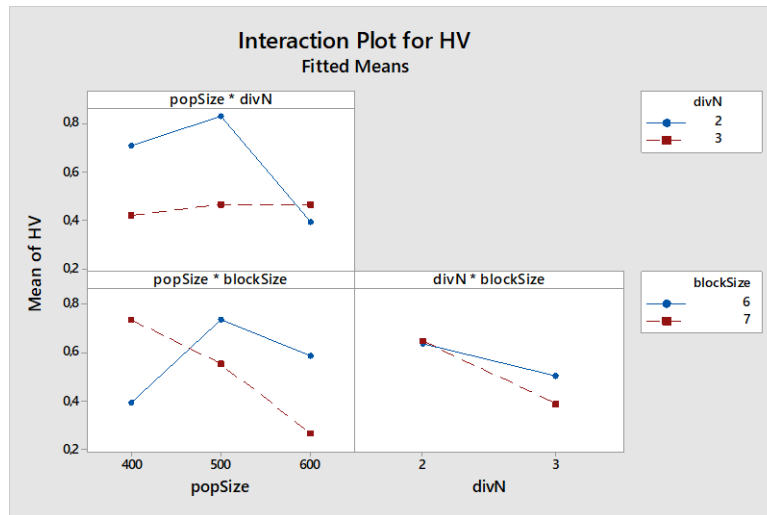


Figure 5.23 Interaction of DoE parameters according to HV for the adapted ϵ -MOEA

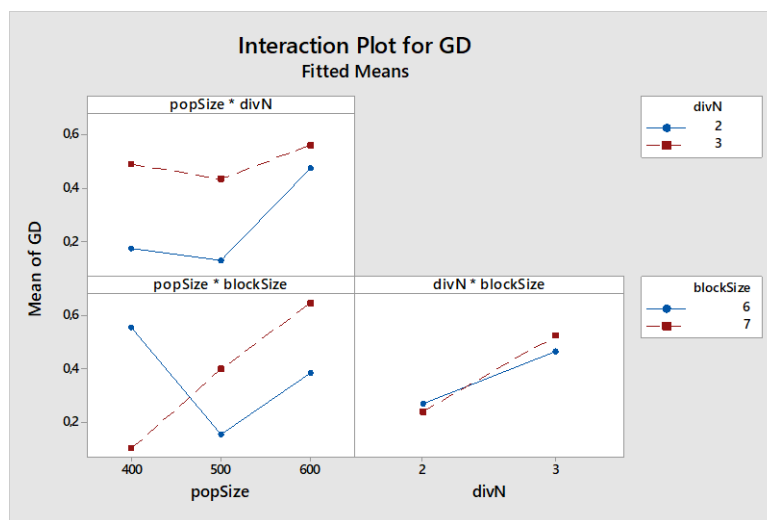


Figure 5.24 Interaction of DoE parameters according to GD for the adapted ϵ -MOEA

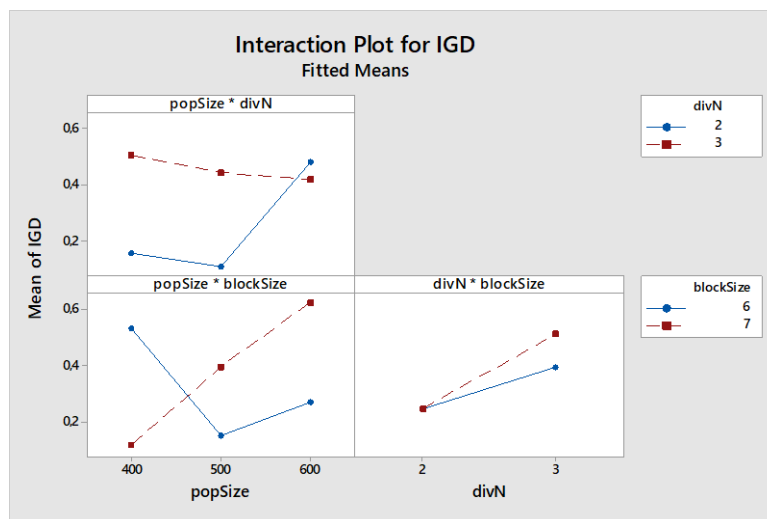


Figure 5.25 Interaction of DoE parameters according to IGD for the adapted ϵ -MOEA

The best combination for the adapted ϵ -MOEA is obtained as popSize = 500, divN = 2 and blockSizeDivisor = 6 when these parameters are evaluated individually. When the two by two interactions of these parameters are taken into consideration, although popSize = 400 and blockSize = 7 seemed to be better than the other options, the other interactions still favor the individual evaluation results of the adapted ϵ -MOEA with HV, GD and IGD. As a result, popSize = 500, divN = 2 and blockSizeDivisor = 6 are used for the in ϵ -MOEA for the final execution of the experimental results.

Both algorithms are run with these parameters respectively on the Christofides and Golden instances. The Pareto front results of them are shown in the figures below and their tables are given in Appendix 2.

