MASTER

Multi-objective carbon emission-based VRP using the Cross-Entropy Method

Naber, P.

Award date:
2016

Eindhoven, December 2015

Multi-objective carbon emission-based VRP using the Cross-Entropy Method

by
Peter Naber

BSc Industrial Engineering & Management Science
Student identity number 0615775

in partial fulfilment of the requirements for the degree of

Master of Science
in Operations Management and Logistics

Supervisors
Prof. dr. Jan Fransoo, Eindhoven University of Technology, OPAC
Dr. J.C. (Josué) Velázquez Martínez, Monterrey Tech

TUE. School of Industrial Engineering.
Series Master Theses Operations Management and Logistics

Subject headings: vehicle routing problem, multi-objective optimization, carbon emissions, cross-entropy method, metaheuristic, computational complexity

Abstract

This study develops a new multi-objective model to investigate the trade-off between carbon emissions and cost for the vehicle routing problem. The proposed model is based on the activity-based estimation model presented in the NTM methodology. This model uses the distance and traffic of each arc coupled with the load of the vehicle in order to estimate the total carbon emissions. This model is solved using an adaptation of the multi-objective cross-entropy method.

Management Summary

As the incentive for companies to reduce carbon emissions grow so does the need for ways to model and implement more eco-friendly business solutions. Freight transportation is one of the biggest and still growing contributors to greenhouse gas pollutants. One of the most investigated problems in the field of transportation is the Vehicle Routing Problem (VRP). The main goal of the VRP is to minimize the total distance or cost of transportation. Many of the factors included in the VRP also affect carbon emissions. In order to reduce carbon emissions in the field of transportation we propose to include carbon emissions in the VRP. Including carbon emissions in the VRP requires accurate and effective carbon emission estimation. Carbon emissions dependent on a large number of factors and requires a great detail of the involved parameters which can result in very complicated models. In this paper we present a model which includes carbon emissions based on the NTM methodology. This methodology is based on an activity-based estimation model which increases the applicability of the overall model as it doesn't rely on very detailed or instance dependent parameters. Since companies are likely to be interested in the cost that will come with the carbon emissions reduction this research aims to find the trade-off between carbon emissions and cost and requires a multi-objective optimization technique. This trade-off is what ultimately decides if carbon reductions are both feasible and interesting for the company.

We therefore present the following research question:

How can a company define the structure of the Pareto frontier to trade-off cost and carbon emissions when deciding vehicle routings?

Model and solution methods

This report presents the Multi-Objective Emission Vehicle Routing Problem (MO-EVRP) which includes cost and carbon emissions as the objectives. Cost as a function of total travelled distance and carbon emissions based on the NTM methodology. Carbon emissions within the NTM methodology are estimated using a function based on the travelled distance, vehicle load and traffic conditions. This model is solved using the adaptation of the cross-entropy method (Rubinstein and Kroese, 2004) presented by Bekker and Aldrich (2011) which enables the algorithm to solve multi-objective optimization problems. This algorithm is further adapted to be able to apply it to the combinatorial multi-objective VRP.

Results

Results show that minimization of cost and minimization of carbon emissions results in different optimal vehicle routings. The objective function of carbon emissions has two more parameters then the cost function, namely load and traffic related variables. Both load and traffic related parameters affect optimal routings for the carbon emissions. We identified that the standard deviation of the customer demand has a big influence on possible carbon emissions savings. In situations where the customer demand is relatively even there is little space for carbon emission reductions and in situations with a high customer demand diversity there is more space for carbon emission reductions. When only accounting for load diversity we can see that emissions reductions are relatively low. In the conducted experiments that only considered load diversity possible carbon emissions savings where between 0 and 1.2%. In the conducted experiments with a high load and high traffic diversity savings of up to 28.3% could be achieved.

Conclusion

Several experiments show that there is indeed a trade-off between cost and carbon emissions when optimizing vehicle routings for the MO-EVRP. In situations where there are is a high diversity in load and a high diversity in traffic between the arcs possible carbon emission savings can go up to 28.3%. The multi-objective nature of the research shows a clear trade-off between the distance/cost objective function and the fuel consumption/carbon emission objective function. One of the conducted experiments is presented in Figure 1.

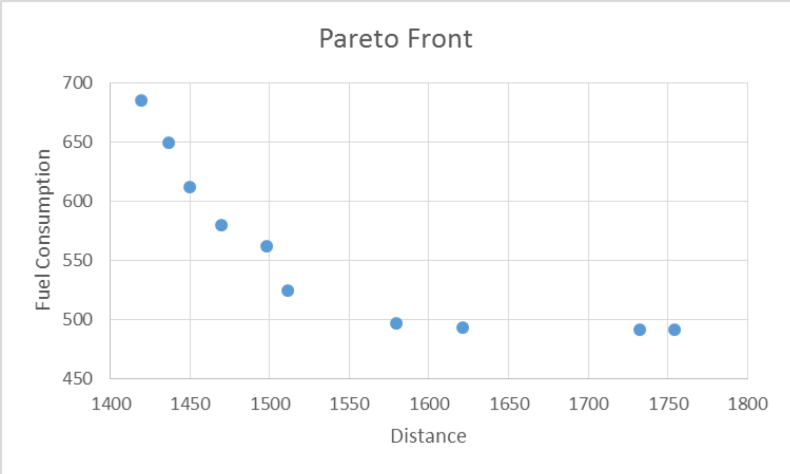

Figure 1 Trade-off between distance and fuel consumption

This multi-objective optimization method provides companies with the information that is needed to make informed decisions when it comes to possible carbon emissions savings. It gives an overview on the amount of carbon emissions that can be saved and at what cost these savings can be realized.

Preface

This thesis is the result of my graduation project for the MSc program in Operations Management and Logistics at the Eindhoven University of Technology. The road to success is never easy but always rewarding. Well, the road to finishing this study was a bumpy road, but nonetheless a success. That being said, I would like to thank my advisors, Prof. Fransoo and Prof. Velazquez, for their patience, understanding and for not giving up on me. Both of my supervisors managed to bring up the best of me enabling me to accomplish this last step of my studies. For this, I will always be grateful.

My parents have always been a great support and stood by me during the most difficult times. They walk next to me in every step of this journey. All of our discussions and laughs are imprinted in this document.

Last, but definitely not least, I would like to thank my family, friends and girlfriend for their constant support. This thesis marks the end of my MSc studies in Operations Management & Logistics at the Eindhoven University of Technology, the time has come to move forward!

Peter Naber

Contents

List of Tables

List of Figures

Abbreviations and terminology

TERM	Meaning
CEM	Cross-entropy Method
CMEM	Comprehensive Modal Emissions Model
CO_2	Carbon Dioxide
COPERT	Computer Programme to calculate Emissions from Road Transport
EA	Evolutionary Algorithm
GHG	Greenhouse Gas Protocol
IFCM	Instantaneous Fuel Consumption Model
MEET	Methodologies for Estimating air pollutant Emissions from Transport
MOCO	Multi-objective Combinatorial Optimization
MOEA	Multi-objective Evolutionary Algorithm
MOO	Multi-objective Optimization
MOOP	Multi-objective Optimization Problem
NSGA-II	Elitist Non-dominated Sorting Algorithm
NTM	Network for Transport Measures
PHEM	Passenger car and Heavy duty vehicle Emission Model
PRP	Pollution Routing Problem
TSP	Traveling Sales Problem
VEGA	Vector Evaluated Genetic Algorithm
VRP	Vehicle Routing Problem
VRPRB	Vehicle Routing Problem with Route Balancing

One

Introduction

Problem Statement

Transportation is one of the biggest factors in today's every growing pollution problem. Road transport alone accounts for almost one fifth of the total CO_2 emissions in the European Union (EU) in which freight transport takes 30% to 40% (EC DG Climate Action, 2010). The European Union introduced the EU emissions trading system in which companies receive or buy emission allowances with the purpose to effectively reduce carbon emissions (EU ETS, 2005). This gives companies, besides a social incentive, a financial incentive to reduce their carbon emission levels. The first step in reducing carbon emissions is measuring actual emissions. The most common approach to measure emissions is the Greenhouse Gas Protocol (GHG 2011). The Greenhouse Gas Protocol categorizes emissions in 3 broad scopes (see appendix A). Scope 1 includes all direct emissions. Scope 2 includes all indirect emissions from purchased electricity, heat or steam. Scope 3 includes other indirect emissions from all other sources such as purchased materials, purchased transportation & distribution, outsourced activities, waste disposal etc. Methods to measure emissions within each scope may vary wildly. While for scope 1 and 2 emission estimates or measurements based on actual fuel usage are easy to obtain, this is much more difficult for scope 3 emissions. Scope 3 emission measurements are mostly based on reference parameters which are calculated per industry activity. For this reason emission measurements for transportation, which mostly lie in scope 3, can be difficult to measure (GHG 2011).

Reducing carbon emissions in the transportation sector can be achieved by adapting the vehicle routing problem (VRP) since many factors effecting transport emissions are also included in the VRP, for example distance and load. The VRP is concerned with finding optimal routes for a fleet of vehicles to serve a set of customers (Laporte, 2009). This problem can be seen in many real life instances, for example the distribution of goods to the end customer. To minimize transport emissions the VRP can be solved for carbon emissions instead of the original cost or distance objective function. The first step in order to include carbon emissions in the VRP is the estimation of carbon emissions. This estimation can be done is several ways, using either energy-based estimation models or activity-based estimation models.

While the VRP could be solved for carbon emissions as the only objective, decision makers are more likely to be also looking at the cost of a given solution. This means that there are multiple objective functions that have to be considered while solving the VRP. Within multiple-objective optimization there is not one best solution, but a set of efficient solutions. One way to solve multi-objective optimization problems is using the Pareto-optimal solutions. Pareto-optimal solutions or non-dominated solutions are those solutions where none of the objectives can be improved without worsening any of the other objectives (Goldberg, 1989). This set of non-dominated solutions is also called the Pareto front.

In this thesis we propose the multi-objective VRP and name it the Multi-Objective Emission Vehicle Routing Problem (MO-EVRP). In order to solve the multi-objective VRP and approximate the Pareto frontier we propose an adaptation of the cross-entropy method, first implemented by Rubinstein and Kroese (2004). The cross-entropy method was first adapted to the multi-objective setting by Bekker and Aldrich (2011). Since the algorithm implemented by Bekker and Aldrich only solves problems that are continuous in nature further adaptations are needed to be able to apply the algorithm to the combinatorial VRP.

