UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

TESIS

SOLVING A GREEN LOGISTICS BI-LEVEL BI-OBJECTIVE PROBLEM

POR

CARMEN SAYURI MALDONADO PINTO

SOMETIDA PARA OBTENER EL GRADO DE

MAESTRÍA EN CIENCIAS CON ORIENTACIÓN EN MATEMÁTICAS

ENERO, 2017

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS
CENTRO DE INVESTIGACIÓN EN CIENCIAS FÍSICO-MATEMÁTICAS

TESIS

SOLVING A GREEN LOGISTICS BI-LEVEL BI-OBJECTIVE PROBLEM

POR

CARMEN SAYURI MALDONADO PINTO

SOMETIDA PARA OBTENER EL GRADO DE

MAESTRÍA EN CIENCIAS CON ORIENTACIÓN EN MATEMÁTICAS

San Nicolás de los Garza, N.L. JANUARY, 2017

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS
CENTRO DE INVESTIGACIÓN EN CIENCIAS FÍSICO-MATEMÁTICAS

Los miembros del comité de tesis de la subdirección de posgrado de la Facultad de Ciencias Físico-Matemáticas, recomendamos que la tesis "Solving a green logistics bi-level bi-objective problem" realizada por la Lic. Carmen Sayuri Maldonado Pinto, con número de matrícula 1534979, sea aceptada para su defensa para opción al grado de Maestría en Ciencias con Orientación en Matemáticas.

El Comité de Tesis

Dr. José Fernando Camacho Vallejo
Asesor

Dr. José Luis González Velarde
Co-asesor

Dr. Omar Jorge Ibarra Rojas

Vo. Bo.

Dr. José Fernando Camacho Vallejo
Coordinador del Posgrado en Ciencias con Orientación en Matemáticas

San Nicolás de los Garza, N.L. JANUARY, 2017

Abstract

by Carmen Sayuri Maldonado Pinto

The situation here addressed is modelled as a bi-level programming problem with multiple objectives in the upper level and a single objective in the lower level. In this problem, a company (hereafter the leader) distribute a commodity over a selected subset of customers; while a manufacturer (hereafter the follower) will fabricate the commodities demanded by the selected customers. The leader has two objectives: the maximization of the profit gained by the distribution process and the minimization of CO_2 emissions. The latter is important due to the regulations imposed by the government. It is clear that exists a compromise between both objectives, since the maximization of profit will attempt to include as much customers for being served as possible. Then, largest routes will be needed causing more CO_2 emissions.

For analyzing the problem, the single-commodity case is studied first. Under this assumption, the problem can be reduced into a single-level one. Hence, a tabu search algorithm for solving the aforementioned case is proposed. The tabu search is designed for solving two single-level simplifications of the problem: a mono-objective problem and the bi-objective one. After that, the multi-commodity bi-level case is studied and the respective adaptation of the tabu search is made. Then, a co-evolutionary algorithm is designed for obtaining good quality bi-level feasible solutions.

The co-evolutionary approach is related with having two separated populations, one for each leader's objective. Then, the solutions will evolve in each population and an interchange of information is made through the process. In other words, a swap between the best solutions from both populations in each generation is conducted. By doing this, the algorithm intends to find efficient solutions. The evolution performed in each population is done through a Biased Random Keys Genetic Algorithm(BRKGA).

Furthermore, a path relinking algorithm is adapted in order to find the Pareto frontier for the bi-level bi-objective multi-commodity problem, in which the no dominated solutions of the tabu search and the co-evolutionary algorithms are used to initialize this procedure. Numerical experimentation showed the efficiency of the proposed methods for finding good quality solutions (for the mono-objective case) and for reaching a good approximation of the Pareto front (for the bi-objective cases) in reasonable computational time.

Contents

Abstract **iii**

1 Introduction **1**
1.1 Problem's description 1
1.2 Motivation 1
1.3 Objectives 3
1.4 Methodology 3
1.5 Thesis structure 4

2 Literature review **6**
2.1 Bi-level programming 6
2.2 Multi-objective programming 7
2.3 Green logistics 7

3 Bilevel bi-objective multi-commodity problem **12**
3.1 Problem statement 12
3.2 Mathematical model 13
3.3 Analizing simplifications of the problem 17
3.3.1 Mono-objective single-commodity problem 21
3.3.1.1 Mathematical model 21
3.3.2 Bi-objective single-commodity problem 22
3.3.2.1 Mathematical model 22

4 Proposed algorithms **24**
4.1 A tabu search for the mono-objective single-commodity problem . . . 25
4.2 A tabu search for the bi-objective single-commodity problem 27
4.3 A tabu search algorithm for the bi-level bi-objective multi-commodity problem 28
4.4 A co-evolutionary algorithm based on a biased random keys for the bi-level bi-objective multi-commodity problem 29
4.5 Using path relinking improving the Pareto frontier of the bi-level bi-objective multi-commodity problem 31
4.6 Performance measures for bi objective problems 33

5 Computational experimentation **36**
5.1 Description of the instances 36
5.2 Results 37
5.2.1 Mono-objective single-commodity 37
5.2.1.1 Solving with an optimizer (CPLEX) 38
5.2.1.2 Solving with a tabu search 38
5.2.2 Bi-objective single-commodity 40
5.2.3 Bi-level bi-objective multi-commodity problem 41

5.2.3.1 A tabu search algorithm for a bi-level bi-objective multi-commodity problem . . . 41
5.2.3.2 A co-evolutionary algorithm for a bi-level bi-objective multi-commodity problem . . . 42
5.2.3.3 Path relinking algorithm for a bi-level bi-objective multi-commodity problem . . . 43

6 Conclusions **45**
6.1 Future research . . . 46

A Results obtained for the comparison between mono-objective and bi-objective single-commodity problem **47**

B Illustrations of Tabu Search for the bi-level bi-objective multi-commodity problem **52**

C Graphics of Co-evolutionary algorithm based on a random keys for the bi-level bi-objective multi-commodity problem **57**

D Analyzing ND solutions lists for the bi-level bi-objective multi-commodity problem **62**

Bibliography **67**

Chapter 1

Introduction

In this chapter the description of the problem is presented. After that, the main motivation of the thesis and the importance of taking care of the environment is exposed.Then, the objectives of the thesis are described and the methodology used to elaborate this research is detailed. Finally the structure of this thesis is shown.

1.1 Problem's description

The problem here studied considers a situation in which two companies interact with each other in a hierarchical way within a supply chain. The first one acquires and distributes different types of commodities over a selected subset of customers; while the other company manufactures the commodities demanded by the selected customers. In this problem, it is assumed that the distributing company decides the routes for satisfying the selected subset of customers such that the profit is maximized. Moreover, due to the regulations imposed by the industry or the government, the routes have to pollute as less as possible. In order to decide the routing phase, an heterogeneous fleet of vehicles is available. It is natural to consider that each type of vehicle has different rates of CO_2 emissions. From the above mentioned, it is clear that the second objective is to minimize those CO_2 emissions.

On the other hand, once the subset of customers are selected, the distributing company has to acquire the commodities needed to cover all the accumulated demand that corresponds to those specific customers. In this case, the manufacturer company has different capacitated facilities to produce the requested demand. Also, this company has to decide the amount of commodities that will be manufactured at each facility. Then, the commodities will be shipped from the facilities to a single depot. The objective of the manufacturer company is to minimize the production and shipping costs. Furthermore, in each facility exists a pollution rate associated with each manufactured commodity and a maximum pollution rate allowed.

1.2 Motivation

The CO_2 in the atmosphere is increasing at an alarming rate since the Industrial Revolution. This is caused by the boom of factories and vehicles impulsed by fossil fuels.

Nowadays, the active carbon reservoirs need hundreds of years to be absorbed by the biosphere and the oceans. Considering this fact, most of the atmospheric carbon reservoir is produced by the industry and motor vehicles in (Forster et al., 2007), where fossil fuel is burned.

Due to the concerns generated by the climate change in our environment because of the pollution, and the importance of taking consciousness about the sources of this pollution; many government regulations and methodologies for calculating emission factors have surged. In Mexico, many regulations have been decreed over the last 40 years. In 1971 the first federal law was promulgated in the Diario Oficial de la Federación (1971), called *Ley Federal para prevenir y controlar el medio ambiente*, where some elements emphasize about the control of the emissions. After that, other laws were declared, one of the most important was published in the Diario Oficial de la Federación (1988a), called *Ley General del Equilibrio Ecológico y la Protección del Medio Ambiente (LGEEPA)* , where in the chapter I and II the control and prevention of the pollution in the atmosphere is established. Due to this law, a regulation has surged (*Reglamento en materia de prevención y control de la contaminación atmosférica*), where the technical procedures for pollutant emission sources are defined in the Diario Oficial de la Federación (1988b). Furthermore, licenses and certificates are created in order to manage and control the industrial activities, as *Licencia Ambiental Única (LAU)* and *Cédula de Operación Anual (COA)*.

Nowadays, there are normatives that established the maximum emission allowed for the industry and the vehicles. For example, the NOM-043-ECOL-1993 in the Diario Oficial de la Federación (1993) establishes the maximum emission levels allowed in the atmosphere by the factories, where they determined two zones: the critical areas and the rest of the country. They presented a table for the two zones, with the emanation of gases and the maximum levels; the NOM-041-ECOL-1999 in the Diario Oficial de la Federación (1999a) establishes the maximum emission levels allowed of pollutants from the tailpipe by motor vehicles in circulation that use gasoline as fuel, where they set up the maximum emissions depending of the year and model of the vehicle; and the NOM-042-ECOL-1999 in the Diario Oficial de la Federación (1999b) establishes the maximum emission levels allowed of unburned hydrocarbons, carbon monoxide, nitrogen oxides by the vehicles, where they fixed the maximum emissions depending of the type vehicle and its weight.

Moreover, environmental programs have been implemented in order to reduce the CO_2 reservoirs. The most popular is the one named*Hoy No Circula*, in which the circulation of the vehicles is restricted once a week depending on the last number of the license plate. These laws, regulations and programs have controlled the emissions and the pollution in our country. Furthermore, the amount of laws have been increasing over the years in order to improve the environment and the quality of our life.

Due to all the laws and regulations that have been decreed and the environmental awareness that has been growing, we decided to study a problem that considers the CO_2 emissions produced by the vehicles and by the manufacturing facilities. Also another aim is to minimize the emissions by this two sources, but without ignore the industrial cost. Due to we take in account the emissions and the profit, this problem is attacked as a bi-objective problem.

1.3 Objectives

In this section, the aims of this thesis are presented.

This study aims to analyze and solve the bi-level bi-objective multi-commodity problem here presented. Some reductions of the problem and its particularly study for each problem are presented. The reductions can be seen as simplifications of the original problem: the mono-objective single-commodity problem and the bi-objective single commodity problem

In the mono-objective single-commodity problem, in order to solve this problem an optimizer is tested. Due to the excessive computational time that the optimizer (CPLEX) requires to find an optimal solution, a tabu search algorithm is proposed to solve it.

Additionally, a tabu search algorithm for a multi-objective problem is adapted to solve the bi-objective single-commodity problem. Then, this tabu search algorithm is adjusted for solving the bi-level bi-objective multi-commodity problem.

Furthermore, a co-evolutionary algorithm based on biased random keys is developed for solving the last mentioned problem. These two algorithms are used to obtain the Pareto frontier (independently). The results obtained from both algorithms are joined and improved with a path relinking procedure. The path relinking algorithm is adjusted for the bi-level bi-objective multi-commodity problem in order to obtain the Pareto frontier.

1.4 Methodology

The implemented methodology is provided in this section. For completing this thesis we followed the next steps:

1. Literature review of bi-level programming
2. Literature review of bi-objectives programming
3. Literature review of environmental problems
4. Literature review of green logistics and supply chain
5. Literature review of bi-objective environmental problems
6. Literature review of bi-level environmental problems
7. Analyze the properties of the problem
8. Examine the mono-objective single-commodity problem
9. Examine the bi-objective single-commodity problem
10. Expose the thesis progress at the 1st International Workshop on Bi-level Programming (IWOBIP'16)

11. A tabu search algorithm was developed for the simplifications of the problem and for the bi-level bi-objective problem, during a research stay at Auburn University, in Auburn, Alabama with professor Alice E. Smith.
12. Implement the tabu search algorithm for the mono-objective single-commodity problem
13. Implement the tabu search algorithm for the bi-objective single-commodity problem
14. Implement the tabu search algorithm for the bi-level bi-objective multi-commodity problem
15. Implement a co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem.
16. Generate a battery of instances for validating and measuring the proposed methodologies
17. Interpret the results of the tabu search algorithm for the mono-objective single-commodity problem
18. Analyze and present the results of the tabu search algorithm for the bi-objective single-commodity problem
19. Analyze and expose the results of the tabu search algorithm for the bi-level bi-objective multi-commodity problem
20. Obtain and analyze the results of the co-evolutionary algorithm for the bi-level bi-objective problem.
21. Compare the results of the mono-objective problems
22. Compare the results of the bi-level bi-objective multi-commodity problem
23. Implement a path relinking algorithm for the bi-level bi-objective multi-commodity problem.
24. Analyze and present the results of the path relinking algorithm for the bi-level bi-objective multi-commodity problem
25. Illustrate and establish the Pareto frontier of the bi-level bi-objective multi-commodity problem.

1.5 Thesis structure

In this Chapter, the problem under study was described and explained. Furthermore, the motivation of this problem was exposed and the aims were presented to clarify the purpose of this research. Finally, the methodology used to accomplish this thesis was written.

A literature review of bi-level programming and multi-objective programming are realized in Chapter 2. Furthermore, a background of logistic problems that take into consideration the environment is done. Similarly, multi-objective and bi-level problems which considered the dioxide carbon emissions in the objective functions are researched and detailed, specifically in logistics and supply chains.

In Chapter 3, the problem is detailed. Hence the bi-level bi-objective multi-commodity problem is explained. Then, the sets, parameters and variables of the problem are presented. After that, the mathematical model is formulated and explained. Moreover, two simplifications of the problem are done. The problem is divided in three cases: mono-objective single-commodity, bi-objective single-commodity and the original problem.

In Chapter 4, the proposed algorithms are written in detail. The first algorithm described is a tabu search, that is used for solving the mono-objective single-commodity problem. Secondly, a tabu search algorithm for solving the multi-objective problem is adapted in order to solve the bi-objective single-commodity. Then, the tabu search algorithm for multi-objective problem is adjusted for the bi-level bi-objective problem. Later, a co-evolutionary algorithm based on a biased random keys is developed to solve the bi-level bi-objective problem. After that, a path relinking algorithm using the no dominated solutions obtained by the two algorithms above is presented in order to improve the Pareto frontier. Finally, the measures used to evaluate the performance of the Pareto frontier obtained by the algorithms are described.

In Chapter 5 the description of the battery instances is presented, in order to analyze the bi-level bi-objective problem and the algorithms mentioned above. Thereafter, an analysis of the obtained results by the tabu search algorithm for solving the mono-objective single-commodity problem are presented. Then, the obtained results of the tabu search algorithm designed for solving the mono-objective single-commodity problem are shown and compared with the develop of the above algorithm. After that, the results of the tabu search and the co-evolutionary algorithm are given and the results are presented, in order to analyze the Pareto frontier. Finally, the results of the path relinking algorithm is given and the Pareto frontier is analyzed.

Finally, in Chapter 6 the conclusions of the analysis obtained by the experimentation is given and possible extensions for future research are presented.