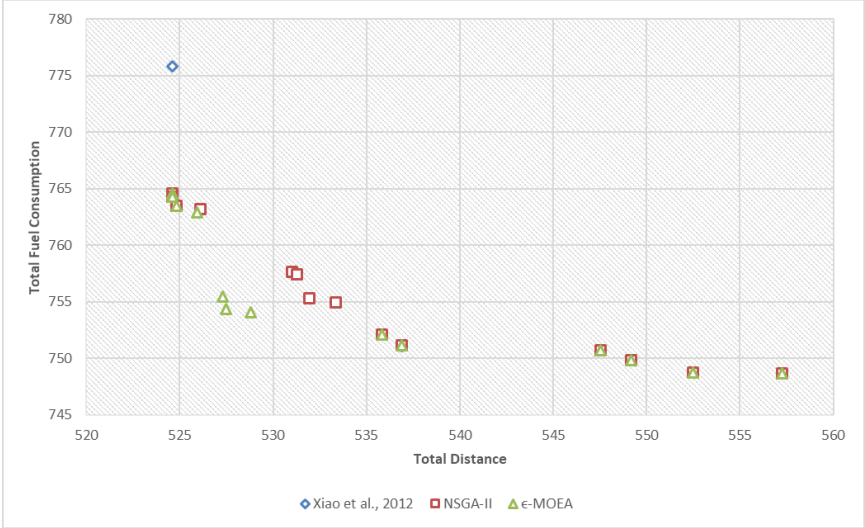


Figure 5.26 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT1

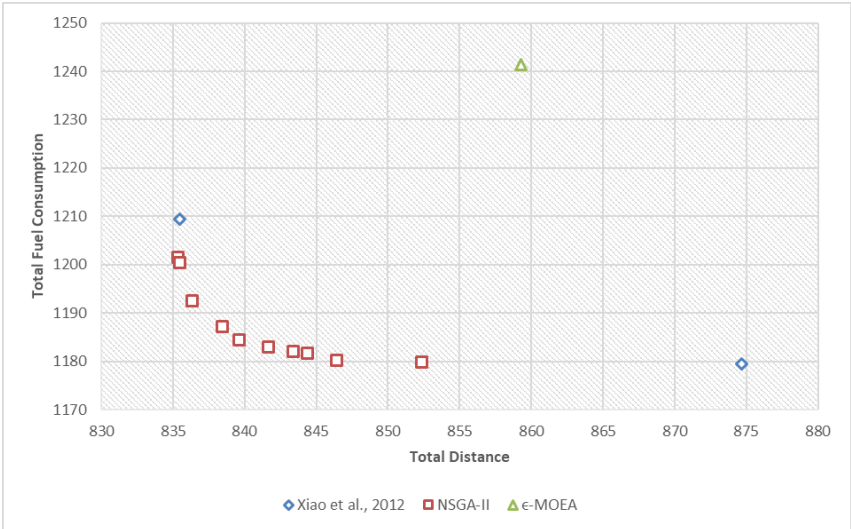


Figure 5.27 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT2

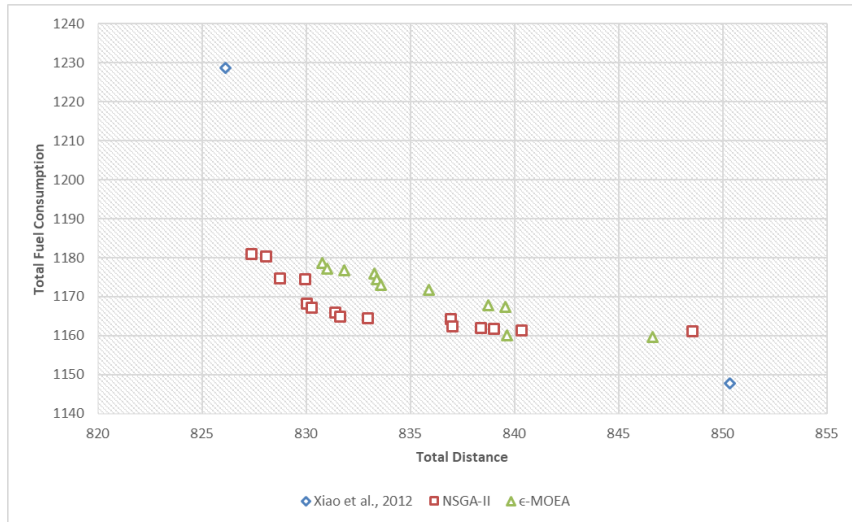


Figure 5.28 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT3

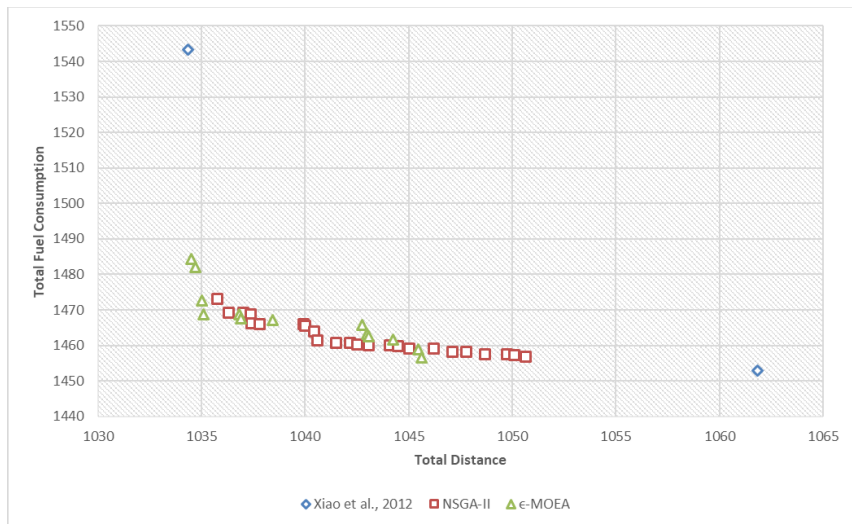


Figure 5.29 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT4

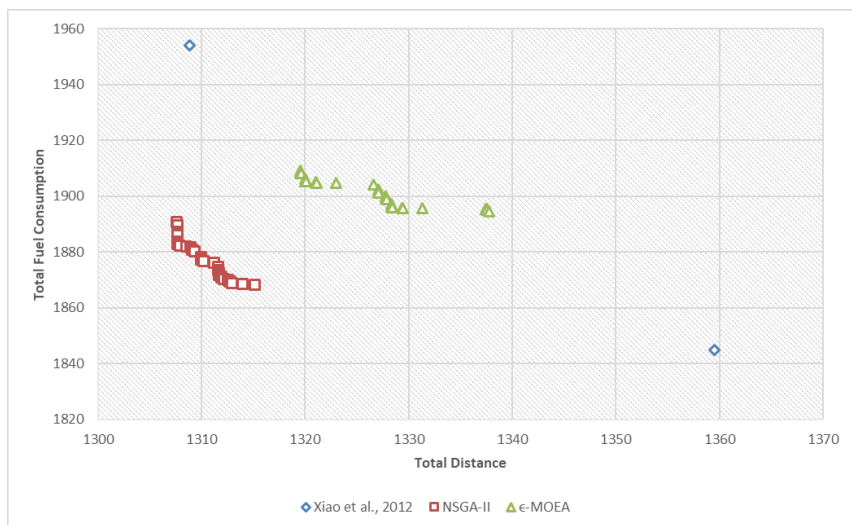


Figure 5.30 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT5

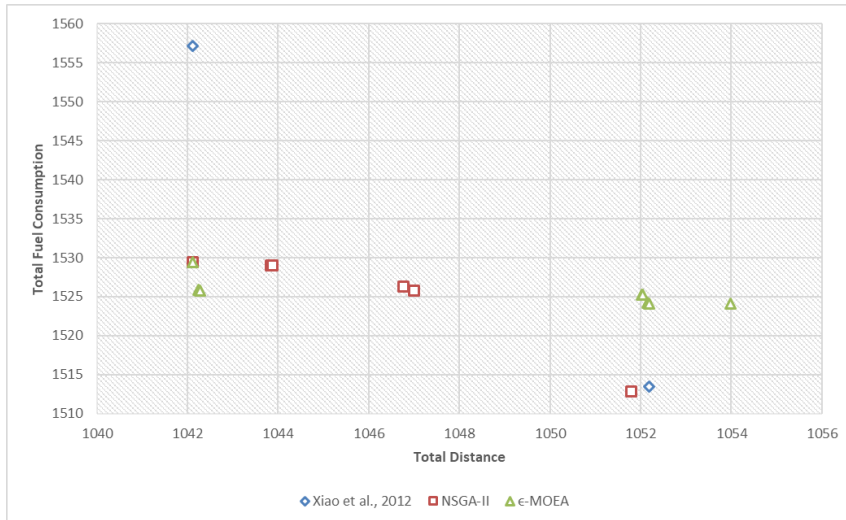


Figure 5.31 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT11

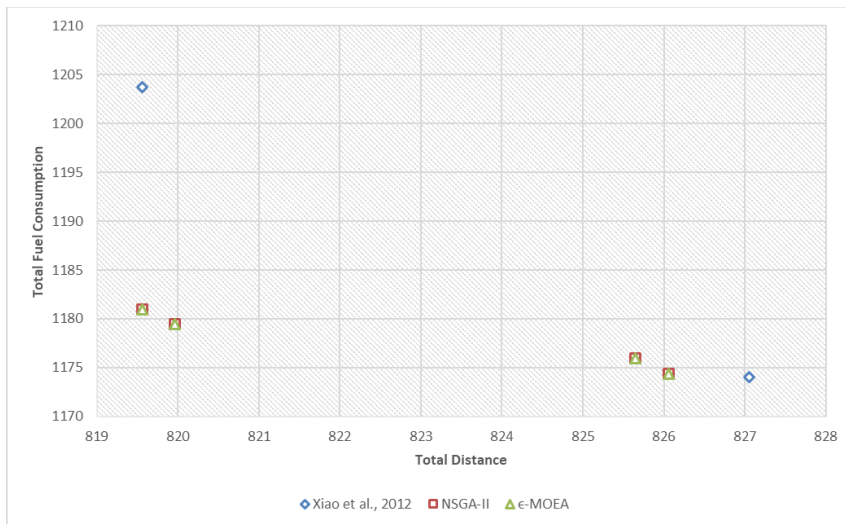


Figure 5.32 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT12

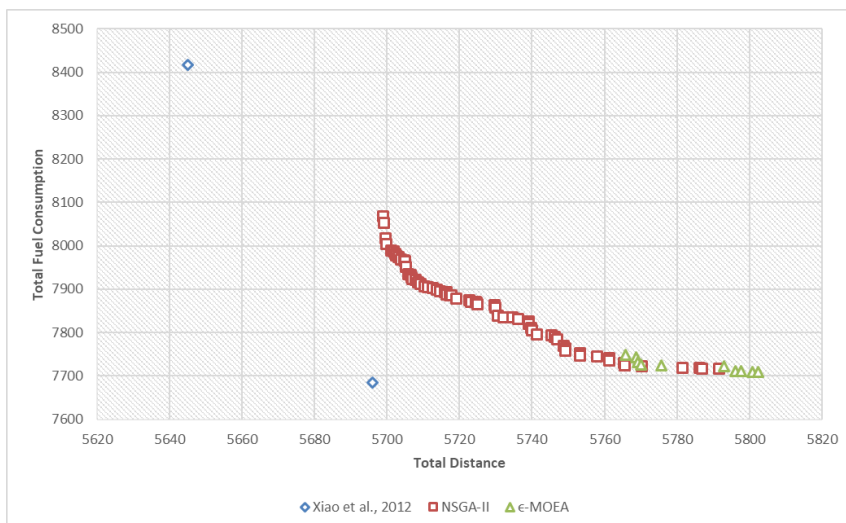


Figure 5.33 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_1

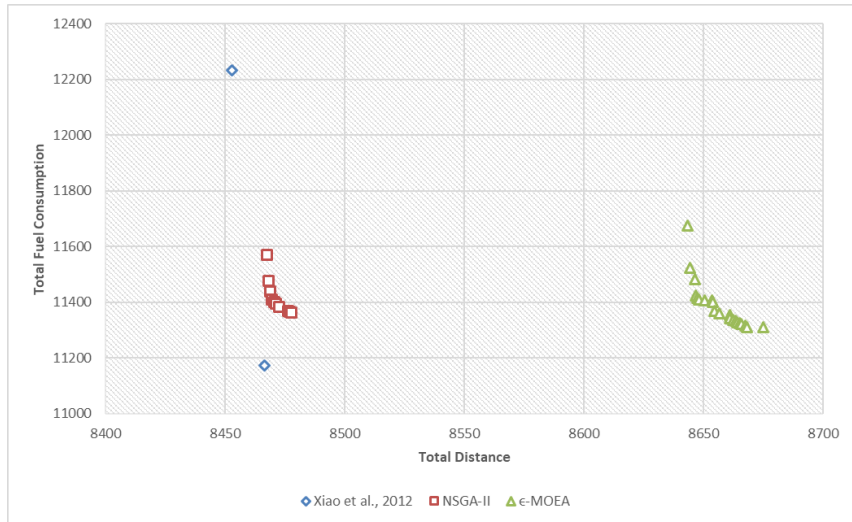


Figure 5.34 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_2

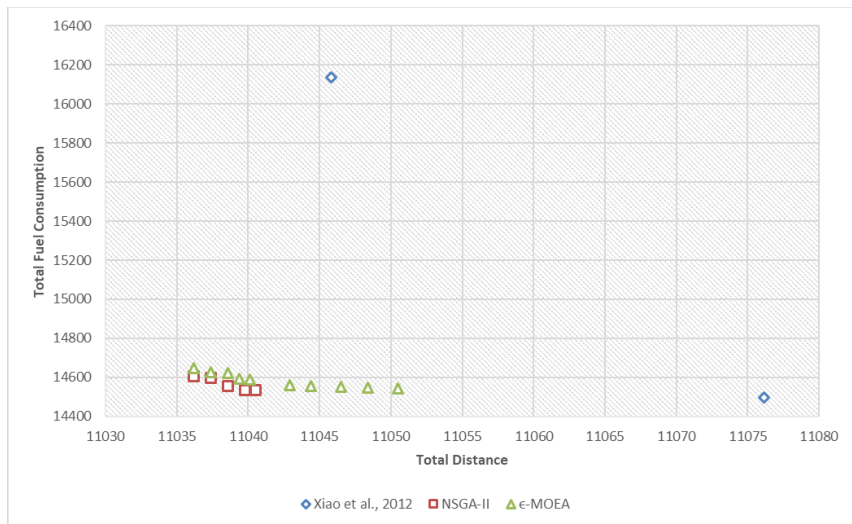


Figure 5.35 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_3

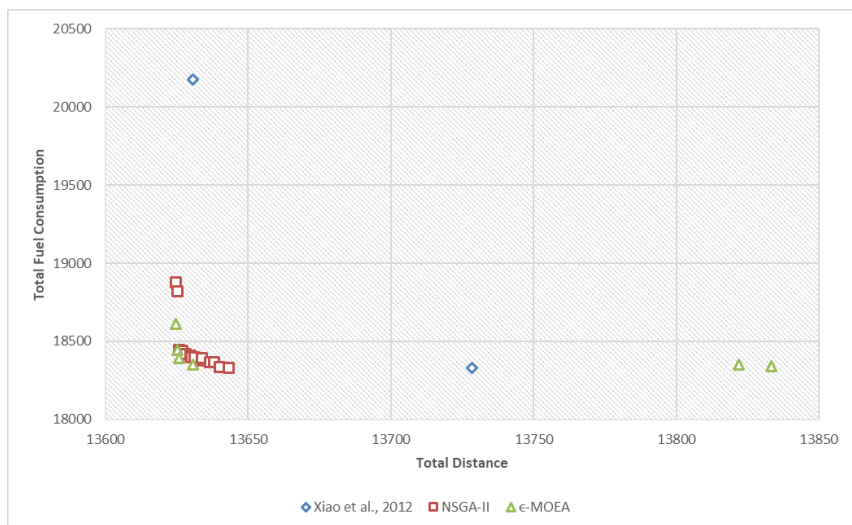


Figure 5.36 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_4

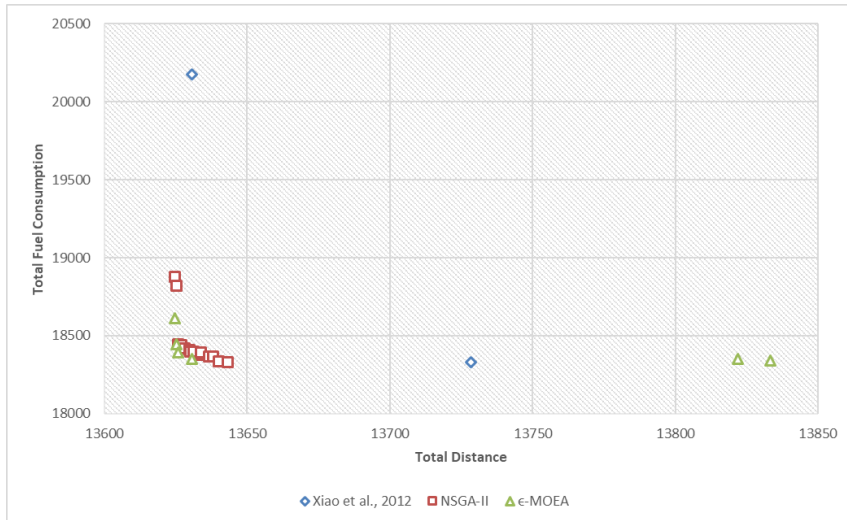


Figure 5.37 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_5

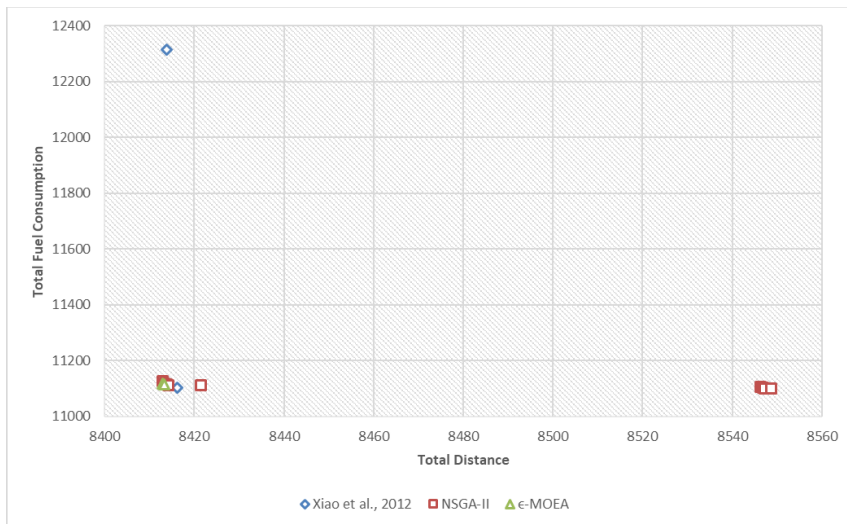


Figure 5.38 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_6

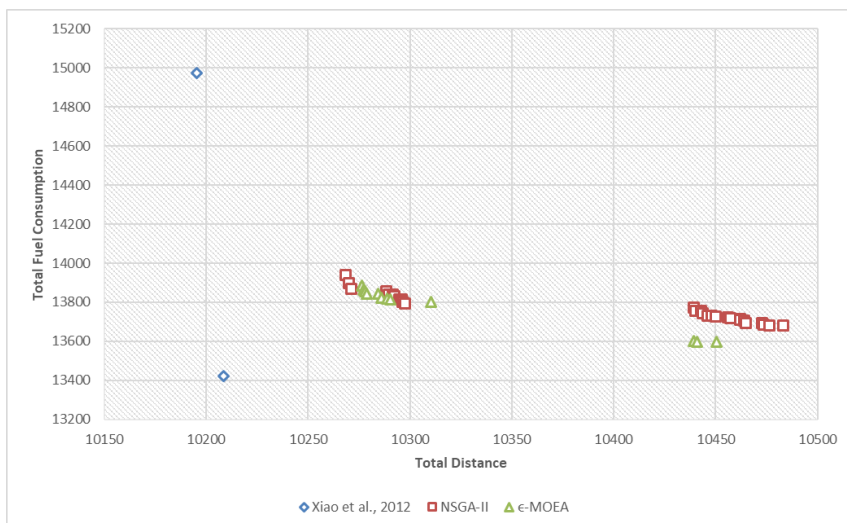


Figure 5.39 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_7

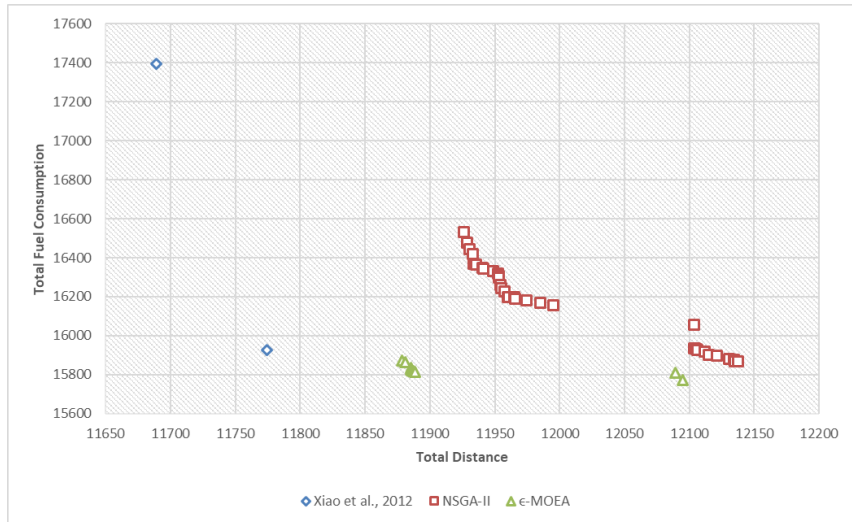


Figure 5.40 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_8

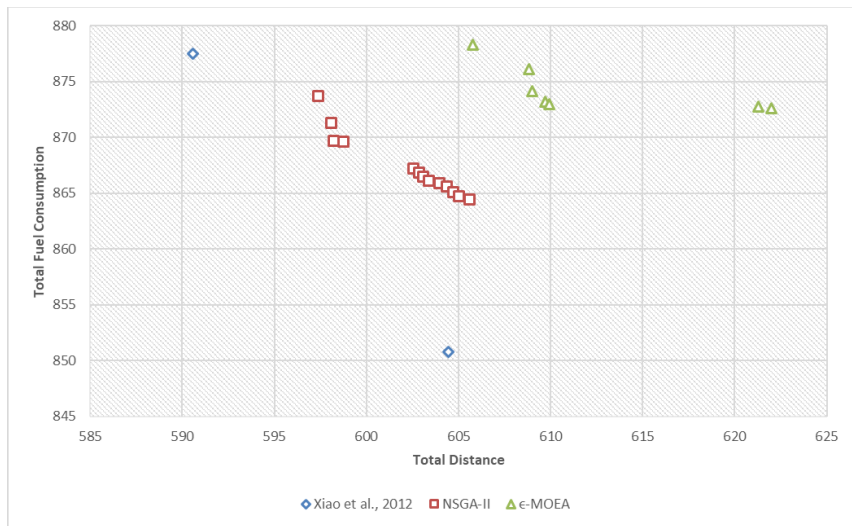


Figure 5.41 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_9

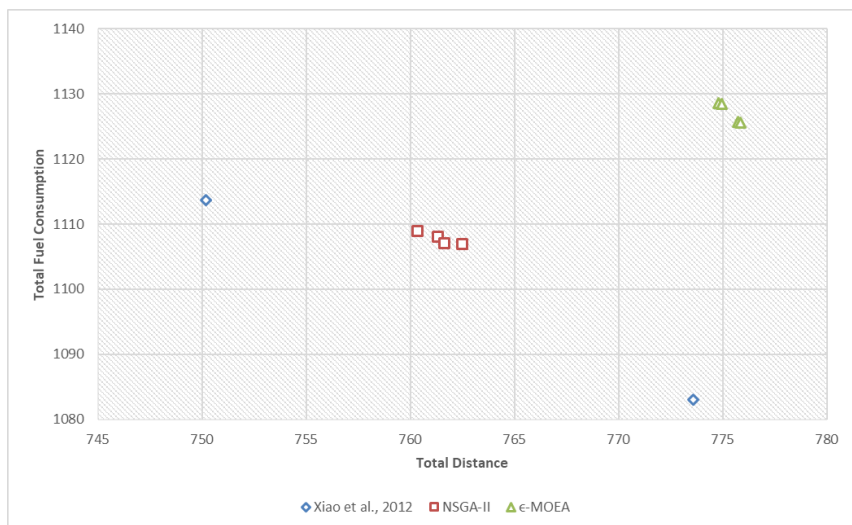


Figure 5.42 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_10

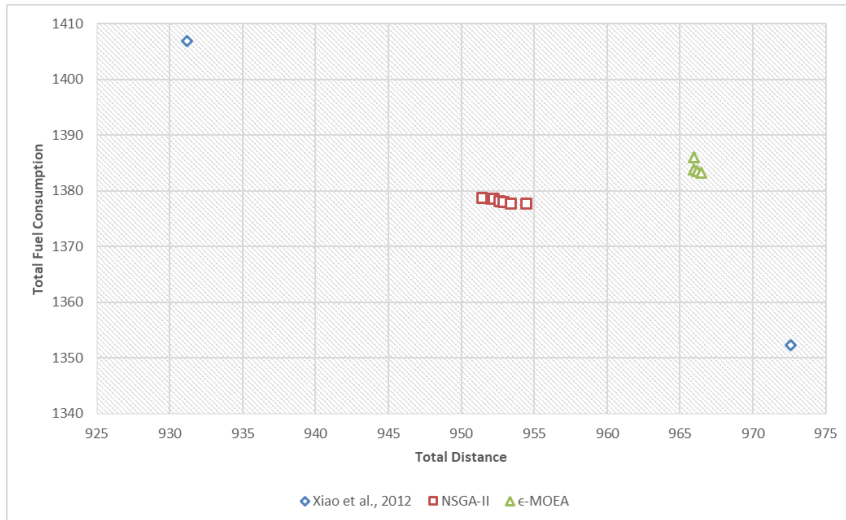


Figure 5.43 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_11

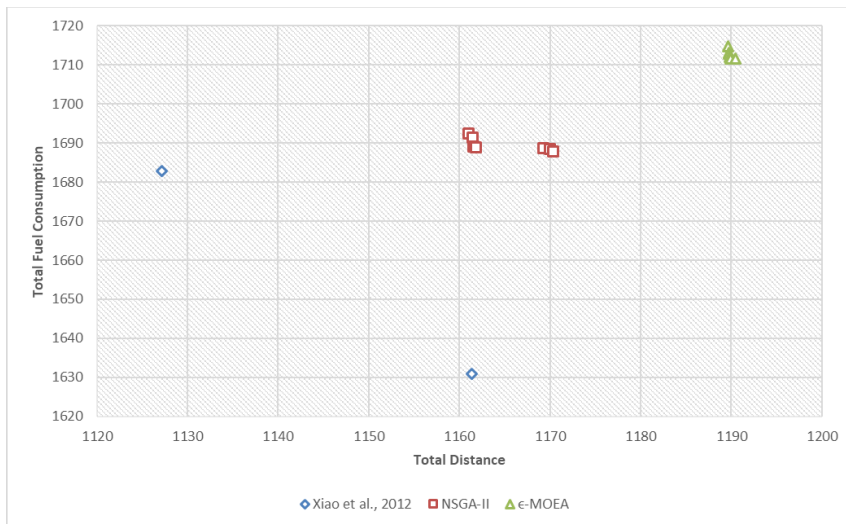


Figure 5.44 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_12

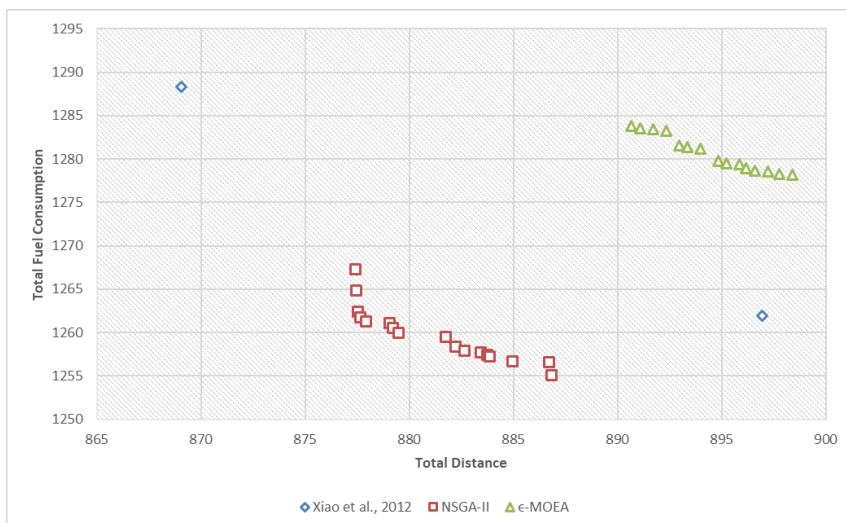


Figure 5.45 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_13

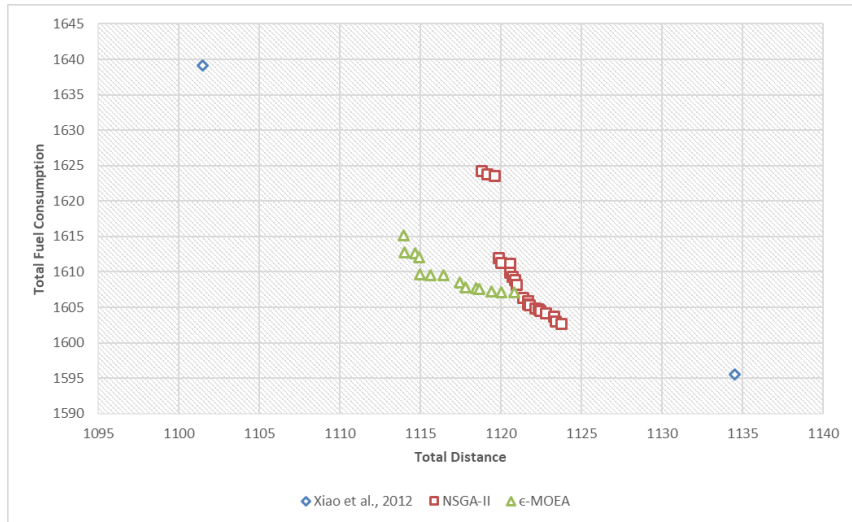


Figure 5.46 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_14

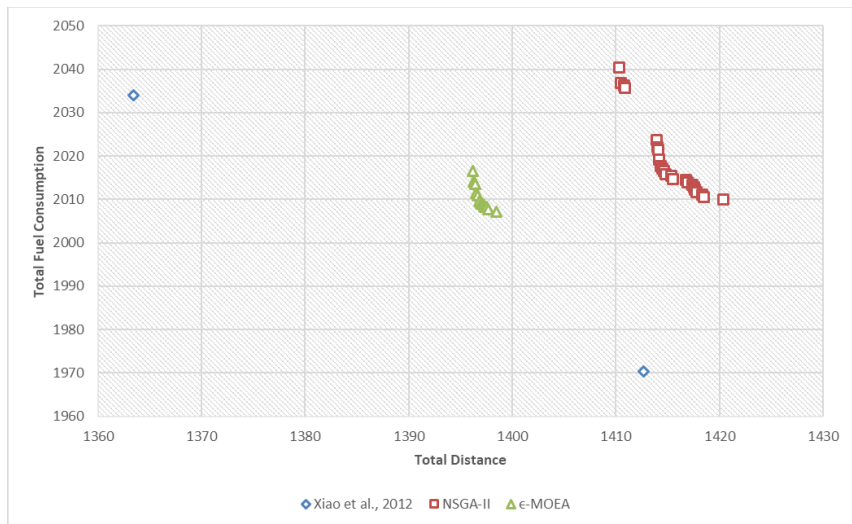


Figure 5.47 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_15

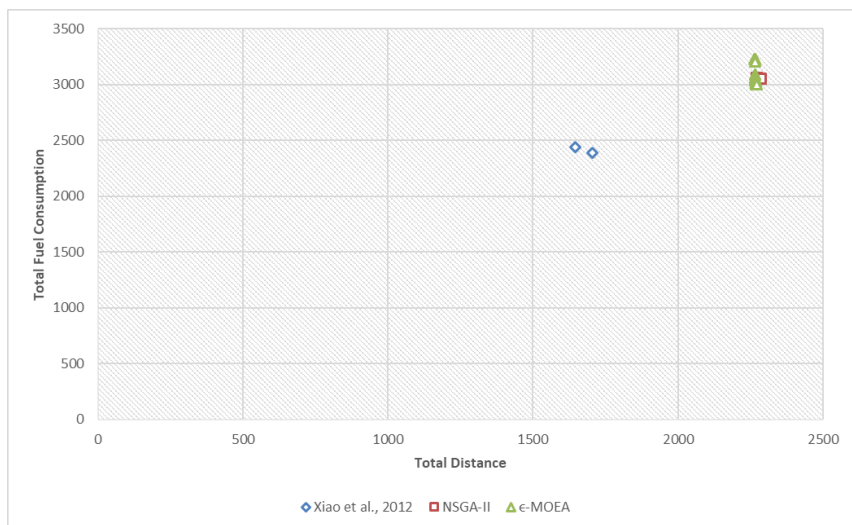


Figure 5.48 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_16

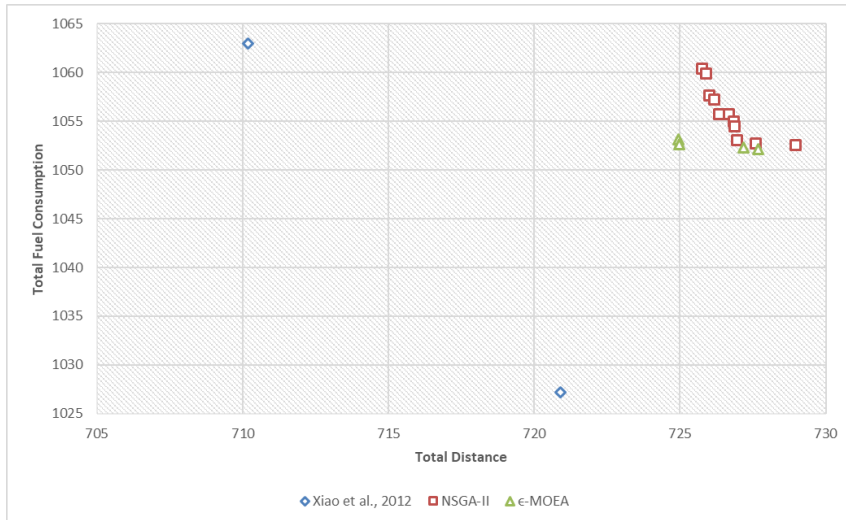


Figure 5.49 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_17

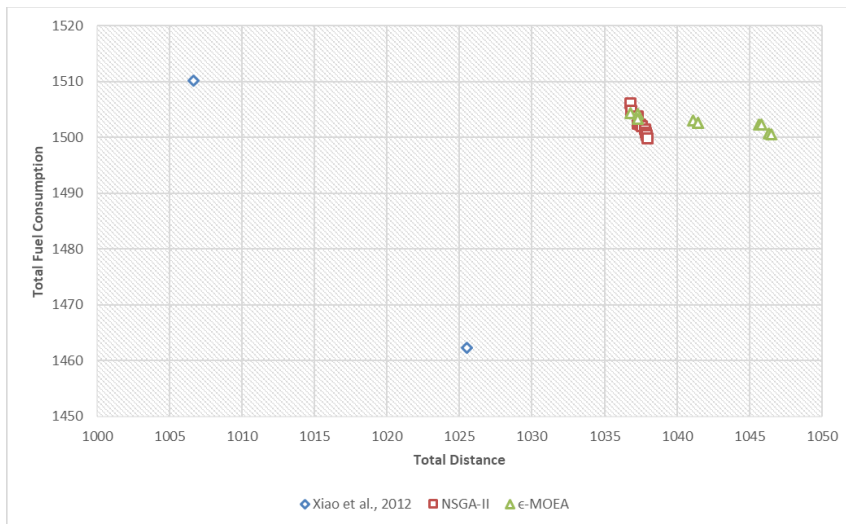


Figure 5.50 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_18

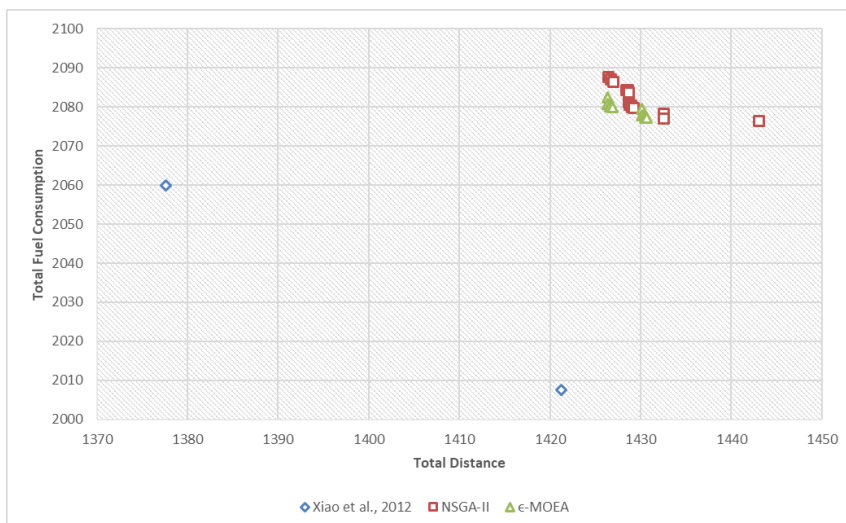


Figure 5.51 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_19

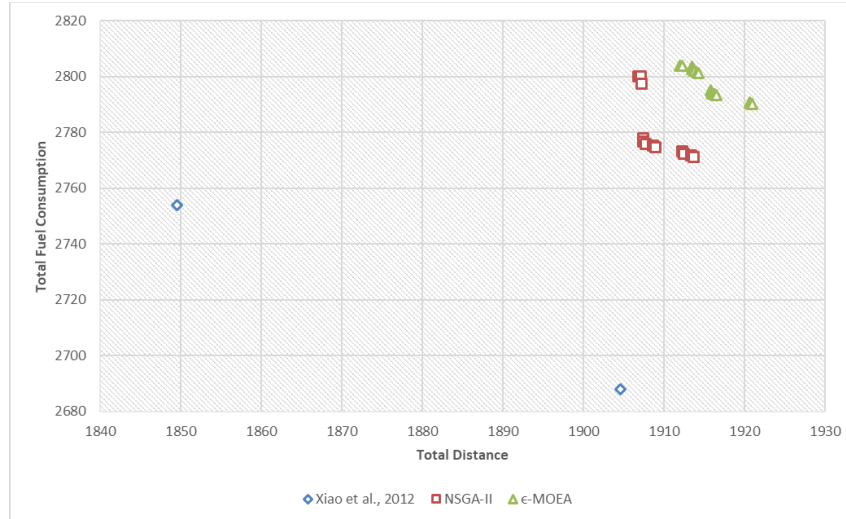