Contributions and motivation

Including carbon emissions in the VRP is not a straight forward task. There are many factors affecting carbon emissions that can be included in the VRP. From literature research we conclude that the most important factors affecting carbon emissions for the VRP are speed, load and traffic related parameters. Literature research on green vehicle routing shows that there is a clear gap in activity-based estimation models that consider both load and traffic related parameters. This study aims to fill this gap by presenting an activity-based estimation method using the NTM methodology. This methodology has two main advantages. The first advantage is that is uses an activity-based estimation model which reduces the required amount of detailed input parameters. The second advantage is that is assumes a linear relationship between load and fuel usage. This improves the accuracy of estimation of the model when compared to other activity-based estimation models which most often categorizes the vehicles into empty, loaded or fully loaded vehicles.

While minimizing only carbon emissions in vehicle routings can result in interesting information most companies will more likely be more concerned about the cost of possible carbon emission reductions. In order to take this into account the model presented in this paper uses a multi-objective approach which makes it possible to show the trade-off between cost and carbon emissions. To solve the proposed multi-objective model we need a multi-objective optimization method. One of the more classical methods is the weighted-sum method. The weighted-sum method scalarizes a set of objective functions to turn the objectives into a single objective by giving a user-supplied weight to each of the objective functions. This single objective optimization problem is then solved for many different weight distributions to approach a set of Pareto optimal solutions. Many methods used for multi-objective optimization use evolutionary algorithms. Evolutionary algorithms use a population-based mechanism with uses biological inspired mechanisms to move from generation to generation to evolve to better solutions. In this study we implement an adaptation of another promising multi-objective optimization method, the cross-entropy method, and test the effectiveness of this method for the multi-objective VRP.

A general overview of the contributions of this study:

- Build a multi-objective model for the VRP that includes carbon emissions and cost as the objectives.
- Include carbon emissions in the model using both load and traffic related parameters for an accurate carbon emission estimation while maintaining a wide applicability of the model.
- Solve the proposed model to get further insight into the trade-off between carbon emissions and cost in the multi-objective VRP.
- Adapt the cross-entropy method to be able to apply it to combinatorial multi-objective VRP.

Research Question

Several studies show that carbon emissions are still rising in the transportation sector. In order to reduce the carbon emissions in this sector we propose to include carbon emissions in the VRP. In order to include carbon emissions in the VRP we need a model to effectively estimate carbon emissions while maintaining the applicability of the model. Since companies are likely to be interested in the cost that will come with the carbon emissions reduction this research aims to find the trade-off between carbon emissions and cost.

We therefore present the following research question:

- How can a company define the structure of the Pareto frontier to trade-off cost and carbon emissions when deciding vehicle routings?

Hypothesis: Including carbon emissions into the multi-objective vehicle routing gives decision makers a tool to investigate where possible carbon emission reductions can be made and at what cost they can be made.

In order to answer this question, we answer the following sub questions:

1. What is the most applicable way to include carbon emissions in the VRP?
2. How can we solve the multi-objective VRP?
3. How does our model compare to other models found in literature?
4. How can we interpret the obtained results regarding the trade-off between carbon emissions and cost?

Methodology

We conduct this study in the following stages. In order to give an appropriate estimation of carbon emissions in the VRP we first conduct a review on carbon estimation models. Here we compare the calculation methods and required data for the selected models. The next step is to introduce the multi-objective emission based vehicle routing problem (MO-EVRP) using cost and carbon emissions as the objective functions. To validate this model we compare the carbon emission objective with the objective found in the Pollution Routing Problem (PRP) (Bektaş and Laporte, 2011) (Demir et al, 2014).

To be able to efficiently solve and find the Pareto frontier of the proposed MO-EVRP we implement an adaptation of the multi-objective cross-entropy method. The validation of this metaheuristic will be conducted in two stages. First we compare the results obtained by the MO-EVRP with the results obtained by a weighted-sum method. Second we compare the results obtained with the results obtained by the multi-objective evolutionary algorithm proposed by Jozefowiez et al. (2009).

After validating the model, we take a closer look at the proposed model and provide insight on the applicability of the model. We also provide possible extensions of the model which can make the model more applicable for a wider range of situations.

Thesis Outline

This thesis is structured as follows. Chapter 2 gives a short overview of the VRP in general and the introduction of carbon emission in the VRP. This chapter also describes solution methods for both the VRP and the multi-objective VRP.

Chapter 3 gives a short overview of the cross-entropy method and the application of the cross-entropy method in the multi-objective setting.

Chapter 4 and chapter 5 introduce the Multi-Objective Emission Vehicle Routing Problem (MO-EVRP) and the cross-entropy-method used to solve this MO-EVRP respectively.

In chapter 6 both the proposed MO-EVRP model and the cross-entropy method for the MO-EVRP are validated. The model is validated by comparison with an energy-based estimation method to validate the accuracy of the model. The cross-entropy method is validated by comparison with the weighted-sum method and a multi-objective evolutionary algorithm to validate the effectiveness of the algorithm for the multi-objective VRP.

Two

The Vehicle Routing Problem

The vehicle routing problem (VRP) is one of the most well-known and studied combinatorial optimization problems in the field of operations (Toth and Vigo, 2001). A Google scholar search as of today on the words “Vehicle Routing Problem” gives approximately 32.200 results. The VRP can be defined as the problem of designing least-cost delivery routes from a depot to a set of geographically scattered customers, subject to side constraints (Laporte, 2009). Since the introduction of the VRP by Dantzig and Ramser in 1959 many variations of the problem and different solutions techniques have been proposed. This chapter gives an overview of the more recent research in green vehicle routing, multi-objective vehicle routing problems and solution methods for the vehicle routing problem.

1.

Green Vehicle Routing Problem

As a result of combustion, road vehicles emit a range of atmospheric pollutants. The amount of CO_2 is directly proportional to the amount of fuel that is consumed by the vehicle. The fuel usage of a road vehicle depends on a variety of factors including speed, load and traffic (Demir et al., 2014). In order to include CO_2 emissions in the VRP the CO_2 emissions have to be estimated.

Multiple models have been proposed to estimate CO_2 emissions of road transport which can be divided in energy-based estimation and activity-based estimation models. Energy-based estimation models include the instantaneous fuel consumption model (Bowyer et al., 1985), the Passenger car and heavy duty emission model (Boulter and McCrae, 2007), and the comprehensive modal emissions model (Barth et al., 2005). Activity-based estimation models are average speed estimation models that estimate emissions according to a database. Activity-based models include the GHG protocol (GHG Scope 3 Calculation Guidance 2013), Methodologies for estimating air pollutant emissions from transport (MEET, 1999), Computer Programme to calculate Emissions from Road Transport (COPERT4), Network for Transport Measures (NTM), (NTM for road transport, 2010).

Several studies have studied the effects of emissions minimization on an operational level by including vehicle emissions in the VRP. The most prominent factors that determine carbon emissions for a given arc are speed, load and traffic conditions. An overview of recent research according to the type of model and included factors is given in Table 2-1.

	Speed	Load	Traffic
Energy-based			
Palmer (2007)	X		
Urquhart et al. (2010)	X		x
Bektaş and Laporte (2011)	X	x	
Demir et al. (2014)	X	x	
Activity-based			
Tavares et al. (2008)		x	
Figliozzi (2010)	x		x
Scott et al. (2010)		x	
Xiao et al. (2010)		x	
Jabali et al (2012)	x		x

Table 2-1: Green Vehicle Routing Overview

As can be seen in Table 2-1 there is a clear gap in literature research for estimation models that consider both load and traffic related parameters. An important note here is that he studies of Tavares et al. (2008) and Scott et al. (2010) only use load assumptions of 0, 50 or 100% load correction in their respective optimization models.

Multi-objective Vehicle Routing

While the VRP could be solved for carbon emissions as the only objective, decision makers are more likely to be also looking at the cost of a given solution. This results in a multi-objective optimization problem. When there is more than one objective there can be more than one optimal solution. If there are conflicting objectives there is no longer a single global optimum but a set of trade-off optimum depending on the preference of the decision maker. In order to classify these trade-off solutions we introduce the concept of Pareto-ranking (Goldberg, 1989). Pareto optimality means that if for a solution there is no other solution that would improve an objective function without worsen one or more of the other objective functions. When all those solutions are plotted in the objective space, these values are known as the Pareto front. This phenomenon is visualized in Figure 2-1.

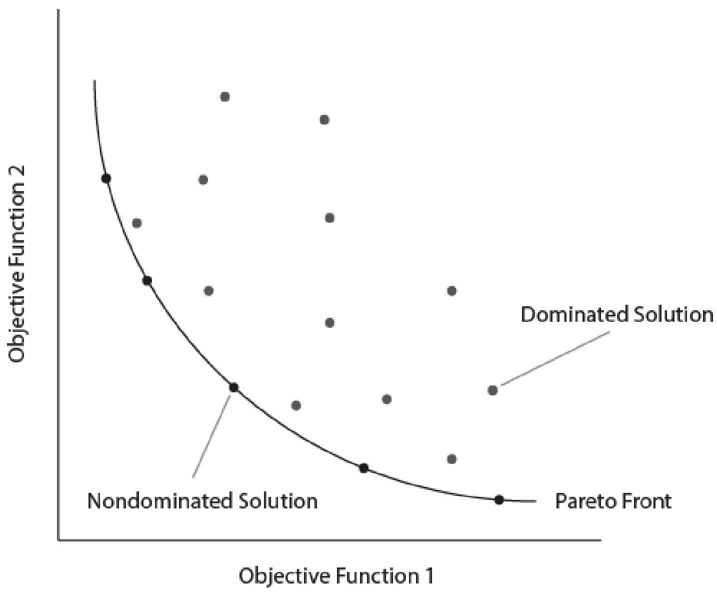

Figure 2-1: The Pareto Front (Deb, 2001)

This results in the notion that multi-objective optimization has two goals, one is to find a set of solutions as close as possible to the true Pareto-optimal set and to find a set of solutions as diverse as possible (Deb, 2001). If we would solve the problem of type:

$$minmize\ f(x) := [f_1(x), f_2(x), \dots, f_m(x)]$$

$$g_i(x) \leq 0, \quad i = 1,2,\dots,p$$

$$h_i(x) \leq 0, \quad i = 1,2,\dots,q$$

where $\boldsymbol{x} = [x_1, x_1, \dots, x_n]$ is the vector of decision variables, f are the objective functions and g and h are the restrictions. To describe the concept of Pareto optimality the following definitions are given (Coello, 2007):

Definition 1. Given two vectors $\boldsymbol{u}$ and $\boldsymbol{v} \in \mathbb{R}^m$, we say that $\boldsymbol{u} \leq \boldsymbol{v}$ if $u_i \leq v_i$ for $i = 1,2,\dots.,m$, and that $\boldsymbol{u}$ dominates $\boldsymbol{v}$ (denoted by $\boldsymbol{u} \prec \boldsymbol{v}$) if $\boldsymbol{u} < \boldsymbol{v}$ and $u \neq \boldsymbol{v}$**.**

Definition 2: We say that a vector of decision variables $\boldsymbol{x}^* \in \mathcal{X} \subset \mathbb{R}^n$ is non-dominated with respect to $\mathcal{X}$, if there does not exist another $\boldsymbol{x}' \in \mathcal{X}$ such that $f(\boldsymbol{x}') \prec f(x)$.