Chapter 2

Literature review

In this chapter a description of bi-level and multi-objective programming considered in this thesis is given. Hence, a literature review regarding green logistic problems is presented. Furthermore, some papers focused on the environment that are closely related with the problem are mentioned and described.

2.1 Bi-level programming

The first formulation of a bi-level programming problem came out in 1934 by Heinrich von Stackelberg, where a hierarchical model was presented in order to describe a problem related with the market economy.

In a bi-level programming problem exists two independent decision makers, where the leader controls the decision variables $y \in Y \subseteq \mathbb{R}^n$, and the follower controls the decision variables $x \in X \subseteq \mathbb{R}^m$. It is a hierarchical problem because a part of the constraints are delimited by a second optimization problem (the follower's problem). The leader will choose y first minimizing his objective function $F(x, y)$, where $F : X \times Y \to \mathbb{R}$. Then, the follower will react with x fixed, minimizing his objective function $f(x, y)$, where $f : X \times Y \to \mathbb{R}$. As in (Dempe, 2002) the follower's problem will be presented first:

$$\min_x \{f(x, y) : g(x, y) \leq 0, h(x, y) = 0\} \tag{2.1}$$

where $g : X \times Y \to \mathbb{R}^p$, $h : X \times Y \to \mathbb{R}^q$. The solution set of the follower's problem is denoted as $\Psi(y)$ and the elements of $\Psi(y)$ are denoted as $x(y)$. After that, the aim of the bi-level problems is to select a vector y in order to solve the lower level problem. This selection of y must satisfy $G(x(y), y) \leq 0, H(x(y), y) = 0$, where $G : X \times Y \to \mathbb{R}^k, H : X \times Y \to \mathbb{R}^l$. Additionally, a bi-level programming problem is defined as:

$$\text{“}\min_y\text{”} \{F(x(y), y) : G(x(y), y) \leq 0, H(x(y), y) = 0, x(y) \in \Psi(y)\} \tag{2.2}$$

The quotation marks are use in case of non-uniquely lower level optimal solutions. If the optimal solution is unique the quotation marks could be removed of the notation.

In the case when a solution is non-unique, there are two common approaches: *optimistic* and *pessimistic*. In the optimistic approach, the leader supposes that the follower will collaborate with the objective function. Bearing this in mind, the follower will select a solution of $\Psi(y)$, which is the best solution with respect to the leader's objective function. On the other hand, in the pessimistic approach the leader cannot influence the decision of the follower. That means, it is assumed that the follower will select the worst solution with respect to the leader's objective function. Therefore, the leader has to consider the impact of the follower's decisions. The previously equations (2.1) and (2.2) used above are part of (Dempe, 2002) and (Kalashnikov et al., 2015).

2.2 Multi-objective programming

The first reference of problems with multiple objectives is attributed to Pareto (1896). The principal concepts and definitions of multi-objective problems, that are presented below were obtained from (Sawaragi, Nakayama, and Tanino, 1985). Furthermore, a remarkable characteristic about the multi-objective problems is that most of the time the objectives have conflict with each other.

For each objective in multi-objective problems, there is an objective function $f_i : X \rightarrow \mathbb{R}$, where $X \subseteq \mathbb{R}^n$. Taking this into consideration, the multi-objective formulation is presented below.

$$\min f(x) = (f_1(x), f_2(x), f_3(x), \cdots, f_p(x)) \tag{2.3}$$

$$g_j(x) \leq 0 \qquad j = 1, \cdots, m \tag{2.4}$$

where the equation (2.3), called a vector optimization, represents the p objective functions of the problem; and the equation (2.4) represents the constraints.

Additionally, obtaining a unique optimal solution is almost impossible when it involves multi-objective optimization problems. Because, solving the problem lead to a set of efficient solutions. A brief definition of some important concepts is presented next:

- Preference order: when a relation representing by the decision maker's preference becomes an order
- Efficient solution: a $f(x)$ is said to be efficient if there is not a $f(x')$ preferred to $f(x)$.
- Pareto optimal solution: It is a solution $\tilde{x} \in X$ of the problem if there is not $x \in X$ s.t. $f(x) \leq f(\tilde{x})$ (in a minimization problem) for at least one objective.
- Weak pareto optimal solution: It is a solution point $\tilde{x} \in X$ of the problem if there is not $x \in X$ s.t. $f(x) < f(\tilde{x})$ (in a minimization problem) for at least one objective.

As we have already mentioned, in a multi-objective problem is difficult select a unique solution, hence it is necessarily to provide a set of pareto optimal solutions and/or weak pareto optimal solutions. Thus, the principal aim of solving a multi-objective problem is to find or approximate that solution set.

2.3 Green logistics

Diverse studies have emerged expressing their worries for the climate impact generated by the increase of carbon dioxide in the atmosphere, hereby some of the primary causes of this increase are the variations of volcanic aerosols and possibly solar luminosity. Therefore, processes that influence the sensitivity of climate models are examined in (Hansen et al., 1981), varying the values of humidity, cloud altitude, snow and vegetation albedo feedback.

Furthermore, studies related to the pollution in the ocean and land have been done see (Arora et al., 2011). Also, an historical simulation about the cumulative land plus ocean carbon uptake is compared with their observation based estimates, considering the temperature. Therefore, they considered atmospheric CO_2 concentration of greenhouse gases and aerosols.

Additionally, a framework for quantified eco-efficiency analysis is done in (Huppes and Ishikawa, 2005), where they defined the eco-efficiency as a tool for sustainability analysis, pointing out an empirical relation in economic activities between environmental cost or value and environmental impact. Besides, they highlight four types of eco-efficiency: the environmental productivity, environmental intensity of production, environmental improvement cost and environmental cost-effectiveness. On the other hand, a frame to evaluate the sustainability of operations in the manufacturing sector is proposed in (Labuschagne, Brenta, and Erck, 2005), where they summarize four issues: Global Reporting Initiative (GRI), United Nations Commission on Sustainable Development, Sustainability Metrics of the Institution of Chemical Engineers and Wuppertal Sustainability Indicators, in order to assess the performance of the manufacturing sector.

In (Szekely and Knirsch, 2005) is mentioned an analysis of the sustainable performance of the companies mentioned above, where the best available metrics to quantify sustainability of twenty major German companies are examined, as Allianz, BMW, Daimler Chrysler, Deutsche Bank, and Volkswagen. Among some of the indicators considered are: economic, environmental and social. In order to evaluate the companies, many approaches have to be done such as surveys, award schemes, benchmarking, sustainability indexes, and accreditation processes.

Similarly, sustainable operations are considered in (Gimenez, Sierra, and Rodon, 2012), and five important contributions had been obtained to analyze the economic, environmental and social indicators. The contributions are: (i) the consideration of the impact in the environment and social programs at the same time; (ii) the impact on the triple bottom line (environmental,economic and social performance), (iii) the relation between the impact of external and internal programs, (iv) the examination of individual plants, and (v) analysis the role of a supply chain collaboration.

Different research areas have appeared in order to confront these environmental and sustainable problems; this is the case of green logistics, in which, information is collected in order to integrate the environmental issues into logistics. In (Dekker, Bloemhof, and Mallidis, 2012), they presented possible developments, focusing on the design, planning and control in a supply chain. Hence, they showed many areas where environmental aspects could be considered as transportation, products and inventories, facilities (as warehouses, ports and terminals), supply and transport chain design, product recovery and closed loop supply chains and operational control of supply and transport chains.

In (Sbihi and Eglese, 2010), they involved wider environmental and social considerations in problems as dynamic lot-sizing, joint and separate set-up cost model, waste management problems, household waste collection, vehicle routing, among others.

Furthermore, in (Lai and Wong, 2012), important contributions in the management of logistic chains have been done, where they consider the environment by examining the effects of environmental regulatory pressure on the Green Logistic Management. Furthermore, they identified its components, related the management with the performance of environment and operations. Finally, they distinguished the antecedents that motivated the use of green logistics management. In (Ubeda, Arcelus, and Faulin, 2011) a case study related to the food distribution sector in Spain (Eroski), is proposed. They showed how it can be reduce the environmental impact in the practice, while simultaneously they improved the efficiency objectives.

In (Linton, Klassen, and Jayaraman, 2007) a background has been provided to understand sustainable supply chains. They summarize it in some subjects as sustainability, the interaction between sustainability and supply chains. Further, with the aim of developing an extension of an environmental supply chain, the environmental factors have been investigated. A description of the additional challenges of this extension is presented in (Beamon, 1999). They focused in solid and hazardous waste, natural resource use, water and air pollution, public pressure, environmental legislation, environmental management standards, the traditional and extended supply chain, and traditional supply chain performance measures, in order to a better understand of the environmental supply chain.

One of the main features of a green supply chain is taking into consideration the $C0_2$ emissions as in (Li et al., 2008), where the relationship between CO_2 emissions and operation cost-income ratio is analyzed in the location of distribution centers. Also, they did a case of study with the crude oil, where they concluded that if the price of the crude oil increase the carbon emission will decrease, and if more distribution centers are opened the carbon emission could decrease in some degree.

As well, in (Diabat and Simchi-Levi, 2009) the environmental impact of CO_2 emissions is considered and a novel model for a green supply chain management is presented, this model integrates the management and environmental impact into the supply chain. Furthermore, a vehicle routing problem is solved in (Erdoğan and Miller-Hooks, 2012), where the environmental effects have been considered, by using alternative fuel vehicles.

Below we present some researches of multi-objective problems with environmental issues.

We have already mentioned many cases where the environment aspects are considered, but if we take into account at the same time other objectives like economics or social, we will have a problem with multiple objectives. For this kind of problems, in the literature are considered several researches with multiple objectives, that are described below.

In (Harris, Mumford, and Naim, 2009) an uncapacitated facility location problem is studied. They simultaneously analyzed the cost, the environmental impact and the uncovered demand. In order to solve this problem, they implemented an evolutionary multi-objective algorithm (NSGA II). Moreover, in (Harris, Mumford, and Naim, 2014) a capacitated facility location-allocation problem is presented, where the objectives are the financial cost and the environmental impact. They applied a simple evolutionary algorithm for multi-objective optimization (SEAMO2) to obtain the set of efficient solutions.

A supplier selection and order allocation problem is shown in (Kannan et al., 2013), in which, they considered suppliers' environmental performance. They integrated a fuzzy analytic hierarchy process (AHP), fuzzy Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), and fuzzy Multi-objective linear programming (MOLP) to solve supplier selection and order allocation. This study focused in a Iranian automobile manufacturing company in order to establish a systematic approach for meeting green supplier selection. A partner selection problem is introduced in (Yeh and Chuang, 2011), where four objectives are considered, (i) is the minimization of the total cost considering the production and transportation cost, (ii) is the minimization of time of production and transportation, (iii) is the maximization of average product quality and (iv) is the maximization of green appraisal score. They implemented two algorithms: multi-objective genetic algorithm 1 (MOGA_1) and MOGA_2, also they tackled the original problem in four, where 3 of them are bi-objective problem and the last one considered all of them.

Similarly, for bi-level problems there are researches in which the environment aspects are considered. In (Mathew and Sharma, 2006) they wanted to solve a network design problem, where the leader determines the link capacity expansion subject to user travel behavior in order to minimize the system travel cost subject to user's travel behavior; and the follower determines the link flows subject to user equilibrium conditions.

In addition, some problems consider the pollution emission as in (Wang, Ma, and Li, 2011), where the government,who is the leader, chooses the optimal prime of the pollution emission with consideration to the response of the firms, which represents the follwer, to the price. The follower aims to maximize its profit. They gave an illustrative example in order to demonstrate the feasibility of their model.

Furthermore, a model that includes a measurement of gas emissions throughout a traffic network for an urban transportation is considered in (Hizir, 2006) , where the leader represents the transportation managers aiming to make the transport systems sustainable, while the follower represents the decisions of the network users minimizing their travel cost. They considered the emission pricing and emission permits, as pollution permits. Also, they developed an extension where they took into consideration the district pricing, capacity enhancement and emission dispersion in order to incorporate different policy measures for sustainability.

Other problem that considers the urban traffic congestion pricing policy is in (Wang et al., 2014), where the carbon emission cost were considered as part of the travel cost. The traffic management decision-making behavior is represented by the leader, maximizing the customer surplus; while the follower describes the user mode choice behavior minimizing the travel cost, where they combined traffic assignment based on network equilibrium distribution in the conditions of the congestion charge and transportation mode choice .

Chapter 3

Bilevel bi-objective multi-commodity problem

3.1 Problem statement

As mentioned in Chapter 1, we consider a situation in which two companies interact with each other in a hierarchical way within a supply chain. This hierarchical problem could be modeled as a bi-level mathematical program. These kind of problems consider two levels of decision known as upper and lower level, respectively. Usually, the upper level is associated with a leader and the lower level with a follower. In this problem, it is assumed that the distributing company will be the leader and the manufacturing company will be the follower. The leader will select the customers to be in the contract (see Figure 3.1a), and distribute the commodities, chossing the routes for the heterogeneous fleet of available vehicles and selecting the customers (see Figure 3.1b). After that, the follower will select the facilities in order to accomplish the request demand (see Figure 3.1c). In Figure 3.1d the two implicated decisions are illustrated.

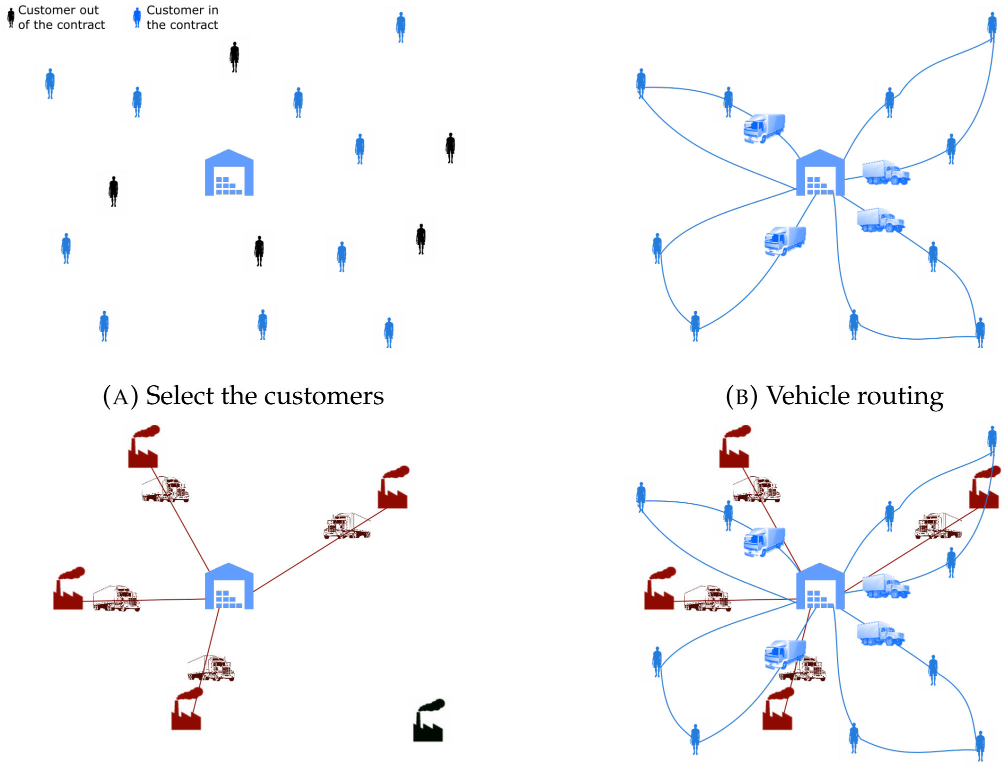
It is worth clarify that the leader will have two objectives: maximize the profit and minimize the CO_2 emissions; while the follower will have the aim of minimize its production costs.