Figure 5.52 Pareto Fronts of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_20

All the adapted NSGA-II results on Christofides instances are non-dominated with the ones obtained by Xiao et al. (2012). Furthermore, the adapted NSGA-II results on four Christofides instances (i.e. CMT2, CMT5, CMT11, and CMT12) dominated the results of Xiao et al. (2012). The adapted NSGA-II results on nine Golden instances are non-dominated with the results of Xiao et al (2012). In five of these instances (i.e. Golden_3, Golden_4, Golden_5, Golden_6, and Golden_13), the adapted NSGA-II results obtained in this thesis dominated the results obtained by Xiao et al. (2012). The adapted NSGA-II results on the other eleven Golden instances (Golden_1, Golden_2, Golden_7, Golden_8, Golden_12, Golden_15, Golden_16, Golden_17, Golden_18, Golden_19, Golden_20) are dominated by the results of Xiao et al. (2012).

The adapted ϵ -MOEA results show that the adapted ϵ -MOEA is not quite successful as the adapted NSGA-II on the same instances. The adapted ϵ -MOEA also found Pareto fronts on six Christofides instances (i.e. CMT1, CMT3, CMT4, CMT5, CMT11 and CMT12). In three of these instances (i.e. CMT1, CMT4 and CMT11), some of the solutions on the adapted ϵ -MOEA Pareto fronts dominated the solutions on the adapted NSGA-II Pareto front. In other words, the adapted ϵ -MOEA improves the results of the adapted NSGA-II results for these instances. In the rest of the Christofides instances, the adapted ϵ -MOEA results are either dominated by the adapted NSGA-II results or these both algorithms construct longer Pareto fronts. Concerning Golden instances, when the adapted ϵ -MOEA Pareto results are compared with the Pareto fronts of the adapted NSGA-II and Xiao et al. (2012), in seven Golden instances (i.e.

Golden_3, Golden_4, Golden_5, Golden_6, Golden_8, Golden_14, Golden_15) , the adapted ϵ -MOEA found Pareto fronts. In five of these instances (i.e. Golden_3, Golden_4, Golden_8, Golden_14, Golden_15), the adapted ϵ -MOEA results either fully or partially dominates the adapted NSGA-II results. Especially for Golden_8 and Golden_15 instances, the adapted ϵ -MOEA found a Pareto front where the adapted NSGA-II is dominated by the Pareto front of Xiao et al. (2012). In the rest of the Golden instances, the adapted ϵ -MOEA results are dominated by the adapted NSGA-II results.

When the adapted NSGA-II and the adapted ϵ -MOEA results are evaluated together, Pareto fronts are found for all the Christofides instances and eleven Golden instances (i.e. Golden_3, Golden_4, Golden_5, Golden_6, Golden_8, Golden_9, Golden_10, Golden_11, Golden_13, Golden_14 and Golden_15). As a result, in 18 out of 27 total instances, Pareto fronts are obtained. The ratio of success in finding Pareto fronts is roughly 67%.