Definition 3. A vector of decision variables $\boldsymbol{x}^* \in \mathcal{F}$ ($\mathcal{F}$ is feasible region) is Pareto optimal if there does not exist another $\boldsymbol{x} \in \mathcal{F}$ such that $f(\boldsymbol{x}) \prec f(\boldsymbol{x}^*)$**.**

Definition 4. The $Pareto\ optimal\ set$ P^* is defined by $\mathrm{P}^* = \{\boldsymbol{x} \in \mathcal{F} | \boldsymbol{x}\ is\ Pareto\ optimal\}$.

Definition 5. The $Pareto\ front$ P_T^* is defined by $\mathrm{P}_T^* = \{f(\boldsymbol{x}) \in \mathbb{R}^n | \boldsymbol{x}\ is\ Pareto\ optimal\}$.

Solution Methods

Given the complexity of the problem exact algorithms can only solve smaller instances with up to 100 customers (Baldacci et al., 2008). Therefor one must rely on heuristics in order to solve the VRP. Classic construction heuristics include the savings algorithm (Clarke and Wright, 1964), sequential insertion (Christofides et al., 1969), sweep algorithm (Gillett and Miller, 1974) and the general assignment heuristic (Fisher and Jaikumar, 1981). Classic construction heuristics only explore a relatively small area of the search space but typically generate good solutions in moderate computing times. In recent years a multitude of metaheuristics have been proposed for the VRP with promising results. Metaheuristics emphasize on exploring the solution space to identify good solutions. Metaheuristics successfully applied to the single-objective VRP include simulated annealing (Kirkpatrick, 1983), tabu search (Glover 1989), genetic algorithm (Holland, 1975), ant colony (Dorigo et al., 1992) and cross-entropy (Rubinstein, 1999). Metaheuristics combine more sophisticated neighborhood search methods, memory structures and recombination of solutions to advance the search process. Metaheuristics which work on a population of solutions are better suited for multi-objective optimization problems since they are able to find multiple Pareto-optimal solutions in one single run. This also causes evolutionary algorithms like the vector evaluated genetic algorithm (VEGA)(Schaffer, 1985) or the elitist non-dominated sorting algorithm (NSGA-II)(Deb, 2001) to be popular for multi-objective optimization problems.

Summary

Since decision makers do not always have access to all the detailed information that energy-based estimation models require we propose the use of an activity-based estimation model in order to include the carbon emission in the VRP. Given the lack of models that consider both load and traffic related parameters we suggest the use of the NTM methodology to formulate the model. The NTM methodology assumes a linear correlation between load and carbon emissions which simplifies the model. The NTM methodology also provides all the required information needed in order to estimate carbon emissions and is updated using the most recent versions of the Handbook Emissions Factors for Road Transport (HBEFA 3.2).

Given the multi-objective nature of the problem we suggest the use of a metaheuristic that works on a population of solutions and uses a global search procedure rather than a local search procedure. The cross-entropy method meets all these criteria and proves to be an efficient metaheuristic to solve multi-objective problems (Bekker and Aldrich, 2011).

Three

Cross-entropy method

Multi-objective optimization problems are best solved using a metaheuristic that works on a population of solutions and use a global search procedure rather than a local search procedure in order to ensure a complete exploration of the solution space. The cross-entropy method is one of those methods and showed promising results for multi-objective optimization problems. In this chapter we describe the cross-entropy method, the application of the cross-entropy method to the VRP and the adaptation to multi-objective optimization.

Cross-entropy description

The cross entropy method is a stochastic meta-heuristic first described by Rubinstein and Kroese (1999). The CE method is an adaptive algorithm based on the Kullback-Leibler distance that can solve combinatorial optimization and continuous multi-extremal problems. In order to do so, the deterministic problem is first translated to an associated stochastic problem. The CE method can be described as an iterative procedure where each iteration can be broken down in two phases (Rubinstein and Kroese, 2004):

1. Generate a random data sample according to a specified random mechanism.

2. Update the parameters of the random mechanism based on the data to produce a "better" sample in the next iteration.

One of the fundamental ideas behind the CE method is the well-known importance sampling (IS) variance minimizing technique. Importance sampling introduces a set of parameters, also called reference parameters, in which a specified system is simulated in order to make the occurrence of a specific rare event more likely. The first step of the CE method is to associate an estimation problem with the original optimization problem. This step of the CE method is considered the most difficult step by many since a suitable probability family need to be found that is well fitted to the structure of the problem (Margolin, 2005). The probability distribution must be set up in such a way that it can produce feasible solutions for the problem. The effectiveness of the CE method is demonstrated in multiple problem cases including: buffer allocation (Alon et al., 2005), control and navigation (Helvik and Wittner, 2001) and vehicle routing (Chepuri and Homem-de-Mello, 2005) (Ma, 2011).

Cross-Entropy for the VRP

Rubinstein and Kroese are the first to apply the CE method to the TSP which forms the basis for the VRP. For a TSP of size N they assume a graph in which notes represent customers and the edges represent connections between the customers. Non existing edges can be assigned ∞. Every solution can be seen as a vector $X = (x_i, \ldots, x_{n)}$ where component i represents customer i. To generate random routes they use a Markov chain starting and ending at the depot. The transitions in this Markov chain are based on a $n \times n$ transition matrix P in which node $P_{(i,j)}$ gives the probability of the truck going from location i to location j. At every iteration the matrix P is updated according to the best generated routes increasing the chance of well performing routes to show up for the next batch of random generated routes. The simulation size, the amount of generated samples is set to $5n^2$, where n is the amount of customers. Since then multiple papers have investigated the effectives of the CE method for the VRP. Chepuri and Homem-de-Mello (2005) propose the CE method for the VRP with stochastic demand. They use a variable simulation size to overcome the added complexity. Ma (2011) proposes the use of the CE method for the VRP with time windows. They implement an additional local search procedure to avoid premature convergence.

Multi-objective Cross-Entropy Method

Bekker and Aldrich (2011) propose the cross-entropy method for multi-objective optimization (MOO CEM) which is an extension of the cross-entropy method (CEM) for single objective optimization. Since CEM is simple and generally converges quickly in the single objective setting, they argue that this method could also be used in the multi-objective setting. All objective values for every sample solution are added into a matrix, as presented in Figure 3-1.

Decision variables				Objectives				Rank
x_{11}	x_{12}	$\cdots$	x_{1n}	f_{11}	f_{12}	$\cdots$	f_{1n}	ρ_1
$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_{N1}	x_{N2}	$\cdots$	x_{Nn}	f_{N1}	f_{N2}	$\cdots$	f_{Nn}	ρ_N

Figure 3-1 Working Matrix of the MOO CEM (Bekker and Aldrich, 2011)

Within the CEM the best quantile of the performances is used to update the probability parameters. In order to compare the different solutions in the MOO CEM they apply the Pareto-ranking algorithm (Goldberg, 1989) which indicates how many solutions dominate that specific solution. Solutions that have a ranking value of less or equal to a specific threshold value are then included in a vector called Elite vector which contains the best performing solutions.

The values of the decision variables in the Elite vector are used to create class frequency histograms for each decision variable. These frequency histograms are then used in order to update the probability parameters such that the best solutions are more likely to influence the next random sample. For every iteration, the new population of possible solutions is formed proportionally according to the class frequencies for each decision variable. To prevent premature convergence there is a preset probability that the histogram frequencies are inverted resulting in an inverted frequency histogram. This ensures that search ranges that where given small proportions of population candidate allocations receive higher proportions of allocations and search ranges with higher proportions of population candidates receive fewer allocations. As the search progresses the number of classes can be increased to maintain good combinations of decision variables as the resolution of the decision variable spaces becomes finer.

After each iteration of the algorithm the Elite vector is ranked again using the Pareto ranking. Just as in the CEM the probability parameters are updating using a smoothing constant. In order to support exploration of the search the initial ranking threshold is relaxed and a threshold value of 2 is used. This means that solutions dominated by at most 2 solutions are included in the Elite vector. This ranking threshold can be reduced after one loop of the algorithm. When the algorithm terminates this threshold is set to zero and the Elite vector contains all the non-dominated solutions.

Multi-objective Combinatorial Optimization Cross-Entropy Method

While Bekker and Aldrich applied the CE method to a continuous multi-objective optimization problem, Velazquez-Martinez et al. () propose the multi-objective cross-entropy method for the combinatorial p-Median problem, the MOCO CEM. They identify two main issues when applying the MOO CEM to combinatorial problems. Since combinatorial problems involve either integer or binary variables, the truncated normal distribution as used in the MOO CEM is no longer applicable. In addition, the solutions in the MOO CEM are not limited by constraints while most combinatorial problems are constrained. To generate random samples they apply the Bernoulli distribution and test feasibility for every solution. To avoid premature convergence they inverse the probability vector with the same probability range as the MOO CEM.

Four

The Multi-Objective Emission Vehicle Routing Problem

In this chapter we present the multi-objective emission vehicle routing problem that aims to show the trade-off between carbon emissions and cost for the multi-objective VRP. In this model the cost of the VRP is a function of the total distance and does not take into account driver or vehicle costs. The carbon emission estimation is based on the NTM methodology.

Mathematical Representation

The Multi-Objective Emission Vehicle routing Problem (MO-EVRP) is based on the capacitated vehicle routing problem. We model this using two-index vehicle flow formulations (Toth and Vigo, 2001) using integer variables associated with each edge of the graph.

Let $G = (V, A)$ be a complete graph where $V = (0, \dots., n)$ is the vertex set and A is the arc set. Vertices $i = 1, \dots, n$ are the customers, also denoted as N and vertex $i = 0$ is the depot. The distance between every vertex is given by d_{ij}. We assume triangle inequality and that the graph is symmetric. This means that $c_{ik} + c_{kj} \geq c_{ij}$ for all $i, j \in V$ and that d_{ij} is equal to d_{ji}. Each customer i $(i = 1, \dots, n)$ is associated with a known nonnegative demand q_i. A set of M vehicles, each with capacity C, are available at the depot. To ensure feasibility we assume $q_i \leq C$ for each i. Vehicle routes start and end at the depot.

The goal consists of finding a collection of circuits (each corresponding to a vehicle route) that minimize the total cost and carbon emissions from the sum of those routes while satisfying that:

1. Each circuit starts and ends at the depot
2. Each customer vertex is visited by exactly one circuit
3. The sum of the demand of each circuit does not exceed the vehicle capacity C.