Furthermore, we can observe that the manufacturing company should decide the production plan once the distributing company has selected a subset of customers. At the same time, the production plan will affect both leader's objectives due to the CO_2 emissions generated at each facility and because of the acquirement cost associated with the commodities from each facility.

Hence, the considerations that need to be taken into account in the problem could be seen as follows:

In the upper level problem:

- Customers included in the contract will be visited only once.
- The demand made by contract customers must be satisfied.
- The route begins and ends in the depot.
- The cycles are prohibited in a route.
- There is a maximum time for the routes.



(A) Select the customers

(B) Vehicle routing

(C) Select the facilities production

(D) Integrated decisions

FIGURE 3.1: Representation of the leader and follower decision

- An heterogeneous fleet of vehicles is considered.
- There is a maximum availability of vehicles from each type.

In the lower level problem:

- Satisfy all the contract customers demand for each commodity.
- There is a maximum rate of pollution allowed in each location.
- There is a capacity considered in each manufacturing facility.

3.2 Mathematical model

In this section the mathematical formulation is described. The sets, parameters, decision variables herein involved and the assumptions considered during this research are presented next.

Sets

I : set of customers

L : set of the different types of vehicles

$V(l)$: the set of the available vehicles from each type l

M : set of locations

N : set of products

$O = \{i \in I : x_i = 1\}$: set of all customers consider in the contract.

Parameters

$k(l)$: denote the $k-$th vehicle of the type l

e_l : constant of CO_2 emission from the vehicle type l

E_{mn} : constant of CO_2 emission of each type of product n from the location m (includes the production and shipping emissions)

g_n : constant of the obtained income of each type of product n.

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

D_{in} : demand of the product n by the customer i

r_l : rental cost of the vehicle type l

c_l : cost factor for distance traveled of the vehicle type l

C^1_{mn} : acquirement cost for the product n from the location m (includes the shipping cost from each location to the depot)

h_j : service time in the customer j

t_{ij} : it is the necessary time to arrive from the customer i to the customer j, it is define as the sum of the corrected distance and the service time in the customer j, it may look like $t_{ij} = d_{ij}\varphi + h_j$, where φ is a factor that converts distance in time, also the $h_0 = 0$

C^2_{mn} : production and shipping costs of the product n in the location m

q_n: resources required to elaborate the product n

Q_l: available capacity of the vehicle type l

t^{max}: maximum duration of the route

ε^{max}_m : maximum CO_2 emission rate permitted in the location m

Π^{max}_m : maximum availability of production in the location m

The binary leader's decision variables considered in the problem and the integer follower's variables are detailed next:

$$x_i = \begin{cases} 1, & \text{if the customer } i \text{ is consider in the contract} \\ 0, & \text{otherwise} \end{cases}$$

$$y^{k(l)}_{ij} = \begin{cases} 1, & \text{if the arc } (i,j) \text{ is in the route of the vehicle } k \text{ of type } l \\ 0, & \text{otherwise} \end{cases}$$

$$z^{k(l)} = \begin{cases} 1, & \text{if the } k\text{-th vehicle of type } l \text{ is used} \\ 0, & \text{otherwise} \end{cases}$$

p_{mn} = amount of the $n-$th commodity manufactured at facility m

Remark: Without loss of generality, consider the depot as the node 0. Also, define the set $I^* = I \cup \{0\}$

On the basis of the above, our bi level formulation is defined as follows

$$\min \quad \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)}(e_l d_{ij})y_{ij}^{k(l)} + \sum_{m\in M}\sum_{n\in N}E_{mn}p_{mn} \tag{3.1}$$

$$\max \quad \sum_{i\in I}(\sum_{n\in N}g_n D_{in})x_i - \sum_{l\in L}r_l(\sum_{k(l)\in V(l)}z^{k(l)}) - \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)}(c_l d_{ij})y_{ij}^{k(l)}$$
$$-\sum_{m\in M}\sum_{n\in N}C^1_{mn}p_{mn} \tag{3.2}$$

subject to :

$$\sum_{j\in I} y_{0j}^{k(l)} = z^{k(l)} \qquad \forall l \in L, k(l) \in V(l) \tag{3.3}$$

$$\sum_{i\in I} y_{i0}^{k(l)} = z^{k(l)} \qquad \forall l \in L, k(l) \in V(l) \tag{3.4}$$

$$\sum_{j\in I}\sum_{l\in L}\sum_{k(l)\in V(l)} y_{ij}^{k(l)} = x_i \qquad \forall i \in I \tag{3.5}$$

$$\sum_{j\in I} y_{ij}^{k(l)} = \sum_{j\in I} y_{ji}^{k(l)} \quad \forall i \in I, l \in L, k(l) \in V(l) \tag{3.6}$$

$$\sum_{i\in W}\sum_{j\in W} y_{ij}^{k(l)} \le |W| - 1 \qquad {}_{\substack{W\subseteq I^*, 2\le |W|\le |I|+1 \\ \forall l\in L, k(l)\in V(l)}} \tag{3.7}$$

$$\sum_{i\in I^*}\sum_{\substack{j\in I \\ i\neq j}} t_{ij}y_{ij}^{k(l)} \le T^{max} \qquad \forall l \in L, k(l) \in V(l) \tag{3.8}$$

$$\sum_{i\in I}\sum_{\substack{j\in I^* \\ i\neq j}}(\sum_{n\in N} D_{in})y_{ij}^{k(l)} \le Q_l^{max} \; \forall l \in L, k(l) \in V(l) \tag{3.9}$$

$$\sum_{k(l)\in V(l)} z^{k(l)} \le |V(l)| \quad \forall l \in L \tag{3.10}$$

$$\sum_{i\in I}\sum_{j\in I} y_{ij}^{k(l)} \le z^{k(l)}\Gamma \qquad \forall l \in L, k(l) \in V(l) \tag{3.11}$$

$$y_{ij}^{k(l)}, x_i, z_l \in \{0,1\} \forall i,j \in I^*, l \in L, k(l) \in V(l) \tag{3.12}$$

in which for a fixed leader's decision $y_{ij}^{k(l)}, x_i, z_l$, the follower's variables p_{mn} solve

$$\min \quad \sum_{m\in M}\sum_{n\in N} C^2_{mn} p_{mn} \tag{3.13}$$

subject to:

$$\sum_{m\in M} p_{mn} = \sum_{i\in O} D_{in} x_i \qquad \forall n \in N \tag{3.14}$$

$$\sum_{n\in N} E_{mn} p_{mn} \leq \varepsilon_m^{max} \qquad \forall m \in M \tag{3.15}$$

$$\sum_{n\in N} q_n p_{mn} \leq \Pi_m^{max} \qquad \forall m \in M \tag{3.16}$$

$$p_{mn} \in Z^+ \cup \{0\} \qquad \forall m \in M, n \in N \tag{3.17}$$

The model defined by (3.1)-(3.17) is a bi-objective bi-level linear programming problem. In equation (3.1) one of the leader's objective function is presented, it measures the CO_2 emissions caused by the route of each type of vehicle and the CO_2 generated by all the facilities. In (3.2) the other leader's objective function is stated, in which the total profit is computed, the first part represents the total income per each commodity demanded by the customers, the second part is the total rental cost associated with the vehicles, the third part is the total transportation cost in all the vehicles and the last term represents the total acquirement cost for commodities from the facilities. Now, for the upper level constraints, (3.3) requires that each vehicle has a single departure away from the depot, (3.4) requires that each vehicle only arrives once to the depot. Also, (3.5) indicates that the customers included in the contract must be only visited once, (3.6) is the flow conservation constraint, and (3.7) corresponds to classical constraint that avoid subtours. In constraint (3.8) ensures that the time associated with each route should not exceed the maximum time established, (3.9) states that the demanded commodities in each route would not exceed the available capacity of the vehicle type l, (3.10) indicates that there is a maximum availability of vehicles from each type, (3.11) ensures that only vehicles that are being used could have an assigned route and (3.12) establishes the binary constraints for each variable $y_{ij}^{k(l)}$, x_i and z_l.

In constraint (3.13), the follower's objective function is presented. The follower's problem is defined by (3.13)-(3.17). In (3.13), the minimization of the manufacturing and shipping costs is aimed. In (3.14), the satisfaction of all the contracted customer's demand is assured. Constraint (3.15) guarantees that the CO_2 emissions for manufacturing the demanded commodities should not exceed the maximum CO_2 emissions rate permitted at each facility, and (3.16) states that the manufactured commodities not exceed the maximum production available at each facility. Finally, (3.17) restricts that the follower's variables p_{mn} are non-negative integers.

In order to have well-defined the proposed bi-level problem, the optimistic approach is assumed. In other words, in the case when the lower level problem has multiple optimal solutions for a leader's decision, the most convenient for the leader among these solutions will be selected by the follower. This can be seen as a cooperative scheme, but since the follower would not take any benefit by affecting the leader's objective function, it is commonly assumed, see (Kalashnikov et al., 2015) and (Sinha, Malo, and Deb, 2016).

3.3 Analizing simplifications of the problem

It is well known that vehicle routing problems are difficult to solve. Also, integrating the manufacturing process and distribution decisions complicates the problem. Furthermore, bi-level problems are in general complex to solve. Hence, considering all these characteristics within the same framework will increase the degree of intractability of the problem. Based on the latter, we decided to start the analysis of the problem herein considered by studying an intuitive simplification.

First, we consider the single-commodity version of the problem; that is, when $n = 1$. Under this assumption, the lower level problem can be solved via a straightforward greedy algorithm. It is enough to assign -as much as possible- the cumulative demand to the cheapest facility. The maximum amount of assigned demand will be given by equation (3.14) or (3.15), that is, the one that depletes its resource first.

The latter property is obtained from the simplification of the problem, defined by (3.1)-(3.17). This simplification yields us to include some additional constraints in order to eliminate the structure of lower level problem but keeping it into consideration. In other words, the bi-level bi-objective problem could be reduced into a single-level one. It is clear that the follower's constraints (3.14)-(3.17) need to be explicitly maintained. On the other hand, the follower's objective function (3.13) can be replaced by two constraints that guarantee the minimization of the manufacturing costs.

Let define $S_{\hat{m}} = \{m \in M | C^2_m > C^2_{\hat{m}}\}\ \forall \hat{m} \in M$ as the set that contains, for each facility $\hat{m}$, all the facilities m that are associated with a more expensive cost than the one for $\hat{m}$. Also, let $MaxProd_m = \min\{\Pi^{max}_m, \varepsilon^{max}_m / E_m\}\ \forall m \in M$ be the maximum manufacturing allowed at facility m. The aim of considering $MaxProd_m$ is to select the minimum between the CO_2 emissions allowed and the manufacturing. Furthermore, the following auxiliary variable is included:

$$w_m = \begin{cases} 0, & \text{if } p_m = MaxProd_m \\ 1, & \text{if } p_m < MaxProd_m \end{cases}$$

Then, the next two constraints are included to guarantee the optimal in equation (3.13).

$$w_m \leq MaxProd_m - p_m \leq w_m P \quad \forall m \in M \tag{3.18}$$

$$\sum_{m \in S_i} p_m \leq P(1 - w_i) \quad \forall i \in M \tag{3.19}$$

These constraints assure that the cheapest available facility will achieve the maximum production before the other facilities, where P is a large number. For example, suppose that the cheapest facility is the m-th. So, if the customer's demand is less than $MaxProd_m$, then w_m will be equal to 1; and by equation (3.19) it is forced the other facilities does not have any manufacturing assigned to them. On the contrary, that is, if customer's demand exceeds $MaxProd_m$, then w_m will be equal to 0, so equation (3.19) allows that some of the other facilities could manufacture a part of the demand.

Therefore, it is convenient to emphasize that under the single-commodity assumption, the problem can be reformulated as a single-level one.

Proposition 3.1 *If $(\bar{x}, \bar{y}, \bar{z}, \bar{p})$ is a feasible solution for the bi-level single-commodity problem, then the inequalities (3.18) and (3.19) are satisfied, therefore it is a feasible solution for the single level problem*
Proof. *Let suppose that $(\bar{x}, \bar{y}, \bar{z}, \bar{p})$ is not a feasible solution for the single level problem. Hence, either of (3.18) or (3.19) is not satisfied. Then, exists an $i \in M$ such that*

$$\sum_{\bar{m} \in S_i} \bar{p}_{\bar{m}} > P(1 - w_i)$$

As P is a large number, then $w_i = 1$ and at least an $\bar{p}_{\bar{m}} > 0$ such that $\bar{m} \in S_i$. For the constraint (3.18) we have that if $w_i = 1$, $\bar{p}_i < MaxProd_i$. Then we have that

$$\sum_{m \in M} C^2_m \bar{p}_m = \sum_{\substack{m \in M \wedge \\ m \neq i, \bar{m}}} C^2_m \bar{p}_m + C^2_{\bar{m}} \bar{p}_{\bar{m}} + C^2_i \bar{p}_i$$

Consider that $\bar{p}_{\bar{m}}$ can be expressed as $\bar{p}_{\bar{m}} = \bar{p}_{\bar{m}1} + MaxProd_i - \bar{p}_i$ such that $p_{\bar{m}'} \geq 0$ then substitute $\bar{p}_{\bar{m}}$ in the above expression

$$\begin{aligned}
&\sum_{\substack{m \in M \wedge \\ m \neq i, \bar{m}}} C^2_m \bar{p}_m + C^2_{\bar{m}} \bar{p}_{\bar{m}} + C^2_i \bar{p}_i \\
&\qquad = \sum_{\substack{m \in M \wedge \\ m \neq i, \bar{m}}} C^2_m \bar{p}_m + C^2_{\bar{m}} (\bar{p}_{\bar{m}'} + MaxProd_i - \bar{p}_i) + C^2_i \bar{p}_i \\
&\qquad = \sum_{\substack{m \in M \wedge \\ m \neq i, \bar{m}}} C^2_m \bar{p}_m + C^2_{\bar{m}} \bar{p}_{\bar{m}'} + C^2_{\bar{m}} (MaxProd_i - \bar{p}_i) + C^2_i \bar{p}_i
\end{aligned}$$

As $C^2_{\bar{m}} > C^2_i$ given that $\bar{m} \in S_i$ then

$$\begin{aligned}
&\sum_{\substack{m \in M \wedge \\ m \neq i, \bar{m}}} C^2_m \bar{p}_m + C^2_{\bar{m}} \bar{p}_{\bar{m}1} + C^2_{\bar{m}} (MaxProd_i - \bar{p}_i) + C^2_i \bar{p}_i \\
&\qquad > \sum_{\substack{m \in M \wedge \\ m \neq i, \bar{m}}} C^2_m \bar{p}_m + C^2_{\bar{m}} \bar{p}_{\bar{m}1} + C^2_i (MaxProd_i - \bar{p}_i) + C^2_i \bar{p}_i
\end{aligned}$$

We found a facility with less cost (i) than the facility ($\bar{m}$) that improve the objective function of the lower level. Furthermore, this is a contradiction because $\bar{p}$ was an optimal solution for the lower level of the bilevel problem.
Therefore,

$$\sum_{\bar{m} \in S_i} \bar{p}_{\bar{m}} \not> P(1 - w_i) \qquad \blacksquare$$