In order to compare the success of each algorithm (i.e. the adapted NSGA-II and the adapted ϵ -MOEA) according to one another, ultimate Pareto fronts are obtained with the combination of two Pareto fronts found by two algorithms for all the instances. Then each algorithm's HV, GD and IGD performance metrics are calculated with regards to the ultimate Paretos for each instance. Since there are differences in value ranges in both objectives, the values of the solutions on the Pareto fronts are normalized before these performance measures are calculated. For HV, the greater values show the better performance of the algorithms. For GD and IGD, on the other hand, the smaller values represent the better performance of the algorithms. The results of these performance measures are shown in Table 5-1 below.

Table 5-1 Performance Measures of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour on all Christofides and Golden Instances

	HV		GD		IGD	
	NSGA-II	ϵ -MOEA	NSGA-II	ϵ -MOEA	NSGA-II	ϵ -MOEA
CMT1	0.683497	0.762114	0.046143	0	0.033476	0
CMT2	0.946663	0	0	1.04107	0	1.18082
CMT3	0.760264	0.563496	0.012077	0.147646	0.012077	0.185655
CMT4	0.741566	0.739884	0.062816	0.03684	0.053004	0.054819
CMT5	0.960098	0.138975	0.003517	0.718941	0	0.752425
CMT11	0.286957	0.233049	0.056284	0.292471	0.087126	0.215366
CMT12	0.252041	0.252041	0	0	0	0
Golden_1	0.749671	0.337396	0	0.021217	0.004334	0.474205
Golden_2	0.84724	0.124374	0	0.121653	0.474559	0.403982
Golden_3	0.882599	0.593339	0	0.350762	0	0.256122
Golden_4	0.985454	0.953694	0.083649	0.294451	0.062127	0.021489
Golden_5	0.100637	0	0	0	0	1.04902
Golden_6	0.617323	0.4653	0.04969	0	0.011959	0.576547
Golden_7	0.444354	0.49715	0.222956	0.009479	0.065334	0.023388
Golden_8	0.408271	0.9528	0.482998	0	0.474355	0
Golden_9	0.899187	0.202718	0	0.595763	0	0.551235
Golden_10	0.993208	0.00146	0	1.24667	0	1.24271
Golden_11	0.988256	0.009293	0	1.10412	0	1.09754
Golden_12	0.986207	0.002894	0	1.12743	0	1.18577
Golden_13	0.938819	0.037269	0	0.899962	0	0.946233
Golden_14	0.321121	0.728837	0.141339	0	0.198177	0.088666
Golden_15	0.234111	0.992845	0.792566	0	0.752829	0
Golden_16	0.534843	0.928557	0.359287	0	0.355676	0
Golden_17	0.581536	0.96536	0.454765	0	0.286691	0
Golden_18	0.933153	0.513913	0.045212	0.480939	0.008793	0.274546
Golden_19	0.775472	0.873302	0.16336	0	0.091762	0.070519
Golden_20	0.919272	0.130127	0	0.573554	0	0.654776
Average	0.695253	0.444451	0.110247	0.335665	0.110084	0.418735

5.3. Visualization of the Results

A graphical user interface (GUI) is developed in Java programming language in order to visualize the solutions obtained in the experimental studies. The red dots represent the customers in the problem instance (Figure 5.53). They are located on the canvas according to their Euclidean coordinates. The size of the red dots is determined proportionally to their demands. The greater dots represent the customers that have bigger demands, while smaller dots represent customer with smaller demands. The obtained solutions are also shown on the GUI (Figure 5.54). Each vehicle tour is drawn

with a different color and each of them can be seen individually (Figure 5.55). The following figures visualize the adapted NSGA-II solution for Golden_3 instance.