In order to include the carbon emissions as an objective function we use the Network for Transport and Environment methodology. The estimation of carbon emissions is based on the fuel consumption and the distance travelled. Fuel consumption is a function of the type of vehicle, the load factor, traffic parameter and the type of road.

Let N be a set of customers and V the same set of customers with V_0 as the depot. We define the following parameters:

q_i	= Demand at node $i \in N$
d_{ij}	= Distance between nodes $i \in V$ and $j \in V$
t_{ij}	= Traffic parameter between nodes $i \in V$ and $j \in V$
C	= Truck Capacity
M	= Number of available trucks
k	= constant emission factor (2,631 grams of CO_2/liter of fuel)

f_t^e = Fuel consumption of the empty vehicle
f_t^f = Fuel consumption of the fully loaded vehicle

We define the decision variable x_{ij} as:

$$x_{ij} = \begin{cases} 1, if\ a\ vehicle\ drives\ from\ i\ to\ j \\ 0, otherwise \end{cases} \tag{4.1}$$

For every edge$(i, j) \in V$ a nonnegative flow-variables is defined as:

$$y_{ij} = the\ vehicle\ load\ when\ a\ vehicle\ travels\ from\ i\ to\ j \tag{4.2}$$

From this we can construct the multi-objective vehicle routing model as:

$$minimize\ OF1 = \sum_{i \in V} \sum_{j \in V} d_{ij} x_{ij} \tag{4.3}$$

$$minimize\ OF2 = \sum_{i \in V} \sum_{j \in V} k \cdot d_{ij} \cdot [f_t^e \cdot x_{ij} + \frac{(f_t^f - f_t^e)}{C} \cdot y_{ij}] \tag{4.4}$$

Subject to

$$\sum_{i \in V} x_{i0} \leq M \tag{4.5}$$

$$\sum_{i \in V} x_{0j} = \sum_{i \in V} x_{i0} \tag{4.6}$$

$$\sum_{j \in V} x_{ij} = 1 \qquad \forall i \in N \tag{4.7}$$

$$\sum_{i \in V} x_{ij} = 1 \qquad \forall i \in N \tag{4.8}$$

$$\sum_{j \in N} y_{ji} - y_{ij} = q_i \qquad \forall i \in N \tag{4.9}$$

$$q_j x_{ij} \leq y_{ij} \leq (Q - q_i) x_{ij} \qquad \forall (i, j) \in A \tag{4.10}$$

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(S) \qquad \forall S \subseteq N, S \neq \emptyset \tag{4.11}$$

$$y_{ij} \geq 0 \qquad \forall i \in N \tag{4.12}$$

$$x_{ij} \in \{0,1\} \qquad \forall (i, j) \in A \tag{4.13}$$

Constraints 1.1 and 1.2 constraint the maximum amount of vehicles. Constraint 1.3 and 1.4 state that every customer must be visited once and only once. Constraint 1.5 and 1.6 define the arc flows. Constraint 1.7 is the capacity-cut constraint which impose the connectivity of the solution and vehicle capacity requirements where $r(S)$ is the minimum amount of vehicles needed to serve customer set S. Constraint 1.8 makes sure the flow variable is non-negative and constraint 1.9 constraints the binary nature of the decision variable.

Note that both objective functions rely on different variables. OF1 only depends on the total travelled distance and OF2 depends on the total travelled distance, total load that is carried over that distance and the road and traffic parameters on which that load is carried. In this model we assume that both road and traffic parameters are known in advance.

Five

The MOCO CEM for MO-EVRP

In this chapter we describe the application of the CE method to solve the MO-EVRP model that is presented in chapter 4. First we describe the solution representation for the MO-EVRP to be able to apply the CE method. Then we discuss the adaptions that are needed in order to apply the CE method to the multi-objective combinatorial VRP.

Solution representation

The first step in the MOCO CEM the definition of the representation of a solution for the VRP problem. To represent a solution for the VRP Rubinstein and Kroese (2004) use a vector $(X_0, \dots, X_n)$ to define the sequence of the nodes in the vehicle route. This route always starts and ends in the depot. In combination with the additional information of the demand at every node q_i and the capacity of the vehicle C it can be determined when the vehicle has to make additional returns to the depot. When at a certain node k, $q + q_2 + \cdots + q_k \leq C$ and $q_1 + q_2 + \cdots + q_{k+1} > C$, are true the vehicle must return to the depot between nodes k and $k + 1$. For a given demand structure $\{q_i\}$ and vehicle capacity C, this solution representation corresponds to unique vehicle routes.

However, with this representation not every possible route is included in the solution set. This representation is unable to represent routes where the vehicle returns to the depot before the next node would surpass the capacity. Therefor we propose the solution representation where every return of the vehicle to the depot is included in the solution vector.

$$X = (1,4,2,1,3,1) \tag{5.1}$$

For example equation 5.1 represents a solution where the solution consists of 2 routes starting and ending at the depot, namely $X = (1,4,2,1)$ and $X = (1,3,1)$. This change in representation has an effect on the route generation algorithm that is used to generate random samples.

Route Generation Algorithm

To generate random samples we use a route generation algorithm that generates routes based on a Markov-chain applied to a transition probability matrix as seen in equation 5.2, also termed a right stochastic matrix. Each entity is a nonnegative real number representing a probability with each row summing to 1. This matrix represents all the transition probabilities from going from one node to another. Given that nodes never transition to itself the diagonal entities are all equal to zero.

$$P_{ij} = \begin{matrix} P_{1,1} & P_{1,2} & \cdots & P_{1,n} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n,1} & P_{n,2} & \cdots & P_{n,n} \end{matrix} \tag{5.2}$$

In reality the VRP is highly constrained and routes depend on the feasibility of the transitions between the nodes. To generate feasible routes the route generation algorithm sets the transition probabilities of non-feasible transitions to zero and normalizes the remaining transition probabilities. Every node has to be served only once in every solution. This means that when the next random node in the route is selected it sets the transition probability from all nodes to the selected node to zero. However, with the proposed representation this no longer holds, as the depot node can be included multiple times in the route. To allow this to happen, the route generation algorithm checks whether the chosen node is equal to the depot in which case it does not set the probability from all other nodes to that node to zero. The route generation algorithm is given as follows:

Route Generation Algorithm

```
Set current node to 0 (depot) → a
While not all nodes are visited, do:
    For all nodes not yet visited, do:
        Check feasibility for vehicle load else p_ai is equal to zero
    Normalize transition matrix P
    Pick random destination node → b
        If b is equal to 0 Then Current load is equal to zero
        Else Current load is equal to current load plus demand at b
            p_ib is equal to zero
        End if
Set a = b
End while
```

MOCO CEM algorithm

Here we describe the MOCO CEM to solve the MO-EVRP. We start by creating an empty vector E that contains all the efficient solutions and iteration counter $t = 1$ and population size M. To generate random samples X_i for $i = 1, \dots, M$ we initialize the transition probability matrix $\hat{p}_{0,ij}$ as follows:

$$\hat{p}_{0,ij} = \begin{cases} \dfrac{1}{n-1}, & i \neq j \\ 0, & i = j \end{cases} \tag{5.3}$$

Using this matrix we generate feasible routes X_i using the route generation algorithm. We then calculate both objective functions and rank each solution according to the Pareto-ranking algorithm (Goldberg 1989). The solutions with a cutoff value of $th = 2$ (Bekker and Aldrich, 2011) are added to the elite vector E. With this elite vector we update the transition probability matrix $\hat{p}_{t,ij}$ according to:

$$\hat{p}_{t,ij} = \begin{cases} \dfrac{\sum_{k=1}^{N} I_{\{S(X_k \in E\}} I_{\{X_k \in \chi_{ij}\}}}{|E|}, & if\ i \neq 1 \\ \dfrac{\sum_{k=1}^{N} I_{\{S(X_k \in E\}} I_{\{(i,j) \in X_k\}}}{\sum_{k=1}^{N} I_{\{S(X_k \in E\}} I_{\{i \in X_k\}} - |E|}, & if\ i = 1 \end{cases} \tag{5.4}$$

Where $(i,j) \in X_k$ denotes the amount of times the transition from node i to j is included in solution X_k and $i \in X_k$ denotes the amount of times node i is included in solution X_k. We smooth $\hat{p}_{t,ij}$ using the smoothing factor α according to:

$$\hat{p}_{t,ij} = \alpha\tilde{p}_{t,ij} + (1-\alpha)\hat{p}_{t-1,ij} \tag{5.5}$$

The newly acquired probability matrix is used to generate new random samples X_i. To avoid premature convergence the MOO CEM uses a preset probability between 0.1 and 0.3 during each iteration t to invert the frequency diagrams for all decision variables. Since we use a transition probability matrix and not the histograms used in the MOO CEM we adapt the histogram inversion by changing the probabilities on the transition probability matrix with the same probability between 0.1 and 0.3 as follows:

$$\hat{p}_{0,ij} = \begin{cases} if\ i \neq j, & \dfrac{1-P_{i,j}}{n-2} \\ if\ i = j, & 0 \end{cases} \tag{5.6}$$

Note that n is the total number of nodes in the network including the depot. Since nodes are unable to transition to itself, the probabilities on the main diagonal remain zero. An example is given in equation 5.7.

$$P_{ij} = \begin{bmatrix} 0 & 0.1 & 0.6 & 0.3 \\ 0.8 & 0 & 0.1 & 0.1 \\ 0.2 & 0.2 & 0 & 0.6 \\ 0.8 & 0 & 0.2 & 0 \end{bmatrix} \xrightarrow{inverse} \hat{P}_{ij} = \begin{bmatrix} 0 & 0.45 & 0.2 & 0.35 \\ 0.1 & 0 & 0.45 & 0.45 \\ 0.4 & 0.4 & 0 & 0.2 \\ 0.2 & 0 & 0.8 & 0 \end{bmatrix} \tag{5.7}$$

After a given number of iterations, the elite vector E is ranked again using $th = 1$. After a certain amount of iterations the elite vector E is ranked one final time using $th = 0$ to obtain all the non-dominated solutions. The pseudo-code for the MOCO CEM is as follows:

MOCO CEM Algorithm

1. Set vector $E = \emptyset$, $t = 1$ and $k = 1$.
2. Set $p_{0,ij} = \begin{cases} \frac{1}{n-1}, & i \neq j \\ 0, & i = j \end{cases}$
3. Generate a sample of $X_1, \dots, X_N$ of routes using transition probability matrix $\hat{p}_{t-1}$ using the route generation algorithm
4. Calculate the performances $S(X_a), 1 \leq a \leq N$ by solving the MO-EVRP problem.
5. Rank the performance values using the Pareto-ranking algorithm (Goldberg, 1989) with $t_h = 2$ to obtain an updated elite vector E.
6. Calculate $\hat{p}_t$ using $\hat{p}_{t,ij} = \begin{cases} \dfrac{\sum_{k=1}^{N} I_{\{S(X_k \in E\}} I_{\{X_k \in \chi_{ij}\}}}{|E|}, & if\ i \neq 1 \\ \dfrac{\sum_{k=1}^{N} I_{\{S(X_k \in E\}} I_{\{(i,j) \in X_k\}}}{\sum_{k=1}^{N} I_{\{S(X_k \in E\}} I_{\{i \in X_k\}} - |E|}, & if\ i = 1 \end{cases}$
7. Smooth the transition matrix P_t using $P_t = \alpha\hat{P}_t + (1-\alpha)P_{t-1}$
8. If all $\sigma_{t,j} > \epsilon$ or if t is less than the allowable number of evaluations, increment t and reiterate from step 3.
9. Rank E using the Pareto-ranking algorithm (Goldberg, 1989) with $t_h = 1$
10. If k is less than the allowable number of loops increment k and return to Step 2.
11. Rank E using the Pareto-ranking algorithm (Goldberg, 1989) with $t_h = 0$ to obtain the final vector of efficient solutions.