Proposition 3.2 *Let $(x^*, y^*, z^*, p^*, w^*)$ be the optimal solution for the single level problem. Then, it is a feasible solution for the bi-level problem.*
Proof. *As (x^*, y^*, z^*) satisfied the constraints (3.3)-(3.12) for the bi-level problem and we know that p^*, w^* satisfy the constraints (3.18) and (3.19).*
Without loss generality, we will suppose that $C_1^2 < C_2^2 < C_3^2 < ... < C_{|M|}^2$.
The proof will be divided in cases:

- *Let be the customers' demand less than the maximum production allowed ($\sum_{i \in I} D_i x_i \leq MaxProd_1$), then by equations (3.18) and (3.19) we have that $p_1^* = \sum_{i \in I} D_i x_i$ and $p_{\bar{m}}^* = 0\ \forall \bar{m} > 1 \in M$*
$$C_1^2 p_1^* = \sum_{m \in M} C_m^2 p_m^*$$
As C_1^2 is the lowest cost for every other costs we have that $C_1^2 p_1^* < C_{\bar{m}} \sum_{i \in I} D_i x_i$ $\forall \bar{m} \in M$ then
$$C_1^2 p_1^* = \min_p \sum_{m \in M} C_m^2 p_m$$

- *Let be the customers' demand between the maximum production allowed in facility 1 and the sum of the two firsts facilities ($\sum_{i \in I} D_i x_i > MaxProd_1$ and $\sum_{i \in I} D_i x_i \leq MaxProd_1 + MaxProd_2$), then by equations (3.18) and (3.19) we have that $p_1^* = MaxProd_1 \wedge p_2^* = (\sum_{i \in I} D_i x_i - MaxProd_1)$ and $p_{\bar{m}}^* = 0\ \forall \bar{m} > 2 \in M$*
$$C_1^2 p_1^* + C_2^2 p_2^* = \sum_{m \in M} C_m^2 p_m^*$$
As C_1^2 is the lowest cost for every other costs we have that $C_1^2 p_1^* < C_{\bar{m}}\ MaxProd_1 \forall \bar{m} \neq 1 \in M$ and $C_2^2 p_2^* < C_{\bar{m}}\ (\sum_{i \in I} D_i x_i - MaxProd_1) \forall \bar{m} > 2 \in M$ then
$$C_1^2 p_1^* + C_2^2 p_2^* = \min_p \sum_{m \in M} C_m^2 p_m^*$$

- *Let suppose that the lower level achieves the optimal when the customers demand is between the sum of the first maximum production allowed in facilities $k-1$ and the sum of the first maximum production allowed in facilities k such that*
$$\sum_{m=1}^{k} C_m^2 p_m^* = \min_p \sum_{\tilde{m} \in M} C_{\tilde{m}}^2 p_{\tilde{m}}$$
Therefore, the customers demand is $\sum_{i \in I} D_i x_i > \sum_{m=1}^{k-1} MaxProd_m$ and $\sum_{i \in I} D_i x_i \leq \sum_{m=1}^{k} MaxProd_m$, then by equations (3.18) and (3.19) we have that $p_m^* = MaxProd_m$ $\forall m < k \wedge p_k^* = (\sum_{i \in I} D_i x_i - \sum_{m=1}^{k-1} MaxProd_m)$ and $p_{\bar{m}}^* = 0\ \forall \bar{m} > k$ such that $\bar{m} \in M$
$$\sum_{m=1}^{k} C_m^2 p_m^* = \sum_{\tilde{m} \in M} C_{\tilde{m}}^2 p_{\tilde{m}}^*$$
To prove that the lower level achieves the optimal when the customers demand is between the sum of the firsts maximum production allowed in facilities k and the sum of the firsts maximum production allowed in facilities $k+1$, that means $\sum_{i \in I} D_i x_i > \sum_{m=1}^{k} MaxProd_m$ and $\sum_{i \in I} D_i x_i \leq \sum_{m=1}^{k+1} MaxProd_m$, then by equations (3.18) and (3.19) we have that $p_m^* = MaxProd_m\ \forall m < k+1$ and $p_{k+1}^* = (\sum_{i \in I} +1 D_i x_i -$

$\sum_{m=1}^{k} MaxProd_m$) *and* $p^*_{\bar{m}} = 0\ \forall \bar{m} > k+1$ *such that* $\bar{m} \in M$

$$\sum_{m=1}^{k+1} C^2_m p^*_m = \sum_{\bar{m} \in M} C^2_{\bar{m}} p^*_{\bar{m}}$$

We have that

$$\sum_{m=1}^{k+1} C^2_m p^*_m = \sum_{m=1}^{k} C^2_m p^*_m + C^2_{k+1} p^*_{k+1}$$

As $\sum_{m=1}^{k} C^2_m p^*_m$ *is the minimum possible value with a demand equal or less than* $\sum_{m=1}^{k} MaxProd_m$ *by the last assumption.*
Then, we know that $C^2_{k+1} p^*_{k+1} < C_{\bar{m}} \left(\sum_{i \in I} D_i x_i - \sum_{m=1}^{k} MaxProd_m \right)\ \forall \bar{m} > k+1 \in M$

$$\sum_{m=1}^{k} C^2_m p^*_m + C^2_{k+1} p^*_{k+1} < \sum_{m=1}^{k} C^2_m p^*_m + C_{\bar{m}} \left(\sum_{i \in I} D_i x_i - \sum_{m=1}^{k} MaxProd_m \right)$$

Then

$$\sum_{m=1}^{k+1} C^2_m p^*_m < \sum_{m=1}^{k} C^2_m p^*_m + C_{\bar{m}} \left(\sum_{i \in I} D_i x_i - \sum_{m=1}^{k} MaxProd_m \right)$$

Therefore

$$\sum_{m=1}^{k+1} C^2_m p^*_m = \min_p \sum_{\bar{m} \in M} C^2_{\bar{m}} p_{\bar{m}}$$

Therefore, the lower level will achieve the optimal with any customers demand.
Then, it is a bi-level feasible solution ■

Proposition 3.3 *The bi-level problem defined by the equations (3.22)-(3.27) in section 3.3.2.1 and the single level problem defined by the equations (3.21)-(3.27) in section 3.3.1.1 have the same optimal solution*
Proof. *Let* (x^*, y^*, z^*, p^*) *be an optimal and feasible solution for the bi-level problem and be represented in the single level problem as* $(x^*, y^*, z^*, p^*, w^*)$*, then by Proposition 3.1 we have that it is a feasible solution for the single level problem.*
Then, lets assume that $(x^*, y^*, z^*, p^*, w^*)$ *is not an optimal solution for the single level problem. Then there is a* (x, y, z, p, w) *in the single level problem (represented as* (x, y, z, p) *in the bi-level problem) such that*

$$\sum_{i \in I} g D_i x^*_i - \sum_{l \in L} r_l (\sum_{k(l) \in V(l)} z^{*k(l)}) - \sum_{i \in I^*} \sum_{j \in I^*} \sum_{l \in L} \sum_{k(l) \in V(l)} (c_l d_{ij}) y^{*k(l)}_{ij} - \sum_{m \in M} C^1_m p^*_m$$
$$< \sum_{i \in I} g D_i x_i - \sum_{l \in L} r_l (\sum_{k(l) \in V(l)} z^{k(l)}) - \sum_{i \in I^*} \sum_{j \in I^*} \sum_{l \in L} \sum_{k(l) \in V(l)} (c_l d_{ij}) y^{k(l)}_{ij} - \sum_{m \in M} C^1_m p_m$$

By Proposition 3.2 (x, y, z, p) *is feasible solution for the bi-level problem. This contradicts that* (x^*, y^*, z^*, p^*) *is an optimal solution for the bi-level problem.*
Conversely, let $(x^*, y^*, z^*, p^*, w^*)$ *be an optimal solution for the single level problem be represented as* (x^*, y^*, z^*, p^*) *in the bi-level problem, by Proposition 3.2 we have that is a feasible solution for the bilevel problem.*

Then, exists a feasible solution for the bilevel problem (x, y, z, p) such that

$$\sum_{i\in I} gD_i x_i^* - \sum_{l\in L} r_l(\sum_{k(l)\in V(l)} z^{*k(l)}) - \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)} (c_l d_{ij}) y_{ij}^{*k(l)} - \sum_{m\in M} C_m^1 p_m^*$$
$$< \sum_{i\in I} gD_i x_i - \sum_{l\in L} r_l(\sum_{k(l)\in V(l)} z^{k(l)}) - \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)} (c_l d_{ij}) y_{ij}^{k(l)} - \sum_{m\in M} C_m^1 p_m$$

By Proposition 3.1 (x, y, z, p) it is a feasible solution for the single level problem. It contradicts that $(x^*, y^*, z^*, p^*, w^*)$ is an optimal solution for the single level problem. ∎

3.3.1 Mono-objective single-commodity problem

Now, maintaining the single-commodity assumption, we consider some scenarios for simulating the impact that will have the fact that different maximum levels for CO_2 emissions are fixed. By doing this, the equation (3.1) it is not longer considered as an objective function but as a constraint. Therefore, the bi-objective problem will be reduced into a mono-objective one.

The new constraint that will be included to consider the upper levels regarding the emissions is set up as follows:

$$\sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)} (e_l d_{ij}) y_{ij}^{k(l)} + \sum_{m\in M} E_m p_m \le MaxLevel \tag{3.20}$$

where $MaxLevel$ represents the established maximum emission level. Hence, as it is mentioned above the leader's objective function stated in (3.1) is deleted from the mathematical model and equation (3.20) will be included in the constraints. The resulting model is a single-level mono-objective linear programming problem.

3.3.1.1 Mathematical model

This simplified model will include constrains (3.18), (3.19) and (3.20) but without the structure of a bi-level problem. Hence, it is represented as the following single-level problem:

$$\max \quad \sum_{i\in I} gD_i x_i - \sum_{l\in L} r_l(\sum_{k(l)\in V(l)} z^{k(l)}) - \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)} (c_l d_{ij}) y_{ij}^{k(l)}$$
$$- \sum_{m\in M} C_m^1 p_m \tag{3.21}$$

subject to:

$$\sum_{j\in I} y_{0j}^{k(l)} = z^{k(l)} \qquad \forall l \in L, k(l) \in V(l) \tag{3.3}$$

$$\sum_{i\in I} y_{i0}^{k(l)} = z^{k(l)} \qquad \forall l \in L, k(l) \in V(l) \tag{3.4}$$

$$\sum_{j\in I}\sum_{l\in L}\sum_{k(l)\in V(l)} y_{ij}^{k(l)} = x_i \qquad \forall i \in I \tag{3.5}$$

$$\sum_{j \in I} y_{ij}^{k(l)} = \sum_{j \in I} y_{ji}^{k(l)} \qquad \forall i \in I, l \in L, k(l) \in V(l) \tag{3.6}$$

$$\sum_{i \in W} \sum_{j \in W} y_{ij}^{k(l)} \leq |W| - 1 \qquad {\scriptstyle W \subseteq I^*, 2 \leq |W| \leq |I|+1 \atop \forall l \in L, k(l) \in V(l)} \tag{3.7}$$

$$\sum_{i \in I^*} \sum_{\substack{j \in I \\ i \neq j}} t_{ij} y_{ij}^{k(l)} \leq T^{max} \qquad \forall l \in L, k(l) \in V(l) \tag{3.8}$$

$$\sum_{i \in I} \sum_{\substack{j \in I^* \\ i \neq j}} D_i y_{ij}^{k(l)} \leq Q_l^{max} \qquad \forall l \in L, k(l) \in V(l) \tag{3.22}$$

$$\sum_{k(l) \in V(l)} z^{k(l)} \leq |V(l)| \qquad \forall l \in L \tag{3.10}$$

$$\sum_{i \in I} \sum_{j \in I} y_{ij}^{k(l)} \leq z^{k(l)} \Gamma \qquad \forall l \in L, k(l) \in V(l) \tag{3.11}$$

$$\sum_{i \in I^*} \sum_{j \in I^*} \sum_{l \in L} \sum_{k(l) \in V(l)} (e_l d_{ij}) y_{ij}^{k(l)} + \sum_{m \in M} E_m p_m \leq MaxLevel \tag{3.20}$$

$$w_m \leq MaxProd_m - p_m \leq w_m P \quad \forall m \in M \tag{3.18}$$

$$\sum_{m \in S_i} p_m \leq P(1 - w_i) \quad \forall i \in M \tag{3.19}$$

$$\sum_{m \in M} p_m = \sum_{i \in O} D_i x_i \tag{3.23}$$

$$E_m p_m \leq \varepsilon_m^{max} \qquad \forall m \in M \tag{3.24}$$

$$p_m \leq \Pi_m^{max} \qquad \forall m \in M \tag{3.25}$$

$$y_{ij}^{k(l)}, x_i, z_l \in \{0, 1\} \qquad \forall i, j \in I^*, l \in L, k(l) \in V(l) \tag{3.12}$$

$$w_m \in \{0, 1\} \qquad \forall m \in M \tag{3.26}$$

$$p_m \in Z^+ \cup \{0\} \qquad \forall m \in M \tag{3.27}$$

3.3.2 Bi-objective single-commodity problem

Now, in order to try to return to the original problem, the case for the bi-objective single-commodity problem will be presented. Within this simplification, we are also taking into consideration the single-level problem because of the single-commodity assumption is maintained. Hence, the simple structure associated with the lower level remains.

3.3.2.1 Mathematical model

The resulting model for this case will be almost the same as the mono-objective single-commodity problem, defined by (3.21)-(3.27). The only constraint that needs to be omitted is the one given by equation (3.20). So, the considered objective functions will be the equivalent equations to (3.1) and (3.2) but for the single-commodity case. Then, the mathematical formulation is define as follows:

$$\min \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)} (e_l d_{ij}) y_{ij}^{k(l)} + \sum_{m\in M} E_m p_m \tag{3.28}$$

$$\max \sum_{i\in I} gD_i x_i - \sum_{l\in L} r_l(\sum_{k(l)\in V(l)} z^{k(l)}) - \sum_{i\in I^*}\sum_{j\in I^*}\sum_{l\in L}\sum_{k(l)\in V(l)} (c_l d_{ij}) y_{ij}^{k(l)} - \sum_{m\in M} C_m^1 p_m \tag{3.21}$$

subject to:

$$\sum_{j\in I} y_{0j}^{k(l)} = z^{k(l)} \qquad \forall l \in L, k(l) \in V(l) \tag{3.3}$$

$$\sum_{i\in I} y_{i0}^{k(l)} = z^{k(l)} \qquad \forall l \in L, k(l) \in V(l) \tag{3.4}$$

$$\sum_{j\in I}\sum_{l\in L}\sum_{k(l)\in V(l)} y_{ij}^{k(l)} = x_i \qquad \forall i \in I \tag{3.5}$$

$$\sum_{j\in I} y_{ij}^{k(l)} = \sum_{j\in I} y_{ji}^{k(l)} \quad \forall i \in I, l \in L, k(l) \in V(l) \tag{3.6}$$

$$\sum_{i\in W}\sum_{j\in W} y_{ij}^{k(l)} \le |W| - 1 \qquad \substack{W\subseteq I^*, 2\le |W| \le |I|+1 \\ \forall l\in L, k(l)\in V(l)} \tag{3.7}$$

$$\sum_{i\in I^*}\sum_{\substack{j\in I \\ i\neq j}} t_{ij} y_{ij}^{k(l)} \le T^{max} \qquad \forall l \in L, k(l) \in V(l) \tag{3.8}$$

$$\sum_{i\in I}\sum_{\substack{j\in I^* \\ i\neq j}} D_i y_{ij}^{k(l)} \le Q_l^{max} \qquad \forall l \in L, k(l) \in V(l) \tag{3.22}$$

$$\sum_{k(l)\in V(l)} z^{k(l)} \le |V(l)| \qquad \forall l \in L \tag{3.10}$$

$$\sum_{i\in I}\sum_{j\in I} y_{ij}^{k(l)} \le z^{k(l)}\Gamma \qquad \forall l \in L, k(l) \in V(l) \tag{3.11}$$

$$w_m \le MaxProd_m - p_m \le w_m P \quad \forall m \in M \tag{3.18}$$

$$\sum_{m\in S_i} p_m \le P(1 - w_i) \quad \forall i \in M \tag{3.19}$$

$$\sum_{m\in M} p_m = \sum_{i\in O} D_i x_i \tag{3.23}$$

$$E_m p_m \le \varepsilon_m^{max} \qquad \forall m \in M \tag{3.24}$$

$$p_m \le \Pi_m^{max} \qquad \forall m \in M \tag{3.25}$$

$$y_{ij}^{k(l)}, x_i, z_l \in \{0,1\} \forall i, j \in I^*, l \in L, k(l) \in V(l) \tag{3.12}$$

$$w_m \in \{0,1\} \forall m \in M \tag{3.26}$$

$$p_m \in Z^+ \cup \{0\} \forall m \in M \tag{3.27}$$

.