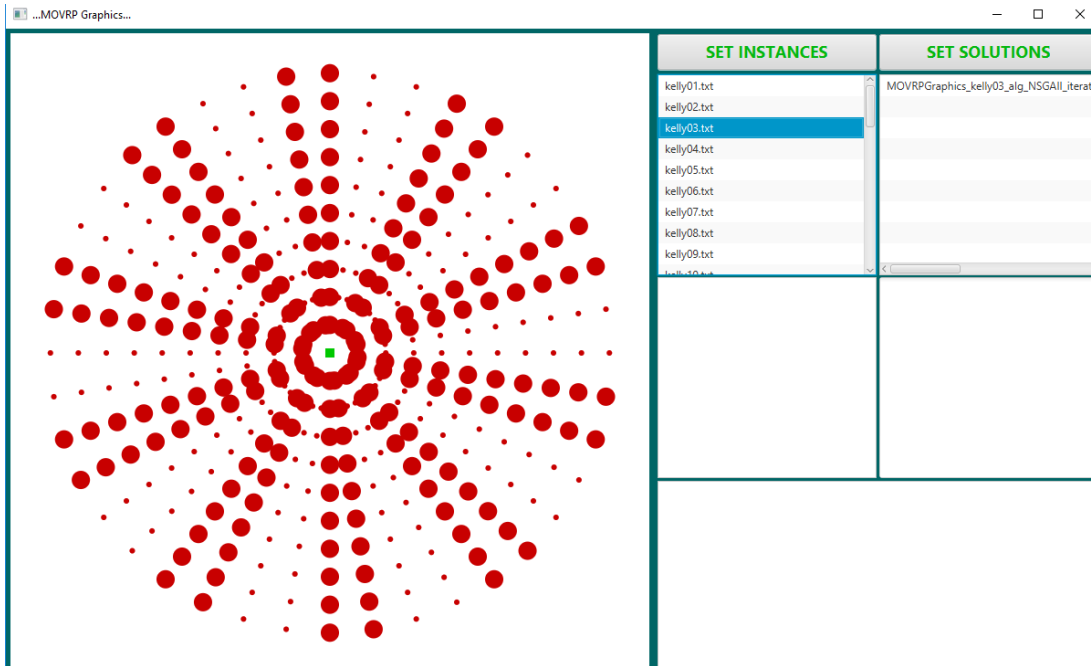


Figure 5.53 MOGVRP GUI with JAVA – Instances

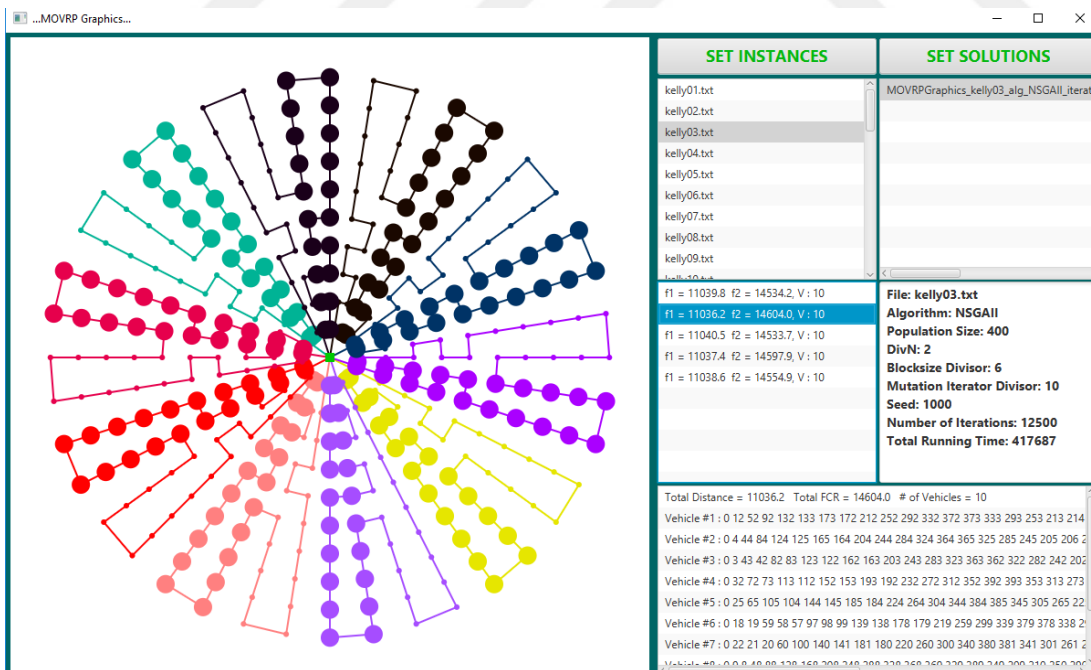


Figure 5.54 MOGVRP GUI with JAVA – Solutions

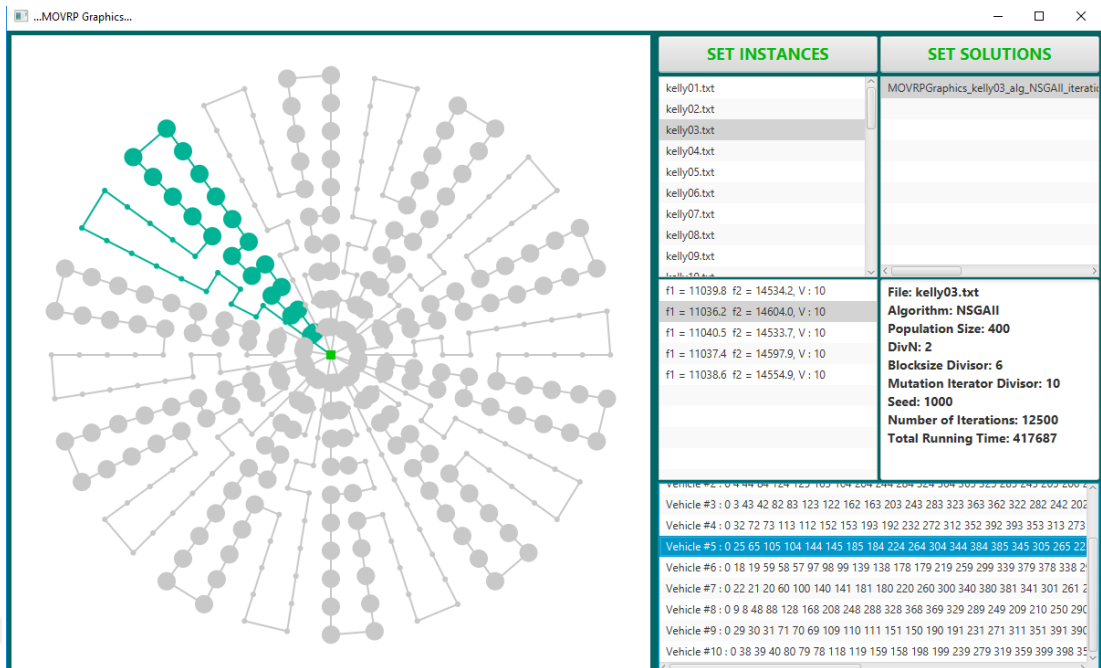


Figure 5.55 MOGVRP GUI with JAVA – Individual Vehicle Tours

CONCLUSIONS AND FUTURE RESEARCH

In this thesis, a MOGVRP was studied and two state-of-the-art MOEAs (i.e. NSGA-II and ϵ -MOEA) were applied to this problem. Minimizing the total distance cost and minimizing the total fuel consumption objectives were selected as the two objective functions for the problem. Since there are more than one fuel consumption formula in literature, the one that did not require external information other than the problem instances provide and that is referenced more than the others was selected as the second objective for the problem. The fuel consumption rate formulated by Xiao et al. (2012) was taken into consideration in this thesis. The advantage of this fuel consumption formulation is that it only requires the distance information between the customers and the demands of these customers. In other words, it could be applied on any VRP instance in literature.

The contributions of this thesis study can be summarized as follows. First, it provides Pareto fronts to the well-known VRP instances (Christofides and Golden instances used in this thesis) by using the objective functions: minimizing the total distance and minimizing the total fuel consumption. For the latter, the formula proposed by Xiao et al. (2012) is used. In 18 out of 27 the Christofides and Golden instances, the produced Pareto fronts are not dominated by the results of Xiao et al. In that sense, these Pareto fronts are the state-of-the-art results for this MOGRVP on these instances until better Pareto fronts are obtained. To the best of our knowledge, there is no benchmark study done for these two objectives on these instances before.

Another contribution of this thesis is the developed version of PR. PR heuristic is used in a multi-objective structure and used in order to function as a crossover operator rather than a local search method. In the classical PR process, the best solution along the path between the initial solution and the guiding solution is searched. PR used in this thesis, on the other hand, produces a set of non-dominated solutions from the two parent solutions selected from the population and/or archive. Then, the mutation and local search heuristics are applied to these solutions. This adapted PR assisted the adopted NSGA-II and ϵ -MOEA with split vehicle tour solution representation in the reproduction phase by generating non-dominant offspring. In that sense, it functioned like a crossover operator generating multiple offspring that fit into MOOP structure.

Another contribution of this thesis is the proposed ELV local search heuristic. During the experimental studies, it was observed that eliminating the vehicles with the least customers by moving its customers to other vehicles minimized the total values of the objectives. In this process, ELV heuristic applied within a multi-objective setting played a critical role. When ELV is used, the results of the algorithms are improved. It is thought that ELV can be used as a local search to improve the obtained results in other VRPs, as well.

There have also been some challenges during this thesis study. Thinking, designing, applying and adopting the methods and algorithms in the MOOP setting was not easy. Some of the methods did not work effectively due to their greedy nature and not contributing to the gain for both objectives. Since there are two partially conflicting objective functions, these methods can not favor both objectives at the same time. Or, some of the methods have to favor one objective or another, otherwise cannot be applied within a MOOP context.

Another challenge was caused by solution representations. In the giant tour representation, intervening directly to the vehicles is not possible. The well-known Bellman split algorithm has full control over the giant tour and its vehicle splits. Although giant tour representation provides opportunities to apply classical crossover operators, it still does not allow to make small but effective changes by directly accessing the individual vehicles. In that sense, the giant tour has shortcomings on the exploitation of the algorithms in the search space. Concerning the split vehicle tour representation, although it has given all the opportunities to intervene directly to the individual vehicles, it does not allow to apply crossover operators which could help algorithms to do wider exploration on the search space. In order to fill that gap, some other local search methods such as PR are adapted in a MOOP setting and applied in the algorithms.

The last challenge that was encountered during this thesis study is related to the characteristic of Golden instances. As it is seen in the results, both algorithms produced promising Pareto front results on Christofides instances. It is because the distribution of the customers and depot in those instances is random. However, in Golden instances, both the locations of the customers and the depot, and the demands of the customers are laid out in a pattern. In all the Golden instances, the neighbors of the customers are located with an even distance. This fact provides multiple options when an algorithm

is to decide on including the next customer in a vehicle. This situation challenges algorithms in making the right selections. In some of the Golden instances, the customers with the heaviest loads are located right around the depot. This situation leads to unbalanced vehicle routes. This is another challenge that the algorithms need to solve. This orderly pattern concerning the locations and the demands of the customers challenged the algorithms used in this thesis in finding the aimed Pareto fronts for Golden instances (especially Golden instance set 5). In order to find the Pareto optimal solutions for these instances, algorithms need to know or get trained according to this pattern. It could be done with setting the parameters in the algorithms specifically favoring those instances by a DoE. Then, however, this DoE setting would be very much problem specific. The same parameters may not be as successful as in other instances. And this type of convergence would be against the nature EAs in general. Therefore, in this thesis, it was avoided such an approach and all the instances were taken into consideration in setting the parameters of the algorithms.

Despite all these challenges, Pareto fronts are found in 67% of all the instances used in the experimental studies. As future work, this percentage can be increased by either improving the current local search moves used in the thesis, or adopting other local search moves, or even implementing other well-known MOEAs in literature.

Another future study would be studying on another variant of MOGVPRP such as VRPs include only electrical cars or vehicles that carry hazardous material, etc. During this thesis study, it was observed that MOGVPRP has a different number of fields that can be studied and can be easily applied to real life problems.