Six

Validation

In this chapter we validate both the proposed MO-EVRP model and the MOCO CEM algorithm. We validate the model to show the validity and applicability of the model. The validation of the algorithm shows its effectiveness for the multi-objective VRP.

Validation of the EVRP

In order to validate the proposed model we compare the results with that of the pollution-routing problem (PRP) proposed by Bektaş and Laporte (2011) in a multitude of test settings. To compare both models we assume freeflow traffic conditions for all arcs and fix the speed of the vehicle depending on the road types that are considered in the NTM. For the EVRP model we use the data for 3 type of vehicles and 3 road types, see Table 6-1.

	Empty Weight	*Maximum Capacity*	*Traffic*	*Motorway 90 km/h*		*Rural 70 km/h*		*Urban 50 km/h*	
Type 4	6	6		0.160	0.194	0.150	0.193	0.170	0.226
			Freeflow	0.157	0.190	0.149	0.195	0.162	0.212
			Saturated	0.151	0.200	0.182	0.243	0.241	0.305
			Stop+Go	0.390	0.421	0.390	0.42	0.409	0.443
Type 6	10	12		0.195	0.283	0.201	0.310	0.251	0.392
			Freeflow	0.189	0.276	0.203	0.318	0.240	0.369
			Saturated	0.210	0.331	0.268	0.419	0.367	0.529
			Stop+Go	0.573	0.650	0.577	0.655	0.613	0.701
Type 8	14	26		0.219	0.345	0.225	0.385	0.284	0.491
			Freeflow	0.212	0.337	229	0.398	0.270	0.457
			Saturated	0.240	0.419	0.305	0.529	0.412	0.649
			Stop+Go	0.656	0.769	0.656	0.769	0.691	0.816

Table 6-1 Fuel Consumption Parameters (NTM, 2010)

For the PRP model we use the suggested parameters as seen in Demir et al. (2014), see Table 6-2. The curb weight is adjusted according to the weight classes used in the NTM methodology.

Notation	Description	Typical values
w	Curb weight in kg	6350
ξ	Fuel-to-air mass ratio	1
k	Engine friction factor	0.2
N	Engine Speed	33
V	Engine displacement	5
g	Gravitational constant	9.81
C_d	Coefficient of aerodynamic	0.7
ρ	Air density	1.2041
A	Frontal surface	3.912
C_r	Coefficient of rolling resistance	0.01
n_{tf}	Vehicle drive train efficiency	0.4
η	Efficiency parameter for diesel engines	0.9
κ	Heating value of a typical diesel fuel	44
ψ	Conversion factor	737
α	Arc-specific constant = $\tau + gsin\,\theta + gC_r \cos\theta$	
β	Vehicle-specific constant = $0.5C_d\rho A$	

Table 6-2: CMEM Parameters used in the PRP (Demir et al., 2014)

In order to compare both models we ran random generated test instances in which we differentiate the distance and maximum capacity of the vehicle. We differentiate 3 types of vehicles and 5 different distances. For each category we solve 10 different 12-customer VRP test instances where both the customer locations and the demand is random generated. The customer locations are random generated in a square of the given distance. The road type used in the EVRP model is adjusted according to the distance. The demand for each customer is random generated between 10% and 50% of the maximum capacity of the truck.

Distance (km)	*Road Type*	*Maximum Capacity (ton)*		
		6	12	26
50	Urban	100%	100%	100%
100	Rural	100%	100%	100%
500	Motorway	100%	100%	90%
1000	Motorway	70%	90%	90%
2000	Motorway	90%	80%	90%

Table 6-3: EVRP and PRP results comparison

The results as seen in Table 4 show that both models give comparable results in all different combinations of variables. Only for very long distances, 1000 and 2000 kilometer, we see that the optimal solutions between the EVRP and the PRP start to differ.

Validation of the MOCO CEM

In order to validate the proposed multi-objective combinatorial cross-entropy method for the MO-EVRP we propose the following steps:

- Compare the algorithm with the weighted sum method
- Compare the algorithm with an evolutionary algorithm

In order to compare the MOCO CEM with the classic weighted sum method we set-up a benchmark case. To compare the MOCO CEM with an evolutionary algorithm we compare the results from the MOCO CEM with the results from the evolutionary algorithm proposed by Jozefowiez et al. (2009) for the VRP with route balancing.

Weighted-Sum Method

The weighted sum method is considered one the most widely used classical multi-objective optimization methods (Deb, 2001). The weighted sum method scalarizes a set of objective functions into a single objective by giving a user-supplied weight to each of the objective functions.

After normalizing the objective functions, a composite objective function $F(x)$ can be formed by summing the weighted normalized objective functions and the MOOP is then given by the following single objective optimization function:

$$minimize\ = \sum_{i=1}^{k} w_i f_i(x) \qquad (6.1)$$
$$where\ w_i \geq 0, \forall i = 1, \dots, k\ and\ \sum_{i=1}^{k} w_i = 1$$

In order to normalize the objective functions different normalization schemes can be applied. In this case we normalize by the differences of optimal function values in the Nadir and Utopia points. This difference gives us the length of the intervals where the optimal objective functions vary within the Pareto optimal set. By first solving the single objective functions we can determine the Utopia and Nadir points respectively given by:

$$f_i^N = f_i(x^i)\ where\ f_i^U = argmin_x\{f_i(x)\} \qquad (6.2)$$
$$f_i^N = \max_{1 \leq j \leq k} (f_i(x^j)\ \ where\ \forall i = 1, \dots, k$$

Resulting in the following normalization values:

$$\theta_i = \frac{1}{f_i^N - f_i^U} \qquad (6.3)$$

Which gives the following normalized objective functions:

$$\bar{f_i} = \frac{f_i - f_i^U}{f_i^N - f_i^U} \qquad (6.4)$$

For a bi-objective optimization problem we can then shift the weight of importance from one objective to the other to investigate all possible trade-off scenarios and construct the Pareto-front. Notice that this method will only guarantee to find all solutions on the entire Pareto-optimal set when the solution space is convex. To compare the weighted-sum method with the MOCO CEM we conducted an experiment for the MO-VRP with 10 customers, see Figure 4. Here we assume a vehicle capacity of 26 tons and the following demand pattern:

$$q = \{0, 19529, 13469, 9649, 3681, 5628, 8379, 8748, 15983, 7780, 18746\} \qquad (6.5)$$

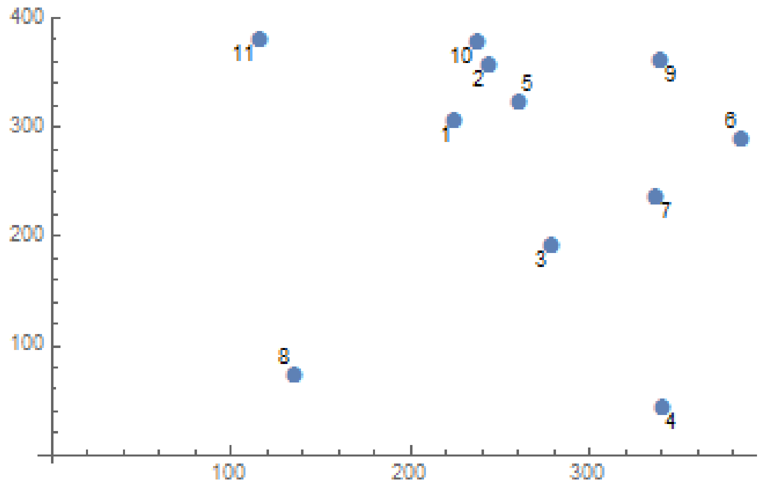

Figure 6-1 Weighted-sum example

To solve the example for both the weighted-sum method and the MOCO CEM we implement both methods within Mathematica. For implementations see chapter 9 and chapter 11.4 respectively. For the weighted-sum method we use incremental steps of 0.01, resulting in a total of 100 optimization problems. Both the weighted-sum method and the MOCO CEM generate the same Pareto front for this example, see Figure 6-2.

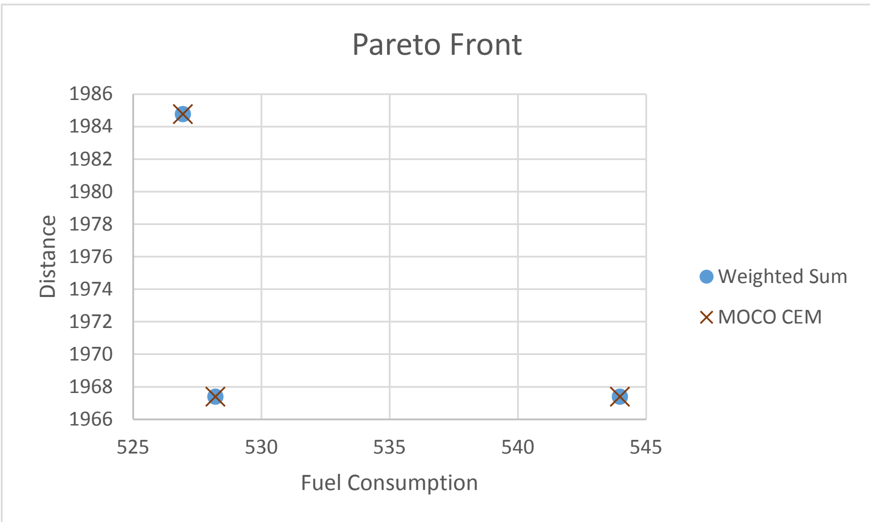

Figure 6-2: Results MO-EVRP and Weighted-Sum comparison

Evolutionary Algorithm

Here we present an analysis of the MOCO CEM when we apply it to a VRP model suggested by Jozefowiez et al. (2009) for the vehicle routing problem with route balancing. In their paper they consider the vehicle routing problem with route balancing (VRPRB) in which 2 objectives are optimized:

1. Minimization of the distance traveled by the vehicles
2. Minimization of the difference between the longest and shortest route length.