Chapter 4

Proposed algorithms

In this chapter the proposed algorithms are detailed. First a brief introduction of tabu search, co-evolutionary algorithms and multi-objective genetic algorithms is given. Second, a tabu search for solving the mono-objective single-commodity problem is described. After that, an adaptation of the tabu search for solving the bi-objective single-commodity problem is explained. Next, an adaptation of the tabu search for solving the bi-level bi-objective problem is given. Later, a co-evolutionary algorithm based on a biased random keys for solving the bi-level bi-objective problem is described. Finally, an adaptation of the path relinking for solving the bi-level bi-objective problem is explained.

Tabu search is a meta-heuristic derives from (Glover, 1986). This meta-heuristic has a strategy of prohibited certain moves, that will have a tabu status; this strategy aims to prevent cycling. An explored move loses its tabu status after a predefined time, becoming an accessible solution. Furthermore, choosing a poor move is evaded, except if we want to avoid a path already examined. In the tabu list the recent movements are recorded in the order in which they are executed.

Co-operative co-evolutionary algorithm is defined in (Potter and De-Jong, 1994), in which multiple subpopulations interact with each other in a co-operative way, representing the coevolution of the species. This system combines 5 ideas: (i) a specie represent a component of a solution, (ii) the involved species represent the complete solution, (iii) the impact of each specie is defined in terms of fitness of the complete solution, (iv) the subpopulations should itself evolve, (v) each especie evolves managed by a genetic algorithm.

Furthermore, most of the multi-objective optimization problems are handled by evolutionary approaches, as it is mentioned in (Konak, Coit, and Smith, 2006). They summarized the most common evolutionary approaches: vector evaluated GA, Multi-objective Genetic Algorithm, Niched Pareto Genetic Algorithm, Weight-based Genetic Algorithm, Random Weighted Genetic Algorithm, Nondominated Sorting Genetic Algorithm, and Strength Pareto Evolutionary Algorithm.

The path relinking algorithm is first introduced in (Glover and Laguna, 1993). The main objective of this procedure is to explore the search path between a set of (two) solutions. In (Ho and Gendreau, 2006) is mentioned that the path relinking should considered three components: rules for building the initial solutions set, rules for choosing the initial and guiding solutions, and a neighborhood structure for moving along paths.

4.1 A tabu search for the mono-objective single-commodity problem

Despite the fact that the model defined by (3.21)-(3.27) is an integer programming problem and it could be solved to optimality through a commercial software, for medium and large size instances the latter is not possible. Therefore, in order to efficiently solve the problem a tabu search algorithm is developed. A detailed description is presented next.

<u>Solution encoding</u>

The solution is represented as a matrix, in which each row corresponds to an available vehicle. Also, an extra row is considered in the matrix that contains the no contracted customers. Each column corresponds to a customer. Hence, the matrix will be of dimension $\sum_{l \in L} |V(l)| \times |I|$.

It is important to specify that all the decision variables included in the model are explicit or implicitly represented in the solution encoding. For example, regarding binary variables x, if the $i-$th customer appears in the first row, it is associated to the no contracted customers, then it implies that $x_i = 0$. If the $i-$th customer appears in a row that is different from the first one, then it implies that $x_i = 1$, that is, the $i-$th customer is included in the contract. On the contrary, variable y, it is implicitly represented. Remember that y is associated with the routing decision.Depending on the solution encoding it could be inferred which customers will be associated to the vehicles, the routes will be decided by a constructive algorithm. Now, for variable z it is easy to realize that if there is a customer assigned in the $j-$th row, then the corresponding vehicle is being used and $z_j = 1$. Finally, the p and w variables will be implicitly obtained after knowing the variable x due to the accumulated demand that needs to be satisfied.

In order to clarify the above mentioned, consider an illustrative example with 10 customers and 2 types of vehicles. Assume that there are 3 and 2 available vehicles for each type, correspondingly. A solution could be represented as in Figure (4.1). It can be seen that there are 6 rows -five associated with the available vehicles and the other one with the no contracted customers-. Also, the same number of columns than customers is considered. Moreover, it can be observed from the solution that customers 1, 5 and 9 are not in the contract. Also, the first type of vehicle will have customer 6 assigned to vehicle 1, customers 8 and 10 are in the vehicle 2, and customers 2 and 4 are in the vehicle 3. For the second type of vehicle, the vehicle 1 does not has customers assigned to it, so is not used, and the customers 3 and 7 are in the vehicle 2.

<u>Initial solution</u>

An initial solution should be constructed in order to enter into the tabu search scheme. First, the number of customers that will be in the contract are selected in a random way between 0 and $|I|$. Then, some customers will be randomly added into the contract until the solution is completed. Later, the selected customers are assigned to a vehicle in a random way (among all the types of vehicles). Finally, the

1	5	9							

6									
8	10								
2	4								

3	7								

FIGURE 4.1: Example of solution encoding

routes for each vehicle will be solved via a constructive algorithm and the facilities' production will be decided via a greedy algorithm.

Constructive algorithm: First, the customers that are assigned to a specific vehicle are sorted in a lexicographically manner. If the vehicle only has one or two customers, then the route is trivially found. But, if the vehicle has more than two customers, then the algorithm follows the next steps: (i) include the first two customers from the sorted list into the route, (ii) find the best position to allocate to the next customer and include him into the route. This step is repeated until all the customers are included into the route.

Greedy algorithm: This algorithm consists in selecting the available facility with the less manufacturing cost and assigning the maximum demand allowed to that facility or until the customers' demand is achieved. In case when the customer's demand is not satisfied, the next available cheapest facility is identified and the procedure is repeated until the whole customers' demand is satisfied.

Since the constructed solution may be infeasible due to the vehicle's capacity or the maximum duration time, a procedure that repairs that solution into a feasible one is included.

Repairing an infeasible solution

Infeasible initial solutions must be repaired. When a route that belongs to an infeasible solution violates the capacity of the vehicle, inter-routes movements are performed; that is, a customer assigned to the infeasible route should be moved into a feasible one. In the case when there are not feasible routes, the selected customer will be moved into a vehicle that is not being used. On the other hand, if the infeasibility is due to the duration of the route, an intra-route movement is performed, that is, swaps between customers assigned into the same route are done. If after intra-route movements the route remains infeasible, inter-routes movements must be conducted in the same manner than the one described above. These procedures are repeated until the solution becomes feasible.

Once a feasible solution is obtained, it will enter into the tabu search framework designed for the mono-objective single-commodity problem.

Step 0 **Initialization** We have considered two tabu structures, one for each kind of neighborhood. The first tabu structure is a matrix, in which, each row represents a customer $i \in I$ and the columns represent the vehicles and an extra column to represent the customers that are not in the contract. The second structure is a square matrix, in which, each column or row represents a vehicle. The tabu structures are initialized as empty. Construct a solution with the procedure described above. If it is feasible, then continue to Step 1. If it is infeasible, then repair it in order to achieve feasibility. Once the solution is repaired continue to Step 1.

Step 1 **Neighborhoods** Two neighborhoods are considered in this algorithm. *Neighborhood 1* (N_1) can be divided in two phases. The first phase consists in move each customer that is not in the contract into the vehicles $k(l) \in V(l) \quad \forall l \in L$; by doing this it is evident that the customer will be included in the contract. The second one deals with the case when the customer is already in the contract assigned to vehicle $k(\hat{l})$. In this case, it will be moved into a different vehicle $k(l) \in V(l) \quad \forall l \neq \hat{l} \in L$ or it will be removed from the contract. Furthermore, *Neighborhood 2* (N_2) consists in swap a vehicle $k(l) \in V(l)$ with the vehicles $k(\hat{l}) \in V(\hat{l}) \quad \forall \hat{l} \neq l \in L$. Hence, $N = N_1 \cup N_2$. Only for feasible solutions contained in N the objective function value will be computed. Death penalty is applied for the infeasible ones.

Step 2 **Select the best movement** The best solution contained in N that is not in the tabu list will be selected. The aspiration criterion is when the best solution in N belongs into the tabu list but it is better than the incumbent solution.

Step 3 **Update the tabu structures** Update the incumbent solution and both tabu structures. The tabu list size will be chosen in a random way between 8 and 15 depending on the best solution found in the iteration. Then, the tabu structure will be modified by adding a movement into it. Also, the incumbent solution is updated.

Step 4 **Stopping criterion** If the stopping criterion is not satisfied, then return to Step 2. The stopping criterion considered through this algorithm is a consecutive number of iterations in which the incumbent solution is not improved.

4.2 A tabu search for the bi-objective single-commodity problem

Naturally, the tabu search algorithm proposed for solving the model defined by equations (3.28-3.27) for the bi-objective single-commodity case is adapted from the tabu search described in section 4.1. However, important considerations should be taken into account, mainly due to the fact that now we are dealing with a bi-objective problem. Moreover, the adapted tabu search for the bi-objective problem will be described below using the ideas described in (Kulturel-Konak, Smith, and Norman, 2006).

Step 0 **Initialization** The initialization of a solution and the tabu structure will be the same as the tabu search for the mono-objective problem mentioned before. Also, initialize the no dominated (ND) solutions' list as empty.

Step 1 **Select the objective** Select one of the two objectives to become active by using a Bernoulli probability mass function. This probability will be variable in each iteration.

Step 2 **Search the neighborhood** The neighborhood N is the same as the one described in section 4.1. The best solution (with respect to the active objective) in N that is not in the tabu structures is chosen. In the case when the best solution of N is in a tabu structure but dominates any solution in the ND solutions list, then, it will be included into the ND set (aspiration criterion).

Step 3 **Update the ND solutions list** Compare each feasible candidate solution with the current ND solutions list as follows: if a candidate solution dominates at least one solution in ND, then remove these dominated solutions from the ND set and add the solution into ND. Also, the candidate solutions that are not dominated by any current solution belonging to ND, must be added to ND.

Step 4 **Update the tabu structures** Add the accepted movement at Step 2 to the corresponding tabu structure and update both tabu structures. The tabu number will be chosen in a random way between 8 and 15.

Step 5 **Diversification** A diversification scheme based on restart is used. If the set of ND solutions has not been updated in the last (*stopping criterion*/4) moves, one of the ND solutions found during the search is uniformly randomly selected as the new current solution. Both tabu structures are reset to empty, and the search restarts from the selected solution (that is, return to Step 1).

Step 6 **Stopping criterion** While the stopping criterion is not satisfied, return to Step 1. In this algorithm, this criterion is defined as the maximum number of iterations conducted without updating ND or an stopping criterion of time.

4.3 A tabu search algorithm for the bi-level bi-objective multi-commodity problem

Exploiting the ideas of the tabu search algorithms proposed in sections 4.1 and 4.2, an adaptation for solving the bi-level bi-objective multi-commodity case is made.

The solution encoding for the tabu search designed to solve the original problem is almost the same than the one mentioned before in sections 4.1 and 4.2. The only difference is that for a solution of the bi-level bi-objective multi-commodity case only the leader's decision variables are explicit or implicitly included in the solution encoding, that is, the variables x, y and z.

In order to evaluate the fitness of each solution, the follower's decision variables are needed. So, the lower level defined by equations 3.13-3.17 is solved by an optimizer. After the lower level has been solved, an infeasibility test is conducted. Therefore, for any infeasible solution, the same idea of the tabu search algorithms designed for the single-commodity case will be used. Once a feasible solution is obtained, the evaluation of the corresponding leader's objective function is done and the tabu search is performed. It is important to emphasize that despite the procedure of this algorithm remains almost the same as the previously described tabu search algorithms, the lower level problem is needed to be solved after each exploration of the solutions within the revised neighborhoods in each iteration of the algorithm. Under this scheme,the steps 1 and 3-6 are exactly the same. A flow chart of this algorithm is depicted in figure 4.2 and 4.3.

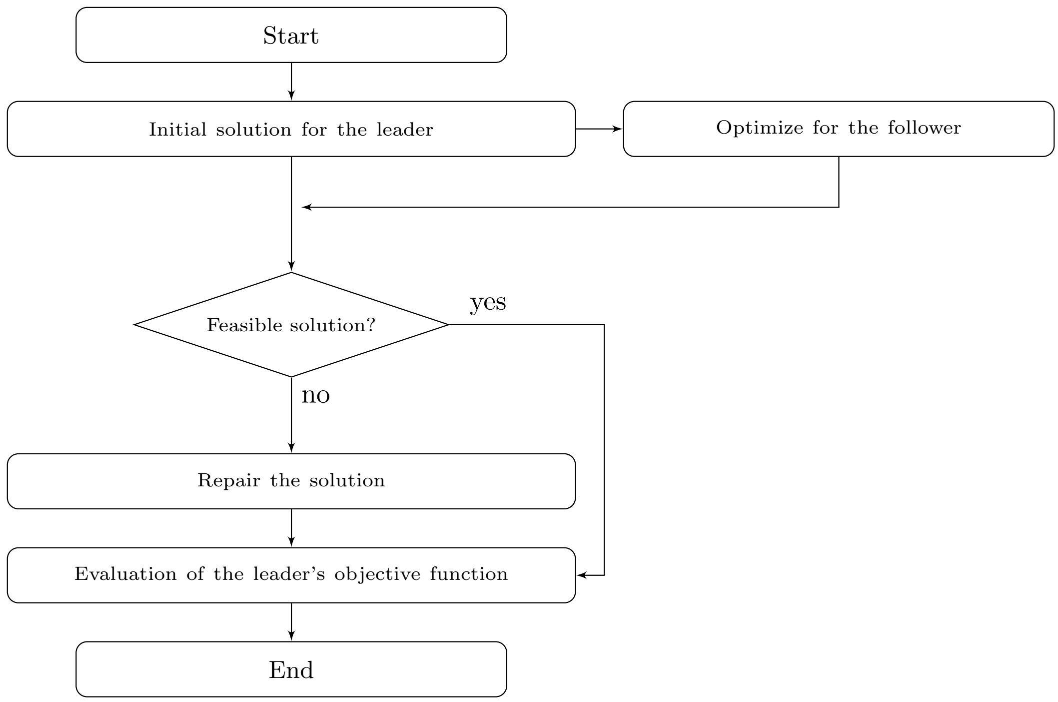


FIGURE 4.2: Construction of a feasible solutions

4.4 A co-evolutionary algorithm based on a biased random keys for the bi-level bi-objective multi-commodity problem

In this section, a co-evolutionary algorithm that considers biased random keys (BRK) is described. The aim is to apply it for solving the bi-level bi-objective multi-commodity problem and to compare its performance against the tabu search algorithm. First, let us make a brief introduction about the general ideas of this algorithm.