REFERENCES

- Araujo, S. A. De, Poldi, K. C., & Smith, J. (2014). A genetic algorithm for the one-dimensional cutting stock problem with setups. *Pesquisa Operacional*, *34*, 165–187. <https://doi.org/10.1590/0101-7438.2014.034.02.0165>
- Ardenkani, S., Hauer, E., & Jamei, B. (2001). Traffic impact models. *Traffic Flow Theory: A State-of-the Art Report*, *7*, 7–24.
- Bektaş, T., & Laporte, G. (2011). The Pollution-Routing Problem. *Transportation Research Part B: Methodological*, *45*, 1232–1250. <https://doi.org/10.1016/j.trb.2011.02.004>
- Boulter, P. G., McCrae, I. S., & Barlow, T. J. (2007). A review of instantaneous emission models for road vehicles.
- Carić, T., & Gold, H. (2008). *Vehnicle_Routing_Problem*. (T. Carić & H. Gold, Eds.). In-Teh. <https://doi.org/10.5772/62148>
- Christofides, N., Mingozzi, A., & Toth, P. (1979). The Vehicle Routing Problem. *Revue Française d'automatique, Informatique, Recherche Opérationnelle. Recherche Opérationnelle*, *10*. <https://doi.org/10.1051/ro/197610V100551>
- Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operation Research*, *12*(4), 568–581. <https://doi.org/10.1287/opre.12.4.568>
- Coello, C. A. C., & Cortés, N. C. (2005). Solving Multiobjective Optimization Problems Using an Artificial Immune System. *Genetic Programming and Evolvable Machines*, *6*(2), 163–190. <https://doi.org/10.1007/s10710-005-6164-x>
- Davis, L. (1985). Job Shop Scheduling with Genetic Algorithms. In *Proceedings of the 1st International Conference on Genetic Algorithms* (pp. 136–140). Hillsdale, NJ, USA: L. Erlbaum Associates Inc. Retrieved from <https://dl.acm.org/citation.cfm?id=645511.657084>
- Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: John Wiley & Sons, Inc.
- Deb, K., Mohan, M., & Mishra, S. (2003). A Fast Multi-objective Evolutionary Algorithm for Finding Well-Spread Pareto-Optimal Solutions. *KanGAL Report*, *2003002*.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182–197. <https://doi.org/10.1109/4235.996017>
- Demir, E., Bektaş, T., & Laporte, G. (2011). A comparative analysis of several vehicle emission models for road freight transportation. *Transportation Research. Part D: Transport and Environment*, *16*(5), 347–357. <https://doi.org/10.1016/j.trd.2011.01.011>
- Demir, E., Bektaş, T., & Laporte, G. (2012). An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, *223*(2), 346–359. Retrieved from <https://eprints.soton.ac.uk/340723/>

- Demir, E., Bektaş, T., & Laporte, G. (2013). The bi-objective pollution-routing problem. *European Journal of Operational Research*, 232(3), 464–478. <https://doi.org/10.1016/j.ejor.2013.08.002>
- Dükkancı, O., Kara, B. Y., & Bektaş, T. (2019). The green location-routing problem. *Computers and Operations Research*. <https://doi.org/10.1016/j.cor.2019.01.011>
- EEA. (2018). *Monitoring CO2 emissions from new passenger cars and vans in 2017*.
- Eglese, R., & Bektaş, T. (2014). Chapter 15: Green Vehicle Routing. In *Vehicle Routing* (pp. 437–458). <https://doi.org/10.1137/1.9781611973594.ch15>
- EPA. (2018). *Fast Facts: U.S. Transportation Sector Greenhouse Gas Emissions 1990-2016*.
- Figliozzi, M. (2010). Vehicle Routing Problem for Emissions Minimization. *Transportation Research Record*, 2197(1), 1–7. <https://doi.org/10.3141/2197-01>
- Friedrich, T., Horoba, C., & Neumann, F. (2009). Multiplicative Approximations and the Hypervolume Indicator. In *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation* (pp. 571–578). New York, NY, USA: ACM. <https://doi.org/10.1145/1569901.1569981>
- Glover, F., Laguna, M., & Marti, R. (2000). Fundamentals of scatter search and path relinking. *Control and Cybernetics*, Vol. 29, n, 653–684.
- Goldberg, D. E., & Lingle, R. J. (1985). Alleles, Loci and the Traveling Salesman Problem. In *Proceedings of the 1st International Conference on Genetic Algorithms* (pp. 154–159). Hillsdale, NJ, USA: L. Erlbaum Associates Inc. Retrieved from <http://dl.acm.org/citation.cfm?id=645511.657095>
- Golden, B. L., Wasil, E. A., Kelly, J. P., & Chao, I.-M. (1998). The Impact of Metaheuristics on Solving the Vehicle Routing Problem: Algorithms, Problem Sets, and Computational Results. In T. G. Crainic & G. Laporte (Eds.), *Fleet Management and Logistics* (pp. 33–56). Boston, MA: Springer US. https://doi.org/10.1007/978-1-4615-5755-5_2
- Hof, J., & Schneider, M. (2019). An adaptive large neighborhood search with path relinking for a class of vehicle-routing problems with simultaneous pickup and delivery. *Networks*, 0(0). <https://doi.org/10.1002/net.21879>
- Ishibuchi, H., Masuda, H., & Nojima, Y. (2016). Sensitivity of performance evaluation results by inverted generational distance to reference points. *2016 IEEE Congress on Evolutionary Computation (CEC)*, 1107–1114.
- Ishibuchi, H., & Murata, T. (1998). A multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 28(3), 392–403. <https://doi.org/10.1109/5326.704576>
- Jabali, O., Woensel, T. van, & Kok, A. G. de. (2012). Analysis of travel times and CO2 emissions in time-dependent vehicle routing. *Production and Operations Management*, 21(6), 1060–1074. <https://doi.org/10.1111/j.1937-5956.2012.01338.x>
- Koç, Ç., & Karaoğlan, İ. (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing Journal*.

<https://doi.org/10.1016/j.asoc.2015.10.064>

- Kollat, J. B., & Reed, P. M. (2006). Comparing state-of-the-art evolutionary multi-objective algorithms for long-term groundwater monitoring design. *Advances in Water Resources*, 29(6), 792–807.
<https://doi.org/https://doi.org/10.1016/j.advwatres.2005.07.010>
- Kuo, Y. (2010). Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. *Computers & Industrial Engineering*, 59, 157–165. <https://doi.org/10.1016/j.cie.2010.03.012>
- Li, F., Golden, B., & Wasil, E. (2005). Very large-scale vehicle routing: new test problems, algorithms, and results. *Computers & Operations Research*, 32(5), 1165–1179. <https://doi.org/https://doi.org/10.1016/j.cor.2003.10.002>
- Macrina, G., Laporte, G., Guerriero, F., & Pugliese, L. D. P. (2019). An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows. *European Journal of Operational Research*.
<https://doi.org/10.1016/j.ejor.2019.01.067>
- Oliver, I. M., Smith, D. J., & Holland, J. R. C. (1987). A Study of Permutation Crossover Operators on the Traveling Salesman Problem. In *Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Application* (pp. 224–230). Hillsdale, NJ, USA: L. Erlbaum Associates Inc. Retrieved from <http://dl.acm.org/citation.cfm?id=42512.42542>
- Ow, P. S., & Morton, T. E. (1989). The Single Machine Early/Tardy Problem. *Management Science*, 35(2), 177–191. <https://doi.org/10.1287/mnsc.35.2.177>
- Pastorello, C. (2019). *Monitoring CO2 emissions from new passenger cars and vans in 2017*.
- Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, 31(12), 1985–2002.
[https://doi.org/https://doi.org/10.1016/S0305-0548\(03\)00158-8](https://doi.org/https://doi.org/10.1016/S0305-0548(03)00158-8)
- Prins, C., Labadi, N., & Reghioui, M. (2009). Tour splitting algorithms for vehicle routing problems. *International Journal of Production Research*, 47(2), 507–535. <https://doi.org/10.1080/00207540802426599>
- Riquelme, N., Lüken, C. Von, & Baran, B. (2015). Performance metrics in multi-objective optimization. *2015 Latin American Computing Conference (CLEI)*, 1, 1–11. <https://doi.org/10.1109/CLEI.2015.7360024>
- Sheng, L. K., Ibrahim, Z., Buyamin, S., Ahmad, A., Mohd Tumari, M. Z., Mat Jusof, M. F., & Aziz, N. (2014). Multi-Objective particle swarm optimization algorithms – A leader selection overview, 15, 6–19.
<https://doi.org/10.5013/IJSSST.a.15.04.02>
- Sivanandam, S. N., & Deepa, S. N. (2010). *Introduction to Genetic Algorithms* (1st ed.). Springer Publishing Company, Incorporated.
- Srinivas, N., & Deb, K. (1994). Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2(3), 221–248.
<https://doi.org/10.1162/evco.1994.2.3.221>
- Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J.-Y. (1997). A Tabu

- Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science*, 31(2), 170–186. <https://doi.org/10.1287/trsc.31.2.170>
- Tian, Y., Zhang, X., Cheng, R., & Jin, Y. (2016). A multi-objective evolutionary algorithm based on an enhanced inverted generational distance metric. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 5222–5229). <https://doi.org/10.1109/CEC.2016.7748352>
- Toth, P., & Vigo, D. (2014). *Vehicle Routing: Problems, Methods, and Applications, Second Edition*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.
- TUIK. (2012). *National Greenhouse Gas Inventory Report*.
- Van Veldhuizen, D. A., & Lamont, G. B. (2000). On measuring multiobjective evolutionary algorithm performance. *Proceedings of the Congress on Evolutionary Computation, 2000.*, 1, 204–211 vol.1. <https://doi.org/10.1109/CEC.2000.870296>
- Xavier, I., Uchoa, E., Pecin, D., Pessoa, A., Poggi, M., Subramanian, A., ... Oliveira, D. (1998). CVRPLIB: Capacitated Vehicle Routing Problem Library. Retrieved May 11, 2019, from <http://vrp.atd-lab.inf.puc-rio.br/index.php/en/>
- Xiao, Y., Zhao, Q., Kaku, I., & Xu, Y. (2012). Development of a Fuel Consumption Optimization Model for the Capacitated Vehicle Routing Problem. *Comput. Oper. Res.*, 39(7), 1419–1431. <https://doi.org/10.1016/j.cor.2011.08.013>
- Zhang, X., Bai, Q., & Dong, L. (2010). A New Hybrid Path Relinking Algorithm for the Vehicle Routing Problem. In *2010 International Conference on Intelligent Computation Technology and Automation* (Vol. 1, pp. 234–237). <https://doi.org/10.1109/ICICTA.2010.700>
- Zitzler, E. (1999). *Evolutionary algorithms for multiobjective optimization: methods and applications*. Eidgenössische Technische Hochschule Zürich.
- Zitzler, E., Laumanns, M., & Bleuler, S. (2004). A Tutorial on Evolutionary Multiobjective Optimization. In X. Gandibleux, M. Sevaux, K. Sörensen, & V. T'kindt (Eds.), *Metaheuristics for Multiobjective Optimisation* (pp. 3–37). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Zitzler, E., & Thiele, L. (1998). Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Case Study. In *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature* (pp. 292–304). London, UK, UK: Springer-Verlag. Retrieved from <http://dl.acm.org/citation.cfm?id=645824.668610>

APPENDIX 1 – Pareto Front Results for the Giant Tour Representation

Table A1- 1 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT1

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
524.61	775.82	524.611	764.59	524.611	764.59
536.88	751.11	524.629	764.266	525.72	763.943
		524.81	763.518	525.902	762.501
		525.902	762.501	527.011	761.854
		527.011	761.854	527.109	756.562
		527.109	756.562	527.303	755.443
		527.126	756.238	528.399	754.473
		527.303	755.443	528.593	753.354
		527.32	755.12	529.703	752.707
		527.502	754.371	531.4	751.745
		528.593	753.354	547.535	750.709
		529.703	752.707	549.165	749.811
		531.4	751.745	550.673	749.717
		536.881	751.112	553.303	749.212
		547.535	750.709	553.565	748.792
		550.427	749.784	553.959	747.375
		550.673	749.717		
		550.821	748.367		
		553.959	747.375		

Table A1- 2 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT2

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
835.45	1209.36	846.828	1211.32	846.407	1208.47
874.7	1179.53	847.128	1209.96	848.669	1202.89
		847.775	1208.25	865.045	1201.8
		848.074	1206.88	865.069	1199.82
		851.677	1206.71	865.748	1199.75
		851.701	1206.37		
		855.304	1206.19		
		856.992	1201.96		
		857.292	1200.6		

	860.919	1200.08
	862.857	1199.18
	863.156	1197.81
	866.783	1197.29

Table A1- 3 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT3

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
826.14	1228.67	833.351	1193.59	832.443	1197.99
850.34	1147.83	834.566	1191.72	832.666	1196.89
		834.688	1187.81	833.104	1196.17
		835.257	1187.75	833.138	1194.66
		835.328	1179.67	834.371	1194.59
		836.336	1176.2	834.57	1194
		836.419	1174.77	834.624	1190.58
		837.085	1174.61	834.906	1189.12
		837.38	1174.22	835.057	1187.89
		837.427	1171.3	835.167	1187.27
		838.093	1171.14	835.402	1181.82
		838.388	1170.75	835.625	1180.71
		838.903	1170.59	836.063	1180
		839.157	1169.96	837.361	1178.51
		839.206	1169.86	837.584	1177.4
		839.753	1169.53	838.022	1176.69
		839.911	1167.12	844.844	1176.14
		840.576	1166.96	845.066	1175.29
		840.872	1166.57		
		841.64	1165.78		
		841.689	1165.68		
		842.237	1165.35		
		842.653	1165.05		
		843.201	1164.7		

Table A1- 4 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT4

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption

1034.37	1543.38	1058.05	1524.13	1044.68	1515.28
1061.83	1452.88	1058.16	1522.95	1045.21	1510.26
		1058.41	1522.2	1051.27	1500.22
		1058.54	1519.51	1051.79	1495.21
		1058.65	1518.33		
		1058.76	1517.45		
		1059.01	1516.66		
		1059.14	1513.97		
		1059.25	1512.79		
		1060	1512.04		
		1060.37	1511.63		
		1060.39	1511.11		
		1060.63	1509.47		
		1060.74	1508.29		
		1061.49	1507.54		
		1061.86	1507.13		
		1061.88	1506.6		
		1062.61	1506.38		
		1062.63	1505.86		
		1063	1505.45		
		1063.75	1504.7		
		1065.05	1503.31		
		1065.16	1502.12		
		1065.92	1501.38		
		1066.31	1500.44		
		1067.06	1499.69		
		1068.63	1499.04		
		1069.38	1498.29		