Jozefowiez et al. (2009) propose a multi-objective evolutionary algorithms (MOEA) to solve the VRPRB which includes two new mechanisms. An elitist diversification method which involves parent selection step of the evolutionary algorithm which makes it possible for solutions not in the current population or in the non-dominated archive to become parent. The second mechanism modifies the population update which implements an island model which allows the use of parallelization in order to speed up the search process.

The results from both the evolutionary algorithm and the MOCO CEM for the first instance of the Christofides, Mingozzi and Toth (1979) benchmark cases for 50 customers are given in Figure 6. Note that not the entire Pareto set is supplied in the paper by Jozefowiez et al. (2009).

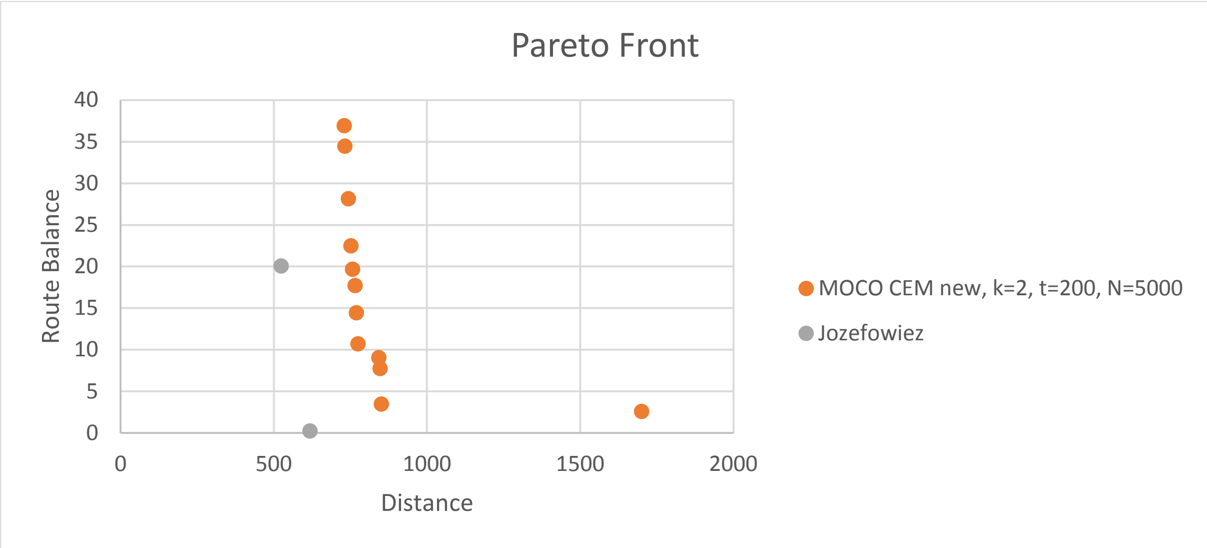

Figure 6-3: Results MOCO CEM and MOEA comparison

Seven

Results

The MO-EVRP model uses both load and traffic conditions on each arc to estimate the carbon emissions. In order to investigate the contribution of both factors we conduct several experiments in which we differentiate the variability of the load and the traffic parameters.

Effects of load

The MO-EVRP model uses the load in order to estimate the carbon emissions. To investigate the importance of the load in the trade-off between distance and carbon emissions in the MO-EVRP we conduct two sets of experiments. For these experiments the demand for each customer is generated using a truncated normal distribution. This allows us to differentiate the demand patterns for both sets of experiments. The first set of experiments uses a low variance of demand and the second set uses a high variance of demand. For both set of experiments we use the same 50 customer problem instance, see Figure 7-1.

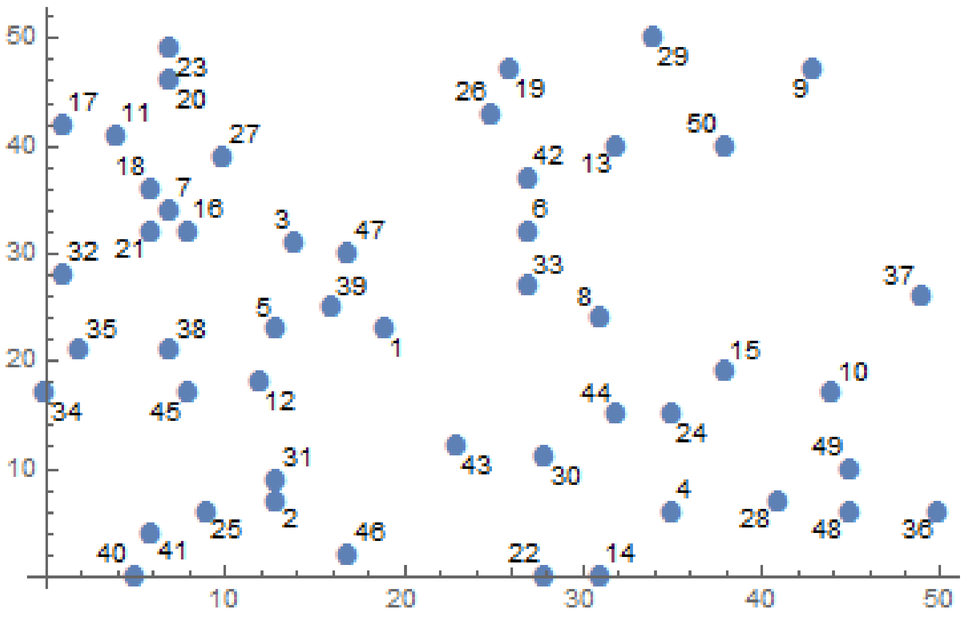

Figure 7-1: Problem instance effect of load

The type of road is assumed to be urban, traffic on all arcs is set to freeflow and the vehicle capacity is set to 12 ton. Following Table 2 this results in the following fuel consumption parameters:

Empty fuel consumption (l/km)	f_t^e	0.240
Full fuel consumption (l/km)	f_t^e	0.369

Table 0-1 Fuel consumption parameters

The demand pattern follows a truncated normal distribution. The sets of experiments both use a mean of $\mu = 1000$ and a standard deviation of $\sigma = 100$ and $\sigma = 500$ respectively. Both demand distributions are truncated using $X \in [1,2000]$ in order to prevent negative or zero demand values.

For the first set of experiments we generated 5 random instances. The results show that in these cases there is no trade-off between distance and fuel consumption and result in one optimal solution.

Instance	Distance	Fuel Consumption
1	523.83	181.499
2	512.32	179.231
3	534.32	183.421
4	523.21	180.109
5	498.23	178.324

Table 7-2: Results low demand variance

For the second set of experiments we also generated 5 random instances. Here the results show that there is indeed a trade-off between both objective functions. However the trade-off is relativity low with fuel savings between 0% and 1.3% depending on the instance.

Instance	Distance	Fuel Consumption	Instance	Distance	Fuel Consumption
1	518.02	148.647	5	562.92	164.756
	513.74	149.084		560.57	165.056
2	491.46	143.791		559.35	165.433
3	569.91	166.298		558.4	165.779
	568.38	167.455		555.93	166.618
4	532.25	157.509		552.27	166.886

Table 7-3: Results high demand variance

Effect of traffic

To measure the effect of congestion on optimal VRP solutions we first solve a small sample case, see Figure 7: Problem instance effect of load. In this sample we see the effects on optimal routing when we change the traffic parameter in just one of the arcs. In this sample we use free-flow for all the arcs and introduce a Stop+Go traffic parameters in the arc $x_{1,6}$. Note that the direction of the route changes when we minimize for CO_2-emissions.

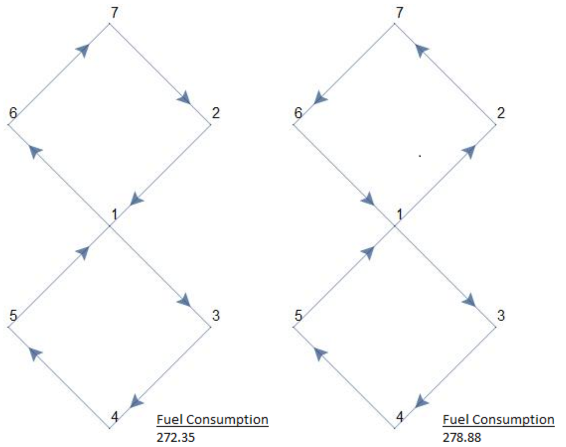

Figure 7-2: Problem instance effect of traffic

To measure the effects of traffic in combination with de variance in load we conducted an experiment on the same problem instance as for the effects of load, see Table 7-4. For this experiment we introduce a random generated arc-dependent traffic parameter. This parameter is an integer from 1 to 3 all with an equal chance describing Freeflow, Saturated or Stop+Go traffic conditions. The fuel consumption parameters are given by:

		Freeflow	Saturated	Stop+Go
Empty fuel consumption (l/km)	f_t^e	0.240	0.367	0.613
Full fuel consumption (l/km)	f_t^e	0.369	0.529	0.701

Table 7-4: Fuel consumption parameters with traffic

The results from this experiment are shown in Figure 9: Results of variance in traffic Here we can clearly see a trade-off between the distance and fuel consumption. Fuel savings of up to 28.3% can be achieved by increasing the distance by 23.6%. In this example it can be seen that after the distance of around 1500 kilometer the savings in fuel consumption are greatly reduced.

This example shows that minimizing for only fuel consumption would result in a substantial increase of total traveled distance. However using the multi-objective approach shows that there is a steep drop off in fuel savings after a certain point and shows where that point is located on the Pareto front.