BRK are used to solve combinatorial optimization problems, see (Gonçalves and Resende, 2011), in which an explanation about its implementation for solving this kind of problems is presented. It has also been used for solving more specific problems with good results; for example, a routing and wavelength assignment problem in (Noronha, Resende, and Ribeiro, 2011) and the family traveling salesperson problem in (Morán-Mirabal, González-Velarde, and Resende, 2014). On the other hand, the co-evolutionary approach is commonly used for handling problems with various objective functions -nested or simultaneously- just as multi-objective or multi-level problems, see (Sakawa and Nishizaki, 2002), (Oduguwa and Roy, 2002), (Legillon, Liefooghe, and Talbi, 2012), (Yin, Lyu, and Chuang, 2016),& (Dorronsoro et al., 2013).

Solution encoding and decoding

The random key consists in $2|I|+1$ chromosomes. The first $|I|$ chromosomes represent the customers, the following $|I|$ components are associated with the vehicles and the last chromosome indicates the total number of customers included in the solution. Considering the same example described in section 4.1, the corresponding

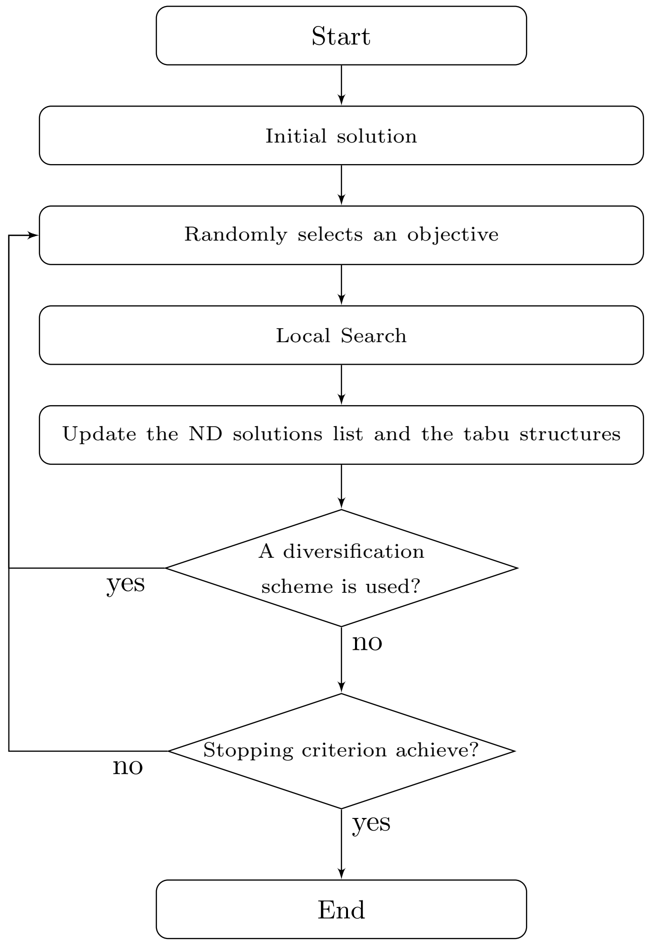


FIGURE 4.3: Tabu search algorithm

BRK will be of size 21 as in figure 4.4. In order to have a solution to the problem under study, the BRK is decoded in the following steps:

1. **Sort and select the customers**
   Sort the first $|I|$ chromosomes in an increasing order and select the first k-th customers, k is the number of the chromosome $2|I| + 1$ times $|I|$, also that means, that the selected customers will be in the contract, as in figure 4.4, that the chromosome $2|I| + 1$ is 0.61 and $|I| = 10$ so the customers that will be in the contract are 6, and the other will not be assigned.

2. **Assign to the vehicle**
   To assign the selected customers to the vehicle, the interval $[0, 1]$ will be divided in $\sum_{l \in L} |V(l)|$ intervals, corresponding to each vehicle, so if the value of the chromosome is in some of those intervals that means that the customers will be in the corresponding vehicle.
   In the example presented in figure 4.4, we have 5 vehicles so the customer 2 is assigned to the fist vehicle because the value of the corresponding chromosome is 0.05 and it is in the first interval. This procedure is done for all the customers that are in the contract.

0.09	0.50	0.56	0.18	0.42	0.94	0.88	0.18	0.96	0.57
0.88	0.05	0.80	0.40	0.44	0.92	0.77	0.01	0.03	0.09
0.61									

Customers	6	7	9	10	2	8	4	5	1	3
Vehicles	0	0	0	0	1	1	2	3	5	5

FIGURE 4.4: Example of a random key

Initial population

In this part, a predefined number of random keys are generated as mentioned above. To compute the corresponding objective function values, the random key should be decoded and repair in case of infeasibility, as the same that in the tabu search.

Selection

In this part, the no dominated solutions are chosen. Also according to the active objective function each population is partitioned in elite (the best solutions) and no-elite. All the elite solutions are passed to the next generation.

Crossover and mutation

The crossover is done with one solution of the elite and with one of the no-elite population chosen it in a random way, and the procedure is done as in (Morán-Mirabal, González-Velarde, and Resende, 2014). That means, to determined each chromosome of the new solution, this inherits with probability $p_e > 0.5$ of the elite solution and $1 - p_e$ of the no-elite solution, see figure 4.5. Also the mutation is done creating a new randomly generated keys.

Co-evolution operator

This operator consists on interchanging some solutions between both population, sending a number of the best solutions of the first population to the second one, to be added to the population 2 to be selected, and vice versa, see Figure 4.6.

4.5 Using path relinking improving the Pareto frontier of the bi-level bi-objective multi-commodity problem

In this section, the proposed path relinking algorithm is detailed. The aim of this algorithm is to improve the Pareto frontier obtained by both, the tabu search and the co-evolutionary algorithm, for the bi-level bi-objective multi-commodity problem. The algorithm described below is an adjustment for the path relinking mentioned in (Ho and Gendreau, 2006).

Moreover, the most important difference between the path relinking proposed in (Ho and Gendreau, 2006) and our proposal is that for the adjustment we need to

Elite solution

0.09	0.50	0.56	0.18	0.42	0.94	0.88	0.18	0.96	0.57
0.88	0.05	0.80	0.40	0.44	0.92	0.77	0.01	0.03	0.09
0.61									

Select with probability $p_e > 0.5$ each chromosome

No-elite solution

0.71	0.82	0.77	0.32	0.97	0.24	0.33	0.57	0.24	0.58
0.81	0.36	0.30	0.11	0.83	0.09	0.75	0.28	0.35	0.48
0.28									

Combined solution

0.71	0.50	0.56	0.32	0.97	0.24	0.88	0.57	0.96	0.57
0.88	0.05	0.80	0.40	0.83	0.09	0.75	0.01	0.03	0.09
0.28									

FIGURE 4.5: Crossover of the random key

obtain a no dominated solutions set instead of a unique solution. In Figure 4.7,the flowchart of the path relinking algorithm is presented.

Step 0 **Initialization** Initialize the no dominated solutions set R_{ND} as the union of the no dominated solutions obtained by the tabu search and by the co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem.

Step 1 **Select the solutions** Select the initial solution s_i and the guiding solutions s_g in a random way from the no dominated solutions set R_{ND}.

Step 2 **Determine the matching of the routes** After selecting the two solutions, we have to establish the matching of the routes. That is, to match up the routes with more similar customers. Moreover, the greedy approach in (Ho and Gendreau, 2006) is used to solve this problem. The matching procedure consists in associate weights to the pair of routes. The weights represent the number of identical costumers between the routes of the two solutions.
This weights only will be considered for routes of the same vehicle type among the two solutions. Later, the pair of routes with largest weight (k, l) is chosen and the route k of the initial solution is assigned to route l of the guiding solution. This is done until all the routes of the initial solution are assigned to the routes of the guiding solution.

Step 3 **Establish the current solution** Set the current solution x, which is the solution obtained by matching the routes between the initial solution s_i and the guiding solution s_g.

Step 4 **Select a ND solution** Select a no dominated solution $\bar{x}$ from the neighborhood of the current solution. This neighborhood is the union of two neighborhoods. *Neighborhood 1* $(N_1(x))$ consists in move a costumer i from its current route of the current solution to which it belongs in the guiding solution. *Neighborhood 2* $(N_2(x))$ consists in interchange customers between two routes, taking into

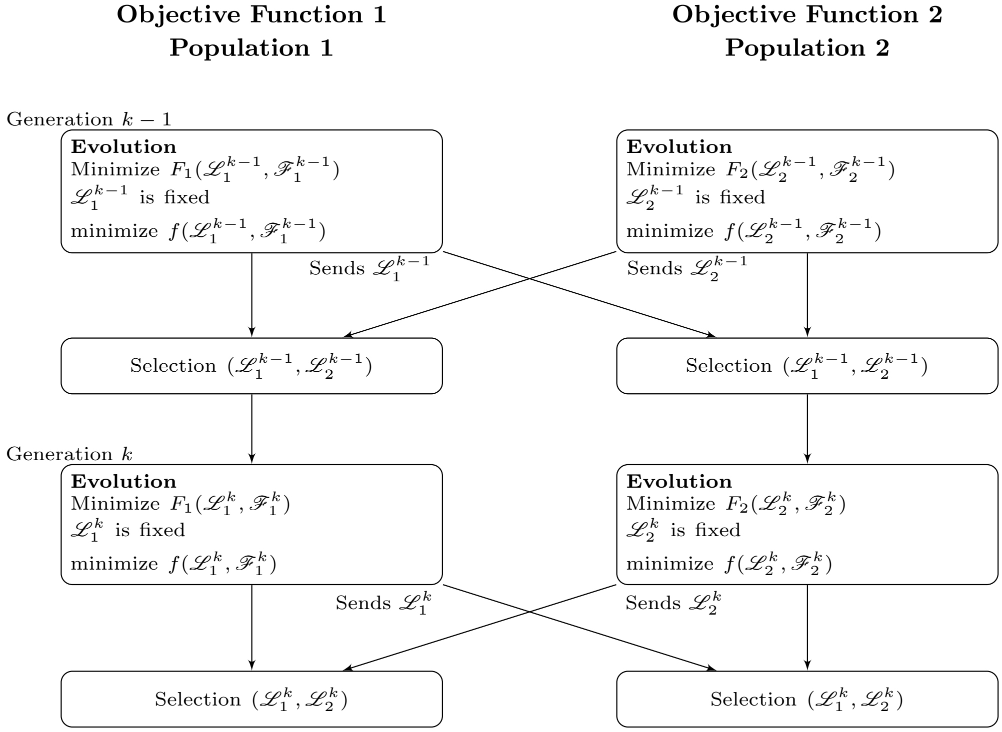


FIGURE 4.6: Co-evolutionary algorithm

account the positions of the customers in the guiding solution. If a movement is associated with an infeasible solution, this move will not be take into consideration for the Neighborhood.

Step 5 **Update the ND solution set** Compare each solution $\acute{x} \in N$ with the no dominated solutions set as follows: if a solution $\acute{x}$ dominates at least one solution in R_{ND}, removes the dominated solution and add the solution $\acute{x}$ into R_{ND}. Also, if the solution $\acute{x}$ is no dominated, it should be added into R_{ND}

Step 6 **Establish the new current solution** Set the current solution x as the no dominated solution $\bar{x}$ obtained in the Step 4.

Step 7 **Achieve the guiding solution** While the current solution is different of the guiding solution, return to Step 4.

Step 8 **Stopping criterion** While the stopping criterion is not satisfied, return to Step 1. In this algorithm, this criterion is defined as a maximum number of iterations.

4.6 Performance measures for bi objective problems

The performance measures used to evaluated the results of the algorithms described above are presented in this section.

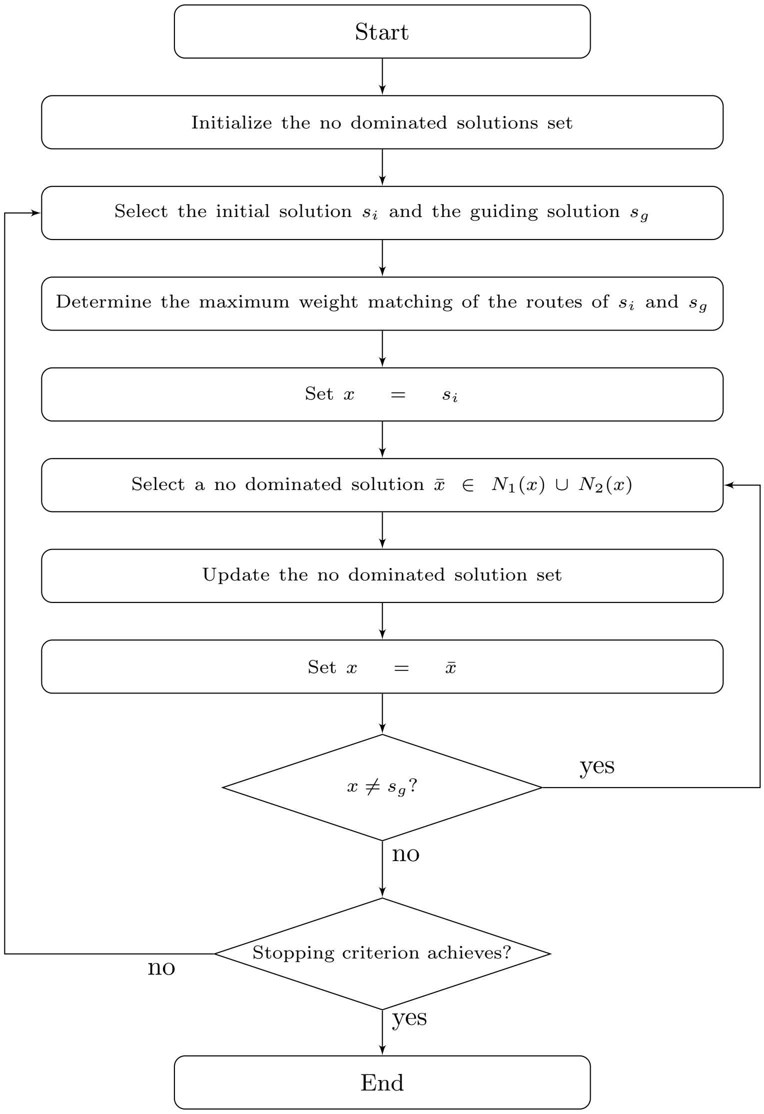


FIGURE 4.7: Path relinking for improving the Pareto frontier

The four measures methods detailed below are those described in (Martí, González-Velarde, and Duarte, 2009).

- **Number of points** This method consists in the average number of efficient points obtained by each algorithm. The efficiency of the algorithm increase with more no-dominated points.
- **The size of the space covered (SSC)** This metric first proposed in (Zitzler and Thiele, 1999) consists to obtained the portion of the objective space that corresponds to the dominated points. Hence, the larger the size of the space covered, the algorithm will have a better performance.
- $k-$**distance** This method is a density estimation technique that calculate the distance from a efficient point to the kth nearest efficient points. We use the same k number proposed as in (Martí, González-Velarde, and Duarte, 2009), that is equal to 5. Furthermore, the smaller the $k-$distance value a better approximation in terms of frontier density.