Table A1- 5 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT5

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1308.88	1954.24	1340.73	1899.47	1347.24	1991.03
1359.49	1844.87	1341.54	1897.16	1347.55	1924.03
		1341.93	1891.69	1347.56	1922.72
		1342.45	1890.05	1347.65	1922.25
		1342.6	1889.89	1347.89	1922.21
		1342.67	1887.84	1347.98	1921.73
		1343.19	1885.69	1348.41	1921.46

	1343.6	1885.53	1348.65	1921.42
	1343.99	1884.34	1348.74	1920.94
	1344.39	1884.18		
	1344.85	1883.36		
	1345.05	1883.21		
	1345.16	1882.99		
	1345.66	1882.25		
	1346.06	1882.09		
	1346.52	1881.27		
	1346.72	1881.12		
	1346.83	1880.89		

Table A1- 6 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT11

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1042.12	1557.13	1044.57	1533.97	1050.68	1533.94
1052.18	1513.48	1044.67	1530.97	1051.55	1530.11
		1044.71	1530.4	1051.83	1523.69
		1044.74	1530.33	1051.86	1523.62
		1044.8	1527.39	1052.1	1521.17
		1044.83	1527.33	1052.38	1514.74
		1045.12	1525.03	1052.4	1514.68
		1045.21	1522.03		
		1045.25	1521.45		
		1045.28	1521.39		
		1045.35	1518.45		
		1045.38	1518.39		
		1053.2	1516.22		

Table A1- 7 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for CMT12

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
819.56	1203.76	819.558	1180.95	820.921	1185.32
827.05	1174.02	819.965	1179.42	821.689	1185.1
		820.188	1179.27	821.912	1184.53

820.733	1179.19	822.047	1182.11
820.956	1179.04	822.27	1181.96
824.009	1178.91	822.815	1181.88
824.232	1178.76	823.038	1181.73
824.777	1178.69	823.922	1181.62
825	1178.54	826.091	1181.6
825.649	1175.92	826.314	1181.45
826.056	1174.39	826.859	1181.38
826.279	1174.24	827.082	1181.23
826.824	1174.17	827.966	1181.11
827.047	1174.02	828.138	1177.08
		828.361	1176.93
		828.906	1176.86
		829.129	1176.71
		830.013	1176.59

Table A1- 8 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_1

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
5645.17	8416.86	5923.23	8144.12	5916.72	8051.54
5696.02	7683.95	5923.96	8119.97	5917.27	8031.51
		5925.25	8115.13	5925.23	8020.89
		5925.31	8099.83	5925.38	8017.32
		5925.87	8097.31	5929.38	8006.43
		5926.59	8094.99	5934.09	8004.55
		5927.15	8092.46	5934.64	7984.51
		5927.21	8077.16	5936.78	7967.24
		5928.49	8072.32	5944.89	7953.05
		5929.35	8070.25	5948.89	7942.17
		5929.9	8066.25	5962.29	7935.48
		5930.03	8065.37		
		5931.31	8060.53		
		5932.04	8059.34		
		5932.17	8058.46		
		5932.72	8054.46		
		5933.45	8053.62		
		5934.8	8051.22		
		5934.86	8047.55		

5936.86	8042.54
5938.84	8040.83
5938.94	8039.31
5939	8035.63
5939.68	8030.75
5941.76	8027.51
5941.82	8023.84
5943.9	8020.6
5944.46	8020.15
5945.8	8017.12
5946.53	8016.92
5946.77	8015.46
5947.45	8010.58
5949.53	8007.35
5949.59	8003.67
5951.67	8000.43
5953.57	7996.95
5958.28	7994.79
5962.98	7993.86
5965.12	7992.22
5969.83	7990.06
5974.54	7989.13
5981.81	7987.97
5986.52	7987.04

Table A1- 9 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_5

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
6460.98	9253.3	6506.57	8711.18	6460.98	8564.89
6460.98	8561.53	6508.66	8673.48	6565.89	8558.1
		6511.47	8670.96		
		6639.26	8638.51		
		6642.06	8620.02		
		6642.14	8619.56		
		6644.87	8606.74		
		6644.94	8606.28		
		6647.76	8600.77		
		6647.84	8600.32		
		6650.57	8587.49		

6650.65	8587.04
---------	---------

Table A1- 10 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_9

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
590.6	877.51	617.558	892.778	608.776	884.347
604.44	850.8	617.793	890.201	608.883	884.238
		618.331	889.94	609.342	883.885
		618.36	889.553	611.915	883.261
		618.899	889.292		
		618.957	884.016		
		619.153	882.838		
		619.209	882.2		
		619.405	881.022		
		619.877	880.873		
		620.025	880.501		
		620.497	880.352		
		620.852	879.98		
		621.324	879.825		

Table A1- 11 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_13

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
869.07	1288.34	905.513	1302.73	905.368	1314.06
896.93	1261.93	905.64	1301.9	905.77	1313.4
		905.775	1300.08	905.919	1310.36
		905.902	1299.25		
		906.103	1297.75		
		906.23	1296.92		
		906.354	1296.6		
		906.4	1296.23		
		906.555	1295.11		
		906.6	1294.73		
		906.682	1294.27		
		906.727	1293.9		

907.078	1293.79
907.124	1293.42
907.52	1293.14
908.083	1293.11
908.164	1293.04
908.479	1292.84
909.123	1292.74
909.504	1292.68
910.148	1292.58
918.859	1292.54
919.077	1292.5
919.205	1292.44
919.423	1292.39
919.515	1292.33
919.733	1292.28

Table A1- 12 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Giant Tour for Golden_17

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
710.19	1063.01	738.342	1068.32	732.726	1079.79
720.89	1027.21	738.379	1067.39	732.751	1068.55
		738.565	1065.79	733.28	1065.77
		738.602	1064.87		
		738.867	1064.7		
		738.903	1063.77		
		739.539	1063.27		

APPENDIX 2 – Pareto Front Results for the Split Vehicle Tour Representation

Table A2- 1 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT1

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
524.61	775.82	524.611	764.59	524.611	764.59
536.88	751.11	524.629	764.266	524.629	764.266
		524.81	763.518	524.81	763.518
		526.13	763.231	525.919	762.871
		531.025	757.621	527.303	755.443
		531.282	757.406	527.502	754.371
		531.945	755.277	528.822	754.085
		533.375	754.949	535.823	752.133
		535.823	752.133	536.881	751.112
		536.881	751.112	547.535	750.709
		547.535	750.709	549.165	749.811
		549.165	749.811	552.481	748.73
		552.481	748.73	557.237	748.651
		557.237	748.651		

Table A2- 2 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT2

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
835.45	1209.36	835.321	1201.45	859.288	1241.33
874.7	1179.53	835.445	1200.48		
		836.366	1192.56		
		838.443	1187.14		
		839.606	1184.37		
		841.64	1182.92		
		843.378	1182.03		
		844.364	1181.7		
		846.399	1180.25		
		852.349	1179.9		

Table A2- 3 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT3

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
826.14	1228.67	827.393	1180.94	830.786	1178.71
850.34	1147.83	828.084	1180.23	831.012	1177.15
		828.737	1174.59	831.832	1176.73
		829.943	1174.48	833.29	1175.9
		830.038	1168.22	833.375	1174.53
		830.261	1167.11	833.601	1172.98
		831.414	1165.94	835.879	1171.72
		831.637	1164.84	838.748	1167.8
		832.972	1164.45	839.567	1167.37
		836.951	1164.3	839.637	1160.15
		837.041	1162.4	846.618	1159.62
		838.376	1162.02		
		839.018	1161.7		
		840.354	1161.31		
		848.542	1161.19		

Table A2- 4 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT4

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1034.37	1543.38	1035.78	1473.11	1034.5	1484.42
1061.83	1452.88	1036.32	1469.2	1034.69	1482.04
		1037.01	1469.16	1035.02	1472.72
		1037.36	1468.76	1035.11	1468.82
		1037.41	1466.32	1036.81	1468.8
		1037.82	1465.96	1036.89	1467.47
		1039.93	1465.9	1038.42	1467.23
		1039.99	1465.62	1042.77	1465.65
		1040.44	1463.96	1043.01	1463.31
		1040.6	1461.46	1043.06	1462.57
		1041.52	1460.72	1044.24	1461.68
		1042.17	1460.71	1045.45	1458.95
		1042.54	1460.29	1045.61	1456.46
		1043.09	1459.97		
		1044.11	1459.9		

1044.5	1459.74
1045.03	1459.17
1046.21	1459.02
1047.13	1458.28
1047.78	1458.26
1048.7	1457.53
1049.73	1457.45
1050.11	1457.29
1050.64	1456.71

Table A2- 5 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT5

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1308.88	1954.24	1307.63	1890.62	1319.52	1909.27
1359.49	1844.87	1307.65	1889.65	1319.53	1908.28
		1307.66	1887.34	1319.61	1908.14
		1307.67	1886.37	1320.04	1906.33
		1307.68	1886.07	1320.05	1905.34
		1307.69	1883.76	1320.12	1905.2
		1307.71	1882.79	1321.06	1904.93
		1307.95	1882.24	1321.13	1904.79
		1308.56	1882.03	1323.02	1904.72
		1308.96	1881.83	1326.58	1904.28
		1309.09	1880.77	1327.08	1902.33
		1309.34	1880.21	1327.09	1901.35
		1309.95	1878.22	1327.8	1900.17
		1309.96	1877.25	1327.82	1899.18
		1310.21	1876.69	1327.89	1899.04
		1311.22	1876.28	1328.32	1897.23
		1311.58	1874.58	1328.33	1896.25
		1311.6	1873.59	1328.4	1896.1
		1311.68	1872.67	1329.35	1895.83
		1311.7	1871.69	1329.42	1895.69
		1311.94	1871.2	1331.3	1895.63
		1311.96	1870.84	1337.43	1895.42
		1312.2	1870.28	1337.45	1895.38
		1312.58	1870.02	1337.5	1895.28
		1312.71	1869.52	1337.52	1895.23
		1312.83	1869.47	1337.79	1894.45

	1312.95	1868.97
	1313.96	1868.55
	1315.18	1868.28

Table A2- 6 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT11

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1042.12	1557.13	1042.12	1529.46	1042.12	1529.46
1052.18	1513.48	1043.84	1529.04	1042.25	1525.88
		1043.87	1528.98	1042.28	1525.82
		1046.76	1526.31	1052.02	1525.3
		1047	1525.73	1052.05	1525.24
		1051.79	1512.86	1052.16	1524.18
				1052.19	1524.12
				1053.99	1524.09

Table A2- 7 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for CMT12

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
819.56	1203.76	819.558	1180.95	819.558	1180.95
827.05	1174.02	819.965	1179.42	819.965	1179.42
		825.649	1175.92	825.649	1175.92
		826.056	1174.39	826.056	1174.39

Table A2- 8 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_1

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
5645.17	8416.86	5699.01	8067.17	5765.9	7749.48
5696.02	7683.95	5699.17	8053.63	5768.72	7742.17
		5699.74	8017.54	5769.15	7731.66
		5699.9	8003.99	5770.18	7726.82
		5701.25	7988.54	5775.62	7725.07

5701.98	7986.46	5792.95	7722.1
5702.28	7981.05	5796.2	7711.3
5702.54	7977.07	5797.66	7711.1
5703.27	7974.6	5800.9	7709.53
5704	7968.7	5802.36	7709.32
5704.73	7966.23		
5705.06	7964.65		
5705.22	7951.1		
5705.84	7934.94		
5706.57	7932.87		
5706.87	7927.45		
5707.14	7923.48		
5707.87	7921.01		
5708.6	7915.1		
5709.33	7912.63		
5710.5	7906.79		
5711.23	7904.32		
5712.56	7901.99		
5713.86	7898.84		
5714.64	7895.07		
5715.91	7894.1		
5716.52	7893.98		
5716.54	7886.76		
5717.28	7886.21		
5717.81	7885.79		
5719.18	7877.9		
5722.84	7874.9		
5723.1	7871.53		
5724.62	7871.28		
5724.74	7868.21		
5725	7864.85		
5729.71	7862.5		
5729.98	7857.92		
5730.71	7837.9		
5732.17	7835.47		
5734.69	7834.82		
5736.15	7830.41		
5739.14	7824.8		
5739.23	7819.23		
5739.87	7810.33		
5739.96	7804.77		
5741.42	7795.3		
5745.4	7793.21		

5746.2	7789.66
5746.93	7784.24
5748.86	7769.55
5749.12	7763.4
5749.21	7757.43
5753.27	7752.82
5753.35	7746.59
5758.06	7745.14
5761.34	7740.56
5761.43	7734.83
5765.49	7729.46
5765.57	7723.72
5770.28	7722.23
5781.6	7718.11
5786.31	7717.9
5787.03	7717.67
5791.74	7716.07