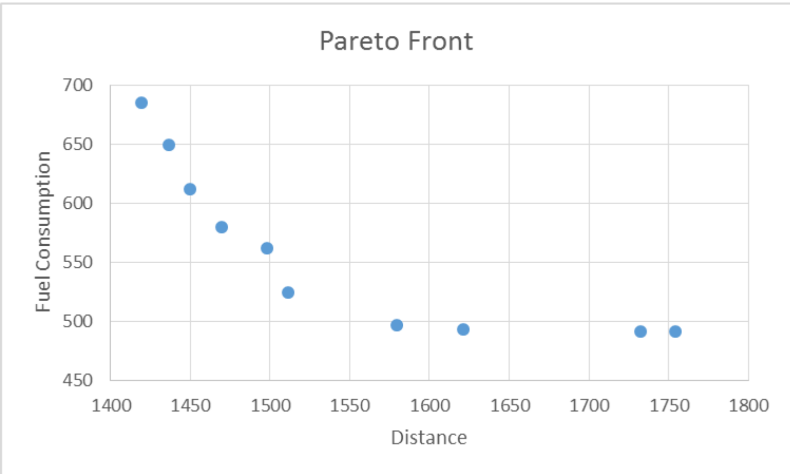

Figure 7-3: Results of variance in traffic

Eight

Conclusion and future research

In this study we investigated the trade-off between cost and CO_2 emissions in the vehicle routing problem. The main contributions of this study are the introduction of the multi-objective emission vehicle routing problem (MO-EVRP), the adaptation to the multi-objective combinatorial cross-entropy method (MOCO CEM) in order to solve this model and the insights into the behavior of the different factors that influence the trade-off between cost and CO_2 emissions.

Research Questions

The research question of this thesis is defined as "How can a company define the structure of the Pareto frontier to trade-off cost and carbon emissions when deciding vehicle routings?" which is answered by the following sub questions:

What is the most applicable way to include carbon emissions in the VRP?

The first step to include carbon emissions in the VRP is by accurately estimating these emissions. As the literature research showed there are many different methodologies to do so. However not every methodology includes every required factor that is needed in order to estimate carbon emissions. This thesis focusses on the contributions of load and traffic factors for emission estimation. These factors can be included in the design of the VRP and are not subject to the driver, such as vehicle speed. Resulting from these factors we proposed the emission vehicle routing problem based on the NTM methodology. Since the MO-EVRP is based on an activity-based estimation the required parameters are substantially less detailed then in the more common energy-based estimation models.

How can we solve the multi-objective VRP?

In this thesis we introduced an adaptation of the multi-objective cross-entropy method in order to solve the MO-EVRP. Comparison with the weighted-sum method and an evolutionary algorithm showed promising results. The biggest advantage is the applicability of the metaheuristic as it can be applied to all applications of the VRP with relatively easy adjustments.

How does our model compare to other models found in literature?

The first important difference between the model proposed in this study and other activity-based models found in literature (see Table 2-1) is that it takes into account both load and traffic related parameters. Including both factors in the model provides further detail on the behavior of optimal vehicle routings. Results show that in relation to energy-based estimation models the model presented in this study gives comparable results. Optimal routings obtained with the MO-EVRP are in many cases identical to those routings obtained with the PRP. Only for very large distances, 1000 kilometers and up, the model starts to differ from the obtained solutions with the PRP.

How can we interpret the obtained results regarding the trade-off between carbon emissions and cost?

Results show that when load is the only differentiating factor between distance and carbon emissions the trade-off is relatively low, with carbon emission savings up to 1.2% when customer demand diversity is high. This relationship between customer demand diversity and carbon emission savings is not easy to formulate mathematically. We do see that when the customer demand diversity is increased possible carbon emission savings also increase. When we also introduce the arc-dependent traffic parameter we see a high increase in possible carbon emission savings with possible savings up to 28.3%. This makes the proposed model especially interesting for companies that operate within urban and high density areas.

Future Research

One of the biggest challenges in this study was the processing performance of the implemented algorithm. The route generation algorithm requires high computing power, therefor it would be interesting to see the results of the performance of the algorithm with more processing power or more efficient ways to generate feasible routes for the VRP.

Further research for improvements for the cross-entropy method for the VRP can result in more efficient and faster algorithms. Since the algorithm can require a high amount of iterations, the inclusion of improvement methods within the elite vector, such as 2-opt, can be a promising addition to the algorithm.

Another promising areas that is left out in this study is the use of time-windows and time-dependent traffic parameters in the MO-EVRP. In many real life cases customers are restricted by time-windows and traffic is a very time-dependent factor. Traffic has a big impact on average vehicle speed which greatly impacts travel times. Furthermore a real case study can show the effectiveness and the usefulness of the model to the decision makers in the field of transportation.

Nine

References

Alon, G., Kroese, D. P., Raviv, T., & Rubinstein, R. Y. (2005). Application of the cross-entropy method to the buffer allocation problem in a simulation-based environment. *Annals of Operations Research*, 134(1), 137-151.

Baldacci, R., Christofides, N., & Mingozzi, A. (2008). An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming*, *115*(2), 351-385.

Bekker, J., & Aldrich, C. (2011). The cross-entropy method in multi-objective optimisation: An assessment. *European Journal of Operational Research*,*211*(1), 112-121.

Bektaş, T., & Laporte, G. (2011). The pollution-routing problem. *Transportation Research Part B: Methodological*, *45*(8), 1232-1250.

Barth, M., Younglove, T., & Scora, G. (2005). Development of a heavy-duty diesel modal emissions and fuel consumption model. California Partners for Advanced Transit and Highways (PATH).

Boulter, P. G., & MCCRAE, I. S. (2007). ARTEMIS: Assessment and Reliability of Transport Emission Models and Inventory Systems-final report. TRL *Published Project Report*.

Bowyer, D. P., Akcelik, R., & Biggs, D. C. (1985). Guide to fuel consumption analyses for urban traffic management (No. 32).

Chepuri, K., & Homem-De-Mello, T. (2005). Solving the vehicle routing problem with stochastic demands using the cross-entropy method. *Annals of Operations Research*, 134(1), 153-181.

Christofides, N., & Eilon, S. (1969). An algorithm for the vehicle-dispatching problem. *OR*, 309-318.

Christofides, N., Mingozzi, A., & Toth, P. (1981). Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. *Mathematical programming*, 20(1), 255-282.

Clarke, G. U., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations research*, *12*(4), 568-581.

Coello, C. C., Lamont, G. B., & Van Veldhuizen, D. A. (2007). *Evolutionary algorithms for solving multi-objective problems*. Springer.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem.*Management science*, *6*(1), 80-91.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms* (Vol. 2012). Chichester: John Wiley & Sons.

Demir, E., Bektaş, T., & Laporte, G. (2014). A review of recent research on green road freight transportation. *European Journal of Operational Research*, 237(3), 775-793.

Demir, E., Bektaş, T., & Laporte, G. (2014). The bi-objective Pollution-Routing Problem. *European Journal of Operational Research*, *232*(3), 464-478.

Dorigo, M. (1992). Optimization, learning and natural algorithms. *Ph. D. Thesis, Politecnico di Milano, Italy.*

Figliozzi, M. (2010). Vehicle routing problem for emissions minimization. *Transportation Research Record: Journal of the Transportation Research Board*, (2197), 1-7.

Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks*, *11*(2), 109-124.

Gillett, B. E., & Miller, L. R. (1974). A heuristic algorithm for the vehicle-dispatch problem. *Operations research*, *22*(2), 340-349.

Glover, F. (1989). Tabu search-part I. *ORSA Journal on computing*, 1(3), 190-206.

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning, 1989. *ISBN: 0-201-15767-5*.

Helvik, B. E., & Wittner, O. (2001). Using the cross-entropy method to guide/govern mobile agent's path finding in networks. In *Mobile Agents for Telecommunication Applications* (pp. 255-268). Springer Berlin Heidelberg.

Holland, J. H. (1975). Adaptation in natural and artificial system: an introduction with application to biology, control and artificial intelligence. *Ann Arbor, University of Michigan Press.*

Jozefowiez, N., Semet, F., & Talbi, E. G. (2009). An evolutionary algorithm for the vehicle routing problem with route balancing. European Journal of Operational Research, 195(3), 761-769.

Jabali, O., Woensel, T., & de Kok, A. G. (2012). Analysis of Travel Times and CO2 Emissions in Time-Dependent Vehicle Routing. *Production and Operations Management*, *21*(6), 1060-1074.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, 220(4598), 671-680.

Laporte, G. (2009). Fifty years of vehicle routing. *Transportation Science*, *43*(4), 408-416.

Margolin, L. (2005). On the convergence of the cross-entropy method. *Annals of Operations Research*, *134*(1), 201-214.

Ma, T. Y. (2011). A cross entropy multiagent learning algorithm for solving vehicle routing problems with time windows. In *Computational Logistics* (pp. 59-73). Springer Berlin Heidelberg.

Network for Transport and Environment Road 2010. Environmental data for international cargo transport.

Palmer, A. (2007). The development of an integrated routing and carbon dioxide emissions model for goods vehicles.

Rubinstein, R. (1999). The cross-entropy method for combinatorial and continuous optimization. *Methodology and computing in applied probability*, 1(2), 127-190.

Rubinstein, R. Y., & Kroese, D. P. (2004). The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning. Springer.

Schaffer, J. D. (1985). *Some experiments in machine learning using vector evaluated genetic algorithms*. Vanderbilt Univ., Nashville, TN (USA).

Scott, C., Urquhart, N., & Hart, E. (2010). Influence of topology and payload on CO2 optimised vehicle routing. In *Applications of Evolutionary Computation* (pp. 141-150). Springer Berlin Heidelberg.

Tavares, G., Zsigraiova, Z., Semiao, V., & da Graça Carvalho, M. (2008). A case study of fuel savings through optimisation of MSW transportation routes. *Management of Environmental Quality: An International Journal*, 19(4), 444-454.

Toth, P., & Vigo, D. (Eds.). (2001). *The vehicle routing problem*. Siam.

Urquhart, N., Hart, E., & Scott, C. (2010, July). Building low CO 2 solutions to the vehicle routing problem with Time Windows using an evolutionary algorithm. In *Evolutionary Computation (CEC), 2010 IEEE Congress on* (pp. 1-6). IEEE.

Xiao, Y., Zhao, Q., Kaku, I., & Xu, Y. (2012). Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & Operations Research*, 39(7), 1419-1431.