- **The coverage of two sets measure $\mathbf{C}(\mathbf{A}, \mathbf{B})$** This metric represents the portion of efficient points in a frontier B that is dominated by the efficient point in a frontier A

Chapter 5

Computational experimentation

This chapter can be composed by 2 main sections. First, in order to validate the development of the algorithms mentioned in chapter 4, the instances to use are detailed. Second, the results of the tabu search algorithm for the mono-objective single-commodity problem are presented. Then, the results of the tabu search algorithm for the bi-objective single-commodity are shown. Later, a comparative of the results for the single level problem is given. After that, the results of the tabu search and the co-evolutionary algorithm for the bi-objective multi-commodity problem are presented. Finally, the results of the path relinking algorithm for solving the bi-level bi-objective multi-commodity problem is shown.

Furthermore, the experimentation was conducted on 3.60 GHz Intel Core i7-4790 with 32GB RAM running under Windows 8.1 Pro operative system. These algorithms were implemented on Visual Studio Express 2012 with C++. For the bi-level bi-objective problem, the lower level was solved with CPLEX 12.6.1.

5.1 Description of the instances

A set of instances were adapted from (Cordeau et al., 2002), in order to realize the computational experimentation of the mono-objective single-commodity problem and the bi-objective single-commodity. Furthermore, for the analysis of the bi-level bi-objective multi-commodity problem, this set of instances were readjusted in order to have multiple commodities in each instance. The Split Deliveries VRP, were the instances to adapt, due to the instances have different vehicle types. From these instances the demand, service time and location of the customers, vehicle types and the among of them, as well as, the maximum time and capacity of the route were used.

The set of instances consists in 12 problems, where the customers vary between 48 and 1008; the number of vehicles between 4 and 6; the number of each type of vehicle between 1 and 21; and the facilities between 3 and 7.

In order to generate the data of the facilities, a similar process that the mentioned in (Calvete, Galé, and Oliveros, 2011) was realized. That is, the number of facilities were generated between 3 and 8 in the square $[-200, 200] \times [-200, 200]$. The acquirement cost of one product from each facility (C^1) was selected in a random way from the interval $[0.5, 1.5]$, the production and shipping costs (C^2) were generated in a random way from $[2, 5] + 0.5 * d(facility, depot)$. Furthermore, the maximum availability of production of each facility varies between $[\frac{totaldemand}{numberoffacilities}, totaldemand]$. The constants are considered as $c_l = 1$ and $\varphi = 1$. This procedure was done for the single-commodity problems.

The parameters that define the size of an instance are shown in table (5.1), where the $2nd$ column represents the number of customers,the $3rd$ the number of vehicle types, the $4th$ the number of facilities and the $5th$ the number of vehicles of each type.

Instances	Customer	Vehicle type	Facility	# vehicle
Instance_01	48	4	4	1
Instance_02	96	4	3	2
Instance_03	144	4	4	3
Instance_04	192	4	3	4
Instance_05	240	4	3	5
Instance_06	288	4	4	6
Instance_07	72	6	3	1
Instance_08	144	6	4	2
Instance_09	216	6	7	3
Instance_10	288	6	5	4
Instance_11	1008	4	4	21
Instance_12	720	6	3	10

TABLE 5.1: Table of instances

In the case of the bi-level bi-objective multi-commodity the above instances were adjusted. The number of commodities vary between 2 and 4, the customers demand was divided in a random way in N parts, such that the customers demand of each product were an integer and the sum of them were equal to the costumers demand of the original instances. The emissions (E), the costs (C^1) and (C^2) were divided in N parts in a random way, such that the sum of the parts were equal to the corresponding value of emissions or cost (C^1 or C^2), respectively. Moreover, the profit (g) was established in an interval between 5 and 10. Finally, the consumption required was fixed in a random way between 1 and 2.

5.2 Results

In this section the results for the mono-objective single-commodity problem, the bi-objective single-commodity problem and the bi-level bi-objective multi-commodity problem are given. Also, a comparative between the two first cases is done. Finally, an analysis for the bi-level bi-objective multi-commodity problem is realized.

5.2.1 Mono-objective single-commodity

For the mono-objective single-commodity problem we established two emission levels, the first one is the high level. This level allows to all the customers to be in the contract, it is different for each instance. Once the high level was established, the medium level is calculated to be fixed, also for each instance, in these cases the medium level will be the 60% of the high level. The levels for each instance are shown in table (5.2), where the first column represents the label of the instance, the second column the high level for each instance and the last column the middle level for each instance.

Instances	High level	Middle level
Instance_01	600,000.00	360,000.00
Instance_02	1,320,000.00	792,000.00
Instance_03	1,500,000.00	900,000.00
Instance_04	1,700,000.00	1,020,000.00
Instance_05	2,300,000.00	1,380,000.00
Instance_06	3,000,000.00	1,800,000.00
Instance_07	1,400,000.00	840,000.00
Instance_08	2,000,000.00	1,200,000.00
Instance_09	2,600,000.00	1,560,000.00
Instance_10	3,300,000.00	1,980,000.00
Instance_11	3,600,000.00	2,160,000.00
Instance_12	3,200,000.00	1,920,000.00

TABLE 5.2: Emission levels for the instances

5.2.1.1 Solving with an optimizer (CPLEX)

First the mono-objective problem is solved with an optimizer and with a stopping criterion of three hours, we can observe that for the first three instances (the smallest ones) the optimal solution is not found (see table 5.3). The values of the profit are the best solution found them in the relaxed problem, i.e. without the constraints of subtours, and the GAP is obtained with the optimizer. Also this GAP is for the relaxed problem. The emission value is the one obtained with the solution of the relaxed problem. The relaxed problem consists in start solving the original problem without the constraints of subtours and after CPLEX finishes to solve we add the respective constraint of subtours obtained in the solution of CPLEX, and restart solving the problem. Due to the optimal solutions for the smallest instances are not found, we concluded that it is not from our interest to test larger instances due to the poor initial results.

5.2.1.2 Solving with a tabu search

In consequence of an optimizer cannot find optimal solutions, a tabu search described in section (4.1) is proposed in order to solve the mono-objective problem. For each instance 10 replications were done, due to the randomness of the algorithm. Hence, the parameters established were:

- Stop criterion $= 100$ iterations without improvement.

In table 5.4 and 5.5 the results of the mono-objective single-commodity problem with the middle and high level, respectively are presented. The first column "Instances" represents the name or number of each instance. The second column shows the time (in seconds) required to solve the mono-objective single-commodity problem, this time is the mean of the 10 values obtained. The third column "Mean (Profit)" indicates the mean of the 10 best values obtained with the objective function (3.21). The fourth column "Best Value (Profit)" indicates the best value of the best values

	Medium level (60%)			
	Time (s)	Profit	Emission	GAP %
Instance_01	10,800	4,017.36	359,208	0.11%
Instance_02	10,800	-	-	0.24%
Instance_03	10,800	12,258.90	899961	0.18%
	High level			
	Time (s)	Profit	Emission	GAP %
Instance_01	10,917	4,346.16	445,845	0.08%
Instance_02	10,880	9,190.95	1,090,000	1.05%
Instance_03	10,817	14,306.90	1,250,000	0.47%

TABLE 5.3: Results of CPLEX for the mono-objective single-commodity problem

found them in the 10 runs. The fifth column "GAP %" represents the gaps calculated as: $GAP = (BVF - BVO)/BVF * 100$ where BVF is the best value found in the 10 runs and the BVO is the mean of the 10 best values obtained. The last column "Emission" represents the biggest amount of CO_2 emission obtained by the best solutions for each run.

Middle level (60 %)					
Instances	Time (s)	Mean (Profit)	Best Value (Profit)	GAP %	Emission
Instance_01	0.01	3,773.84	3,921.10	3.76%	359,117.40
Instance_02	0.05	6,576.18	6,969.21	5.64%	790,320.70
Instance_03	0.22	9,496.71	10,008.60	5.11%	899,554.20
Instance_04	0.53	11,040.61	12,086.20	8.65%	1,019,529.00
Instance_05	1.65	10,908.85	11,404.80	4.35%	1,379,300.00
Instance_06	3.78	12,883.12	13,130.90	1.89%	1,799,004.00
Instance_07	0.02	5,170.82	5,305.10	2.53%	836,922.80
Instance_08	0.20	9,633.15	9,764.73	1.35%	1,199,680.00
Instance_09	1.34	14,017.82	14,662.10	4.39%	1,559,528.00
Instance_10	4.00	12,906.24	13,338.70	3.24%	1,978,593.00
Instance_11	410.09	33,950.36	37,403.20	9.23%	2,158,227.00
Instance_12	186.46	36,500.69	38,041.70	4.05%	1,919,770.00

TABLE 5.4: Results of Tabu Search for the mono-objective single commodity problem with middle level

In table 5.4, we can observe that the variation of the best values obtained is not more than 10%, that means that the performance of the algorithm is *"good"*. Also, we can see that the emission values are to close to the medium level, that means that in order to improve the profit we are going to have more CO_2 emissions. In table 5.5, we can observe that the variation of the best values obtained is not more than 5%. Also,in the medium level, the profit achieves different local optimal solutions because of the gap is worse than the high level, which means that, the performance of the algorithm with a high level is better than with a medium level. Furthermore,

			High level		
Instances	Time (s)	Mean (Profit)	Best Value (Profit)	GAP %	Emission
Instance_01	0.01	4,195.99	4,294.42	2.29%	473,290.80
Instance_02	0.05	9,177.60	9,457.02	2.95%	1,295,121.00
Instance_03	0.24	13,414.95	13,601.10	1.37%	1,448,798.00
Instance_04	0.88	18,911.80	19,561.20	3.32%	1,553,194.00
Instance_05	2.26	23,187.87	24,167.60	4.05%	2,157,262.00
Instance_06	5.20	26,633.34	27,306.60	2.47%	2,852,290.00
Instance_07	0.03	8,117.43	8,180.81	0.77%	1,260,857.00
Instance_08	0.24	13,080.50	13,389.60	2.31%	1,815,438.00
Instance_09	1.71	18,802.08	18,990.20	0.99%	2,485,561.00
Instance_10	5.74	27,633.84	28,248.90	2.18%	3,182,874.00
Instance_11	1,139.03	45,367.56	46,354.90	2.13%	3,505,763.00
Instance_12	320.76	42,534.53	43,007.50	1.10%	2,819,258.00

TABLE 5.5: Results of Tabu Search for the mono-objective single commodity problem with high level

the required time with a high level is more than the required time with a medium level.

5.2.2 Bi-objective single-commodity

In this case, in order to solve the bi-objective single-commodity problem the tabu search described in section (4.2) is established . Due to the randomness of the algorithm 10 replications were done for each instance. Hence, the parameters established were:

- Stop criterion = 100 iterations without improve the ND list or after (3 hrs)
- Probability function is variable between 0.4 and 0.75.

The results are presented in table 5.6, in which, the first column "Time" represents the mean time in seconds. The second and third column "Min" and "Max" represent the minimum and the maximum of the no-dominated solutions in the iteration.

From table 5.6 it can be observed that there is a relation between the size of the instances and the number of no-dominated solutions, while the size of instance is larger, the number of ND solutions increases, except for the instances "Instance_11" and "Instance_12". Furthermore, these two instances stop by the maximum time not by the iterations without improvement. This means, that if the stop criterion of time is larger the ND list could be improved.

In graphics on Appendix A, the results of the best solutions for the mono-objective problem in **(yellow)** and the no-dominated solutions of the bi-objective problem in **(blue)** are presented to compare the performance of both in each instance.

It can be observed that the tabu search algorithm has a good performance for almost all the instance, except for instances "Instance_11" and "Instance_12", where the approximation to the Pareto frontier is not completed, because the nadir points are not found.

		ND Solutions	
	Time(s)	Min	Max
Instance_01	0.97	246	333
Instance_02	61.76	779	1,089
Instance_03	531.45	1,269	1,623
Instance_04	1,193.45	1,259	1,525
Instance_05	4,534.47	1,737	2,179
Instance_06	10,215.01	2,187	2,384
Instance_07	23.15	741	912
Instance_08	543.99	1,191	1,843
Instance_09	4,612.69	1,783	2,547
Instance_10	10,397.62	2,286	2,461
Instance_11	10,800.00	386	689
Instance_12	10,800.00	1,781	2,179

TABLE 5.6: Results of Tabu Search for the bi-objective single commodity problem

5.2.3 Bi-level bi-objective multi-commodity problem

In this case, the tabu search algorithm described in section (4.3) and the co-evolutionary algorithm based on a biased random keys described in section (4.4) are applied to solve the bi-level bi-objective multi-commodity problem.

5.2.3.1 A tabu search algorithm for a bi-level bi-objective multi-commodity problem

The parameters established were:

- Stop criterion $= 100$ iterations without improve the ND list or after (5 hrs)
- Probability function is variable between 0.4 and 0.75.

In table 5.7 the results are presented, in which, the first column "Time" represents the mean time in seconds. The second and third column "Min" and "Max" represent the minimum and the maximum of the no-dominated solutions in the iteration.

From table 5.7 it can be observed that most of the instances stop by the time not by the iterations without improvement. This means, that more ND solutions could be found if the stop criterion of time were larger.

In Figures on Appendix B, the no-dominated solutions of the bi-level bi-objective problem in **(blue)** are presented in order to observe the results of the tabu search algorithm.

It can be observed that the tabu search algorithm has a good performance for instances: "Instance_1", "Instance_2" and "Instance_7", where it can be observe that the Pareto frontier is almost accomplished. Otherwise, for the other instances the approximation to the Pareto frontier is not fulfilled.

		ND Solutions	
	Time(s)	Min	Max
Instance_01	13,780.95	101	231
Instance_02	18,019.72	183	299
Instance_03	18,036.66	159	251
Instance_04	18,061.67	63	143
Instance_05	18,100.24	31	75
Instance_06	18,080.90	27	79
Instance_07	18,010.27	164	374
Instance_08	18,026.57	109	173
Instance_09	18,083.44	65	77
Instance_10	18,129.06	20	43
Instance_11	22,970.92	7	15
Instance_12	20,318.91	2	6

TABLE 5.7: Results of Tabu Search for the bi-level bi-objective multi-commodity problem

5.2.3.2 A co-evolutionary algorithm for a bi-level bi-objective multi-commodity problem

For each instance 10 iterations were done, due to the randomness of the algorithm. Hence, the parameters established were:

- Population = 100
- Generations = 100
- Crossover probability = 0.7
- Mutation probability = 0.1
- Elite population = 0.1

The results are presented in table 5.8 , in which, the first column "Time" represents the mean time in seconds. The second and third column "Min" and "Max" represent the minimum and the maximum of the no-dominated solutions in the iteration.

From table 5.8 it can be observed that the number of no dominated solutions are less than tabu search algorithm solutions but the computational time is less than the tabu.

Furthermore, in figures on Appendix C, the no dominated solutions of the bi-level bi-objective problem in **(blue)** are presented in order to observe the results of the co-evolutionary algorithm. Besides of the few no dominated solutions this algorithm found the nadir point for the emissions and also different no dominated solutions to the tabu search algorithm.

It can be observed that in the co-evolutionary algorithm the approximation to the Pareto frontier is not accomplished, but most of the found solutions are part to the Pareto frontier.