Table A2- 9 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_2

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
8452.72	12234.33	8467.48	11570.9	8643.24	11676.5
8466.58	11172.71	8468.21	11475.5	8644.44	11523
		8468.93	11440	8646.37	11482.2
		8469.67	11410.9	8646.84	11425.4
		8470.39	11400.1	8647.57	11414.9
		8471.12	11393.8	8648.3	11408.1
		8472.53	11383.2	8650.44	11407.6
		8476.34	11369	8653.74	11407.2
		8477.02	11365.2	8653.86	11401.9
		8477.75	11363.9	8654.59	11368.7
				8656.52	11359.2
				8660.94	11354.8
				8661.19	11341.7
				8662.57	11336.9
				8663.58	11333.3
				8664.31	11327.9
				8665.2	11323.6
				8665.97	11320.5

	8667.43	11317
	8668.16	11311.3
	8675.01	11310.2

Table A2- 10 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_3

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
11045.81	16135.45	11036.2	14604	11036.2	14645.2
11076.16	14497.64	11037.4	14597.9	11037.4	14625.6
		11038.6	14554.9	11038.6	14623.5
		11039.8	14534.2	11039.4	14594.4
		11040.5	14533.7	11040.1	14588.1
				11042.9	14561
				11044.4	14553.8
				11046.5	14549.9
				11048.4	14545
				11050.5	14542.3

Table A2- 11 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_4

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
13630.52	20177.36	13624.5	18879.5	13624.5	18612.8
13728.29	18327.03	13625.3	18817.5	13625.3	18443.4
		13625.7	18443.6	13625.7	18394.5
		13626.9	18436.6	13630.5	18350.9
		13628.1	18417.7	13821.8	18348.8
		13629.8	18410.2	13833.4	18342.1
		13630	18396.4		
		13631.1	18395.1		
		13633.9	18378.9		
		13633.9	18392.8		
		13636.7	18368.4		
		13638.1	18365.3		
		13640	18334.1		
		13643.2	18327.4		

Table A2- 12 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_5

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
6460.98	9253.3	6460.98	8561.53	6460.98	8561.53
6460.98	8561.53	6556.51	8532.67		
		6559.31	8522.2		
		6562.21	8513.67		
		6567.91	8513.16		
		6570	8509.69		

Table A2- 13 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_6

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
8413.82	12316.21	8412.9	11126.5	8412.9	11117
8416.13	11102.22	8413.36	11116	8413.36	11114.3
		8413.82	11115.8		
		8414.29	11111.7		
		8421.46	11110.3		
		8546.34	11106.6		
		8546.8	11102.1		
		8547.26	11101		
		8548.73	11100.3		

Table A2- 14 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_7

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
10195.59	14975.25	10268.4	13940.2	10276.4	13884.2
10208.52	13422.16	10270.2	13898.1	10277.3	13864
		10271.1	13869.2	10278.1	13856.6
		10288.4	13855.6	10279	13845.1
		10289.6	13840.8	10284.3	13843.1
		10291.4	13840.4	10286	13821.1
		10292.3	13829.6	10288.7	13819.7

10294.7	13812.8	10290.5	13813.5
10295.8	13812.3	10310.3	13802.6
10296.5	13803.1	10439.2	13600
10297.4	13794.8	10441	13596.9
10439.2	13772.8	10450.4	13595.6
10440.1	13756.2		
10442.8	13755		
10443.7	13744.3		
10446.1	13729.1		
10448.2	13728.7		
10449.9	13726.6		
10456.2	13723.8		
10457.1	13716.5		
10461.9	13715.7		
10462.1	13707.8		
10464	13705.9		
10464.9	13694.6		
10472.9	13691.9		
10473.6	13683.7		
10476.3	13682.5		
10483.3	13679.4		

Table A2- 15 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_8

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
11689.1	17396.39	11926.1	16529.8	11878.6	15871.9
11774.5	15928.26	11928.5	16476.7	11880.9	15862.5
		11930.3	16443.2	11885.1	15834.5
		11933.2	16420.1	11885.7	15826.3
		11933.8	16370.6	11886.3	15819.9
		11935.8	16362.7	11887.5	15816.8
		11940.5	16349.2	11888.7	15814.8
		11941.1	16341.8	12089.4	15810.1
		11948.8	16332.5	12094.7	15773.6
		11952.5	16319.5		
		11953	16307.3		
		11953.1	16296.2		
		11954.2	16261.7		
		11954.8	16244.8		

11957.5	16226.6
11959.8	16197.4
11960.5	16197.4
11964.8	16195.9
11965.7	16187.1
11974.3	16179
11985.1	16168
11995	16155.6
12103.4	16055.9
12103.8	15936.3
12104.4	15933.6
12105.5	15930.5
12106.2	15926.5
12111.7	15919.8
12115.1	15902.7
12121	15896.2
12130.6	15880.2
12134.4	15875.7
12135	15868.9
12137.4	15867.4

Table A2- 16 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_9

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
590.6	877.51	597.39	873.718	605.79	878.313
604.44	850.8	598.108	871.305	608.825	876.153
		598.234	869.727	608.985	874.186
		598.747	869.658	609.691	873.176
		602.575	867.188	609.943	872.963
		602.88	866.819	621.271	872.797
		603.096	866.495	621.982	872.631
		603.401	866.126		
		603.977	865.878		
		604.374	865.603		
		604.724	865.072		
		605.029	864.703		
		605.606	864.456		

Table A2- 17 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_10

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
750.18	1113.7	760.344	1108.89	774.814	1128.58
773.6	1083	761.32	1108.08	774.94	1128.41
		761.645	1107.09	775.725	1125.74
		762.502	1107	775.85	1125.56

Table A2- 18 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_11

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
931.21	1406.91	951.467	1378.77	965.932	1385.98
972.59	1352.31	952.043	1378.65	965.944	1383.88
		952.232	1378.62	966.196	1383.5
		952.638	1378.13	966.448	1383.2
		952.89	1377.93		
		953.404	1377.73		
		954.484	1377.69		

Table A2- 19 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_12

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1127.18	1682.76	1161.02	1692.5	1189.62	1714.78
1161.33	1630.81	1161.48	1691.49	1189.69	1713.02
		1161.57	1689.15	1189.77	1712.78
		1161.85	1688.94	1189.8	1711.96
		1169.29	1688.75	1189.87	1711.73
		1169.98	1688.41	1190.49	1711.53
		1170.31	1687.94		

Table A2- 20 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_13

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
869.07	1288.34	877.419	1267.29	890.671	1283.77
896.93	1261.93	877.468	1264.86	891.083	1283.53
		877.518	1262.42	891.71	1283.4
		877.645	1261.8	892.337	1283.27
		877.922	1261.3	892.954	1281.58
		879.076	1261.14	893.366	1281.33
		879.203	1260.51	893.993	1281.2
		879.481	1260.01	894.822	1279.75
		881.751	1259.54	895.234	1279.5
		882.229	1258.37	895.86	1279.37
		882.64	1257.91	896.168	1278.91
		883.441	1257.69	896.58	1278.66
		883.732	1257.39	897.207	1278.53
		883.853	1257.23	897.752	1278.31
		884.944	1256.71	898.378	1278.18
		886.696	1256.59		
		886.82	1255.07		

Table A2- 21 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_14

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1101.51	1639.14	1118.83	1624.26	1113.98	1615.17
1134.52	1595.48	1119.16	1623.76	1114.03	1612.74
		1119.63	1623.5	1114.66	1612.61
		1119.91	1611.89	1114.96	1612.1
		1120.04	1611.26	1115.01	1609.67
		1120.57	1611.14	1115.63	1609.54
		1120.61	1609.91	1116.45	1609.51
		1120.73	1609.28	1117.48	1608.51
		1120.89	1608.82	1117.84	1607.83
		1121.02	1608.19	1118.46	1607.7
		1121.41	1606.3	1118.68	1607.56
		1121.69	1605.9	1119.41	1607.27
		1121.72	1605.4	1120.03	1607.14

	1121.82	1605.27	1120.85	1607.12
	1122.14	1604.85		
	1122.38	1604.66		
	1122.48	1604.52		
	1122.79	1604.11		
	1123.3	1603.66		
	1123.43	1603.03		
	1123.75	1602.62		

Table A2- 22 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_15

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1363.42	2034.08	1410.35	2040.42	1396.16	2016.61
1412.67	1970.43	1410.47	2036.79	1396.31	2014.2
		1410.8	2036.27	1396.36	2013.92
		1410.9	2035.7	1396.46	2013.62
		1413.96	2023.67	1396.5	2011.51
		1414.06	2021.98	1396.66	2010.94
		1414.1	2021.43	1396.79	2009.63
		1414.2	2019.23	1396.95	2009.05
		1414.35	2017.73	1397.15	2009.01
		1414.51	2017.15	1397.18	2008.87
		1414.67	2016.48	1397.31	2008.44
		1414.83	2015.88	1397.34	2008.3
		1415.33	2015.42	1397.54	2008.26
		1415.49	2014.64	1397.69	2007.68
		1416.76	2014.57	1398.45	2007.16
		1416.92	2014		
		1417.42	2013.33		
		1417.58	2012.75		
		1417.65	2012.28		
		1417.81	2011.7		
		1418.31	2011.03		
		1418.47	2010.46		
		1420.4	2009.93		

Table A2- 23 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_16

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1646.14	2439.68	2271.2	3061.39	2264.52	3231.23
1705.25	2391.12	2271.35	3060.88	2264.68	3230.78
		2271.41	3059.28	2264.91	3230.42
		2271.56	3058.58	2265.07	3229.87
		2271.84	3057.86	2265.35	3208.97
		2287.03	3053.45	2265.47	3092.41
		2287.04	3053.43	2265.62	3091.97
				2265.76	3060.97
				2265.92	3060.65
				2267.21	3039.15
				2267.37	3038.83
				2267.95	3027.22
				2268.27	3027.16
				2268.36	3026.92
				2268.58	3026.54
				2268.67	3004.59
				2268.99	3004.52
				2269.08	3004.28
				2269.3	3003.91
				2269.63	3003.77
				2269.72	3003.6
				2270.04	3003.46
				2270.36	3003.32

Table A2- 24 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_17

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
710.19	1063.01	725.741	1060.37	724.952	1053.13
720.89	1027.21	725.89	1059.91	724.979	1052.66
		726.019	1057.68	727.191	1052.28
		726.169	1057.21	727.679	1052.18
		726.329	1055.75		
		726.675	1055.72		
		726.84	1054.97		

726.867	1054.5
726.954	1053.03
727.592	1052.75
728.957	1052.59

Table A2- 25 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_18

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1006.69	1510.25	1036.8	1506.14	1036.77	1504.31
1025.54	1462.31	1036.86	1504.75	1037.28	1504.16
		1037.29	1503.84	1037.3	1503.28
		1037.3	1502.45	1041.1	1503.07
		1037.45	1502.27	1041.45	1502.56
		1037.6	1501.99	1041.46	1502.54
		1037.79	1501.43	1045.66	1502.34
		1037.85	1500.9	1045.82	1502.31
		1037.92	1500.36	1046.31	1500.66
		1037.98	1499.83	1046.48	1500.63

Table A2- 26 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_19

Xiao et al., 2012		NSGA-II		ϵ-MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1377.58	2059.98	1426.47	2087.63	1426.33	2082.35
1421.28	2007.62	1426.75	2086.97	1426.35	2080.76
		1427.03	2086.32	1426.59	2080.64
		1428.4	2084.42	1426.63	2080.1
		1428.53	2084.39	1426.86	2079.98
		1428.63	2084.3	1430.15	2079.53
		1428.68	2083.77	1430.17	2077.93
		1428.69	2081.16	1430.41	2077.82
		1428.74	2080.54	1430.45	2077.28
		1428.98	2080.42	1430.68	2077.16
		1429.02	2079.88		
		1429.25	2079.76		
		1432.51	2078.34		
		1432.53	2076.93		

1443.01	2076.42
---------	---------

Table A2- 27 Pareto Front Results of the adapted NSGA-II and the adapted ϵ -MOEA with Split Vehicle Tour for Golden_20

Xiao et al., 2012		NSGA-II		ϵ -MOEA	
Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption	Total Distance	Total Fuel Consumption
1849.6	2753.97	1906.86	2800.13	1912.02	2803.99
1904.59	2687.85	1907.16	2800.09	1912.3	2803.76
		1907.25	2797.46	1913.45	2803.57
		1907.45	2778.02	1913.51	2802.87
		1907.49	2776.62	1913.71	2802.35
		1907.75	2775.91	1913.85	2802.05
		1908.67	2775.3	1914.06	2801.53
		1908.93	2774.59	1914.35	2801.37
		1912.26	2773.09	1915.79	2795
		1912.52	2772.39	1915.94	2794.15
		1913.44	2771.77	1916.01	2794.07
		1913.7	2771.07	1916.21	2793.8
				1916.28	2793.73
				1916.55	2793.48
				1920.71	2790.62
				1920.78	2790.55
				1920.98	2790.28