Ten

Implementations

MOCO CEM including Route Generation Algorithm

```
pts={{19,23},{13,7},{14,31},{35,6},{13,23},{27,32},{7,34},{31,24},{43,47},{44,17}
,{4,41},{12,18},{32,40},{31,0},{38,19},{8,32},{1,42},{6,36},{26,47},{7,46},{6,32}
,{28,0},{7,49},{35,15},{9,6},{25,43},{10,39},{41,7},{34,50},{28,11},{13,9},{1,28}
,{27,27},{0,17},{2,21},{50,6},{49,26},{7,21},{16,25},{5,0},{6,4},{27,37},{23,12},
{32,15},{8,17},{17,2},{17,30},{45,6},{45,10},{38,40}};
n=Length[pts];
c =12000;
q=Prepend[Round[RandomVariate[TruncatedDistribution[{0,2000},NormalDistribution[1
000,500]],(n-1)]],0];
ptsn=Range[n];
dmat=Round[Outer[EuclideanDistance,pts,pts,1],0.01];
tmat=RandomInteger[{1,3},{n,n}];
p=Table[1/(n-1),{i,n},{j,n}];
p=ReplacePart[p,{i_,i_}->0];
ev={};
th=2;
alpha=0.7;
t=1;
k=1;
fcempty={0.240,0.367,0.613};
fcfull={0.369,0.529,0.701};


distance[{i_,j_}]:=dmat[[i,j]];(*Distance between i and j*)
traffic[{i_,j_}]:=tmat[[i,j]];(*Traffic between i and j*)
tdistance[k_]:= Total[Map[distance,Partition[k,2,1]]];(*Total distance*)

payload[l:{__}]:=Module[{},
  arcs=Partition[l,2,1];
  routes=Split[l,#2=!=1&];
  m=0;
  tkm=0;
  counter=1;
  While[Length[arcs]>0,
   If[arcs[[1,1]]==1,m+=1;w=Total[q[[routes[[m]]]]]];
   tkm+=(fcempty[[traffic[arcs[[1]]]]]+(fcfull[[traffic[arcs[[1]]]]]-
fcempty[[traffic[arcs[[1]]]]])*w/c)*distance[arcs[[1]]];
   w=w-q[[arcs[[1,2]]]];
   arcs=Drop[arcs,1];
   counter++];
  tkm
  ]

RG[matrix_,demand_,cap_]:=Module[{},
  pn=matrix;
  pn[[All,1]]=0;
  cq=0;
  a=1;
  r=List[a];
  s=q;
  While[Total[s]>0,
   f=1-UnitStep[Table[cq+demand[[i]]-cap,{i,Length[demand]}]];
   pn[[a]]=pn[[a]]*f;
```

```
   If[Total[pn[[a]]]==0,b=1;cq=0,b=RandomChoice[pn[[a]]-
>ptsn];cq=cq+q[[b]];pn[[All,b]]=0];
   r=Append[r,b];
   s[[b]]=0;
   a=b;
   ];
  Append[Flatten[Sort[Split[r,#2=!=1&],#1[[2]]<#2[[2]]&]],1]
  ]

RGdepot[matrix_,demand_,cap_]:=Module[{},
  pn=matrix;
  cq=0;
  a=1;
  r=List[a];
  s=q;
  While[Total[s]>0,
   f=1-UnitStep[Table[cq+demand[[i]]-cap,{i,Length[demand]}]];
   pn[[a]]=pn[[a]]*f;
   b=RandomChoice[pn[[a]]->ptsn];
   If[b==1,cq=0,cq=cq+q[[b]];pn[[All,b]]=0];
   r=Append[r,b];
   s[[b]]=0;
   a=b;
   ];
  Append[Flatten[Sort[Split[r,#2=!=1&],#1[[2]]<#2[[2]]&]],1]
  ]
dominate[l:{{_,_,_}..}]:=Module[{a,b,ls,ln,dom},(*Pareto ranking*)
  ls=SortBy[Sort[l],Last][[All,1]];
  ln=l[[All,2;;3]];
  a=Sort[ln];
  b=SortBy[ln,Last];

dom=Map[Join[#,{Length[Intersection[Cases[{b},{vals___,#,___}:>vals],Cases[{a},{v
als___,#,___}:>vals]]]}]&,b];
  MapThread[Prepend,{dom,ls}]
  ]

pupdate[i_,j_]:=Module[{},
  If[i==1,Count[ev,{___,i,j,___},3]/(Count[ev,1,{3}]-
Length[ev]),Count[ev,{___,i,j,___},3]/Length[ev]]
  ]

While[k<=20,
 While[t<=100,
  samples=DeleteDuplicates@ParallelTable[RG[p,q,c],{x,(n^2)}];
  samples=Thread[{samples,Map[tdistance,samples],Map[payload,samples]}];
  samples=dominate[samples];
  samples=Sort[samples,#1[[4]]<#2[[4]]&];
  best=Cases[samples,{__,__,__,n_}/;n<=th];
  ev=First/@GatherBy[Join[ev,best],Take[#,1]&];
  ev=Drop[ev,None,{4}];
  ev=dominate[ev];
  ev=ev[[Ordering[ev[[All,4]]]]];
  ev=Cases[ev,{__,__,__,n_}/;n<=th];
  newp=Table[pupdate[i,j],{i,n},{j,n}];
  p=alpha*newp+(1-alpha)*p;
  invert=Random[];
  If[invert<= 0.3,p=ReplacePart[p,{i_,j_}:>((1-p[[i,j]])/(n-
2))];p=ReplacePart[p,{i_,i_}:>0]];
  t++;
  ];
 th=1;
 ev=Cases[ev,{__,__,__,n_}/;n<=th];
 k++;
 ]
th=0;
ev=Cases[ev,{__,__,__,n_}/;n<=th];
MatrixForm[ev]
```

VRP constraints and variables

```
VRP[demands_,vehicles_,capacity_]:=Module[{r0a,r0b,r1,r2,r3,r4,r5,r6,r7,r8,r9},

q=demands;
n=Length[demands];
m=vehicles;
Q=capacity;

xij=Array[x,{n,n}];
fij=Array[f,{n,n}];
fji=Transpose[fij];
variables=Flatten[{xij,fij}];

S=Drop[Subsets[Drop[Range[n],1]],1];

r0a=Table[xij[[i,i]]==0,{i,1,n}];
r0b=Table[fij[[i,i]]==0,{i,1,n}];
r1={Total[xij[[1]]]<=m}; (*at most m trucks leave the depot*)
r2={Total[xij[[1]]]==Total[xij[[All,1]]]}; (*all leaving trucks return*)
r3=Table[Total[xij[[i]]]==1,{i,2,n}];(*one arc enters each node*)
r4=Table[Total[xij[[All,j]]]==1,{j,2,n}];(*one arc leaves each arc*)
r5=Table[Total[fji[[i]]]-Total[fij[[i]]]==q[[i]],{i,2,n}];(*flow variable
constraint*)
r6=Flatten[Table[q[[j]]xij[[i,j]]<=fij[[i,j]]<=(Q-
q[[i]])xij[[i,j]],{i,1,n},{j,1,n}]];(*flow variable constraint*)
r7=Flatten[Table[fij[[i,j]]>=0,{i,1,n},{j,1,n}]];(*all flow variables bigger then
0*)
r8=Flatten[Table[0<=xij[[i,j]]<=1,{i,1,n},{j,1,n}]];(*all decision variables 0 or
1*)
r9=Table[Total[xij[[S[[i]],Delete[Range[n],Partition[S[[i]],1]]]],3]>=Ceiling[Tot
al[Part[q,S[[i]]]]/Q],{i,1,Length[S]}];(*capacity-cut constraint*)

constraints=Join[r0a,r0b,r1,r2,r3,r4,r5,r6,r7,r8,r9];
]
```

VRP minimization

```
variables={};
constraints={};

customers={{55,22},{8,139},{20,82},{3,55},{13,58},{194,49},{19,72},{67,192},{39,5
4},{142,186},{62,152}};
q=Prepend[RandomInteger[{0,6000},10],0];
Q=24000;
m=10;
dij=Round[Outer[EuclideanDistance,customers,customers,1],0.01];
fempty=0.219;
ffull=0.345;
fclcu=(ffull-fempty)/Q;

ListPlot[Thread@Labeled[customers,Range[Length[customers]]],AxesOrigin-
>{0,0},PlotMarkers->{Automatic,Medium}]

VRP[q,m,Q]

objdistance=Total[xij*dij,3];
objntm=Total[dij*(xij*fempty+fij*fclcu),3];

q
solution1=NMinimize[{objdistance,constraints,Flatten[variables]∈
Integers},variables];
solution1[[1]]
solution1xij=xij/.solution1[[2]];
solution1fij=fij/.solution1[[2]];
```

```
transitions1=Sort[Drop[Position[MatrixForm@solution1xij,1],0,1]];
Graph[DirectedEdge@@@transitions1,VertexCoordinates-
>customers[[DeleteDuplicates[Flatten[transitions1]]]],VertexLabels-
>"Name",EdgeShapeFunction->"ShortFilledArrow"]
```

VRP Weighted-sum

```
variables={};
constraints={};

customers={{226,306},{245,356},{279,192},{341,42},{261,322},{385,288},{338,236},{
136,73},{340,361},{238,377},{117,379}};
q=Prepend[RandomInteger[{500,24000},10],0];
Q=24000;
m=10;
dij=Round[Outer[EuclideanDistance,customers,customers,1],0.01];
fempty=0.219;
ffull=0.345;
fclcu=(ffull-fempty)/Q;
f1u=1967.39;
f1n=1984.77;
f2u=526.94;
f2n=543.97;

ListPlot[Thread@Labeled[customers,Range[11]],AxesOrigin->{0,0},PlotMarkers-
>{Automatic,Medium}]

VRP[q,m,Q]

objdistance=Total[xij*dij,3];
objntm=Total[dij*(xij*fempty+fij*fclcu),3];

q
solution1=NMinimize[{objdistance,constraints,Flatten[variables]∈
Integers},variables];
solution1[[1]]
solution1xij=xij/.solution1[[2]];
solution1fij=fij/.solution1[[2]];
Total[dij*(solution1xij*fempty+solution1fij*fclcu),3]

transitions1=Sort[Drop[Position[MatrixForm@solution1xij,1],0,1]];
Graph[DirectedEdge@@@transitions1,VertexCoordinates-
>customers[[DeleteDuplicates[Flatten[transitions1]]]],VertexLabels-
>"Name",EdgeShapeFunction->"ShortFilledArrow"]

solution2=NMinimize[{objntm,constraints,Flatten[variables]∈ Integers},variables];
solution2xij=xij/.solution2[[2]];
solution2fij=fij/.solution2[[2]];
Total[solution2xij*dij,3]
solution2[[1]]

transitions2=Sort[Drop[Position[MatrixForm@solution2xij,1],0,1]];
Graph[DirectedEdge@@@transitions2,VertexCoordinates-
>customers[[DeleteDuplicates[Flatten[transitions2]]]],VertexLabels-
>"Name",EdgeShapeFunction->"ShortFilledArrow"]


Do[
 w1=i;
 w2=1-i;
 objwsum=w1*(Total[xij*dij,3]-f1u)/(f1n-
f1u)+w2*(Total[dij*(xij*fempty+fij*fclcu),3]-f2u)/(f2n-f2u);
 min=NMinimize[{objwsum,constraints,Flatten[variables]∈ Integers},variables];
 minxij=xij/.min[[2]];
 minfij=fij/.min[[2]];
 Print[Total[minxij*dij,3]&&Total[dij*(minxij*fempty+minfij*fclcu),3]];
 ,{i,0,1,0.01}]
```