		ND Solutions	
	Time(s)	Min	Max
Instance_01	205.68	18	28
Instance_02	236.22	21	37
Instance_03	283.42	27	43
Instance_04	291.50	8	20
Instance_05	216.27	21	37
Instance_06	396.71	20	32
Instance_07	262.40	19	31
Instance_08	236.76	18	32
Instance_09	456.27	21	46
Instance_10	409.38	23	44
Instance_11	381.31	26	43
Instance_12	533.77	19	30

TABLE 5.8: Results of Co-evolutionary algorithm based on a biased random keys fir the bi-level bi-objective multi-commodity problem

We can observe that the results of the tabu search and co-evolutionary algorithm could be improved and the union of the two no dominated solutions lists could be better than each of them, thus the path relinking algorithm described in section (4.5) is implemented. Furthermore, this algorithm joins the no dominated solutions lists from the previous algorithms.

5.2.3.3 Path relinking algorithm for a bi-level bi-objective multi-commodity problem

The path relinking algorithm described in section (4.5) is performed in order to achieve the Pareto frontier.

The parameter established was:

- Stop criterion = 50 iterations

The results are presented in table 5.9 and in graphics on Appendix D, the merged results of the tabu search and co-evolutionary algorithm for the bi-level bi-objective problem in (blue) and the no-dominated solutions of the path relinking for the bi-level bi-objective problem in (pink) are illustrated to observe how the Pareto frontier is achieved in the path relinking with the previous results.

In table 5.9 the results are given in an similar way than the tables 5.7 and 5.8.

It could be observed from the graphics on Appendix D, that the Pareto frontier is improved with the path relinking except in Instance_01 where the Pareto frontier is approximated with the tabu search and co-evolutionary algorithm. Furthermore the computational time in the path relinking is less than the computational time in the tabu search algorithm.

		ND Solutions	
	Time(s)	Min	Max
Instance_01	2195.89	172	173
Instance_02	2199.41	267	309
Instance_03	1508.09	228	246
Instance_04	2810.65	254	263
Instance_05	2376.56	237	252
Instance_06	2515.58	208	237
Instance_07	4138.02	283	321
Instance_08	3014.12	243	269
Instance_09	1819.07	224	240
Instance_10	3751.65	206	259
Instance_11	17474.90	168	202
Instance_12	4174.78	135	186

TABLE 5.9: Results of Path relinking algorithm for the bi-level bi-objective multi-commodity problem

Chapter 6

Conclusions

In this chapter the conclusions of the analysis done for the simplifications of the problem and the original problem are presented. Furthermore, the possible adaptations and improvement of the algorithms is exposed.

As mentioned before, in this thesis is considered a bi-level bi-objective multi-commodity problem. In the upper level a distribution company is analyzed and in the lower level a manufacturer company is considered. Furthermore, the distribution company aims to maximize its profit and minimize the CO_2 emissions. While the manufacturer company aims to minimize its production costs.

In order to solve the problem, simplifications of the original problem were proposed and analyzed. The first simplification results in a mono-objective single-commodity problem. This problem considered a single objective function (the profit) and the CO_2 emissions are established by emission levels. Also, it considers a single type of commodity. The second simplification is the bi-objective single-commodity problem, in which the production only considers one type of commodity. For both simplifications an specific algorithm that solves the corresponding model in chapters 3 and 4 is proposed.

Furthermore, in the mono-objective single-commodity problem the experimental results show that solving it with an optimizer requires an excessive computational time. In order to obtain results for this problem a tabu search algorithm was proposed. The results of this algorithm show that for an established high level emission the results has a better performance than when a medium level emission is established.

In the bi-objective single-commodity problem the experimental results illustrate "good" Pareto frontiers for almost all the instance (see Appendix A), except for the "Instance_11" and "Instance_12" where the stopping criterion is by the computational time. It could be observed that for "Instance_11", the tabu search for the mono-objective algorithm has better solutions that are no-dominated for the bi-objective single-commodity problem.

The original problem (the bi-level bi-objective multi-commodity problem) is solved via three different algorithms. The first algorithm is a tabu search, the results of this algorithm are illustrated in the Appendix B. It could be observed that for all the instances, the Pareto frontier is not accomplished. Moreover, for the "Instance_11" and "Instance_12" only a few points were found.

The second algorithm is a co-evolutionary based on a biased random keys, the results of this algorithm are illustrated in the Appendix C. It also could be observed that for all the instance the Pareto frontier is not accomplished. But, in this case the point that improves the profit of the distributed company its not found.

After the analysis of the tabu search and the co-evolutionary algorithm we can observe that the two Pareto frontiers could be improved and consequently the path relinking algorithm were proposed. The results of this algorithm are illustrated in the Appendix D. It could be observe that this algorithm really improves the Pareto frontier for all the tested instances. Furthermore, the Pareto frontier are *fulfilled* in all the cases.

6.1 Future research

An extension of this research could use a multithread approach, in which the algorithms may be adapted to simultaneous multithreading, mentioned in (Lenir, Govindarajan, and Nemawarkar, 1992). This approach have the advantage that allows us to solve multiple problems of the lower level, for different leader's solutions in order to reduce the computational time.

Furthermore, this can be applied to the tabu search and the co-evolutionary algorithms for solving the bi-level bi-objective multi-commodity problem. Due to this approach could be tailored to many applications, also it could be adjusted to evaluate different solutions at the same time in all the algorithms mentioned in this thesis.

Moreover, other simplifications of the problem could be given. First, it could be considered a fixed number of customers in the contract. Second, stochastic demands can be allowed or supposed an homogeneous fleet. Also, extensions of the vehicle routing problem could be adapted to this bi-level bi-objective problem, as taking into consideration time windows or multiple depots.

Appendix A

Results obtained for the comparison between mono-objective and bi-objective single-commodity problem

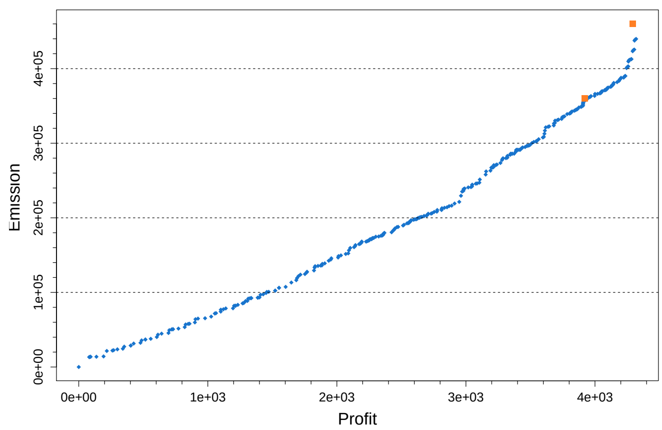


FIGURE A.1: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_1"

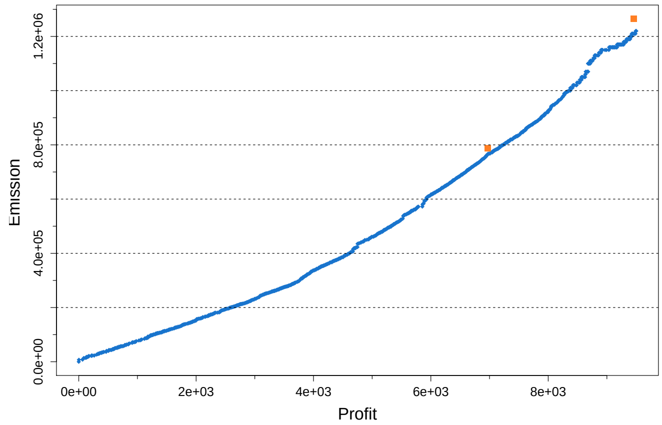


FIGURE A.2: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_2"

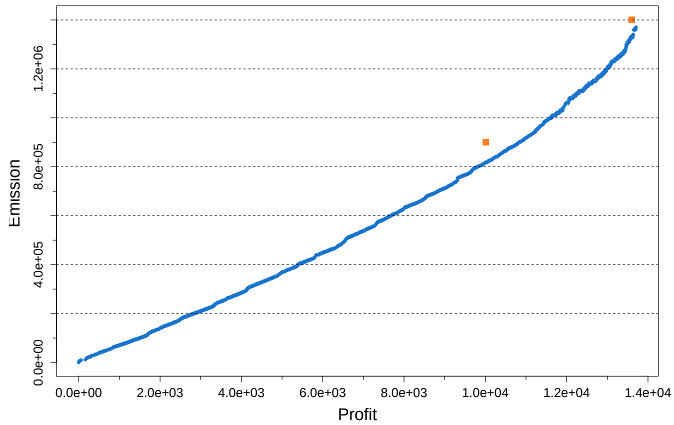


FIGURE A.3: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_3"

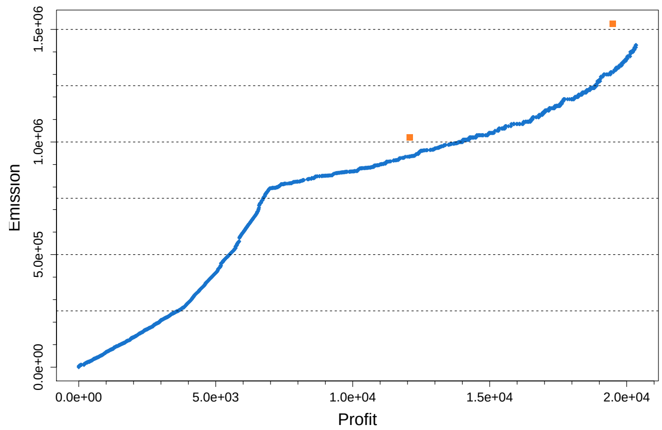


FIGURE A.4: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_4"

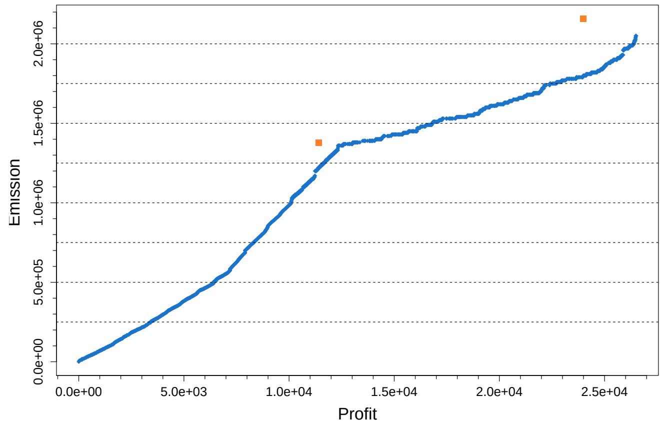


FIGURE A.5: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_5"

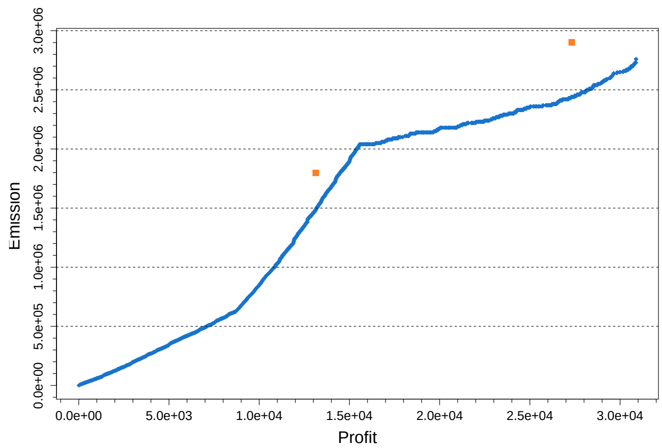


FIGURE A.6: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_6"

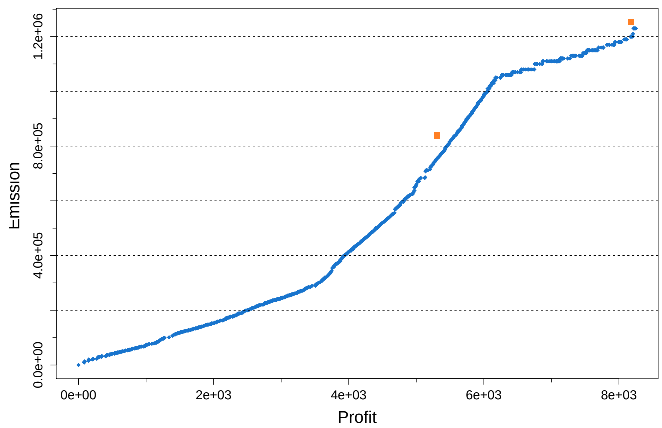


FIGURE A.7: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_7"

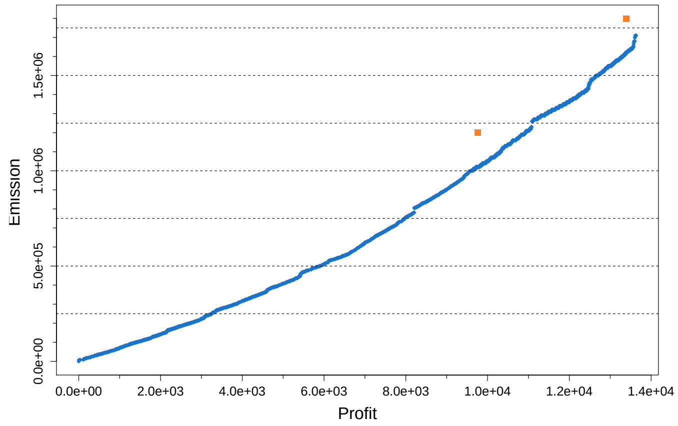


FIGURE A.8: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_8"

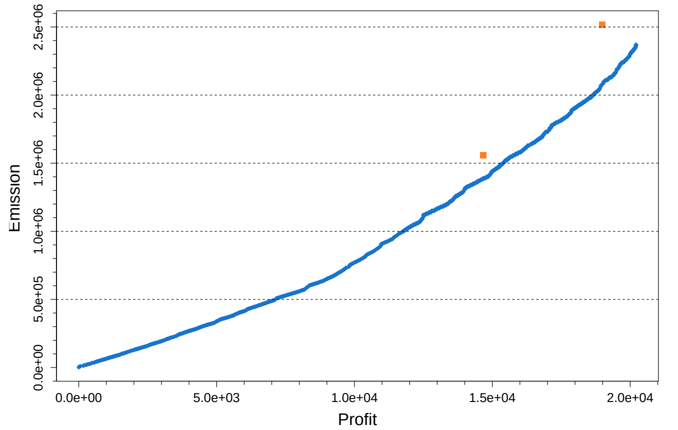


FIGURE A.9: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_9"

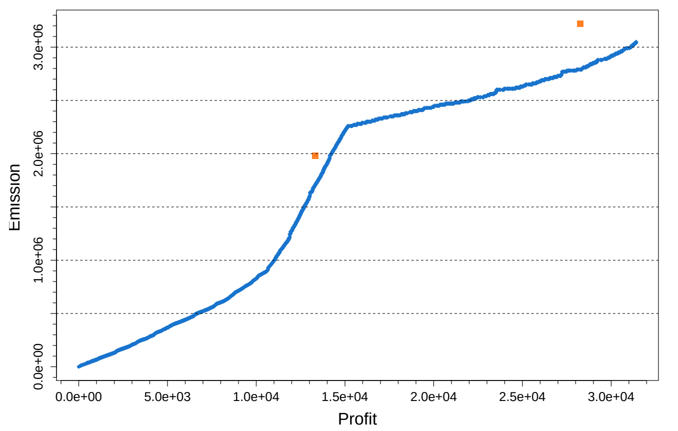


FIGURE A.10: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_10"

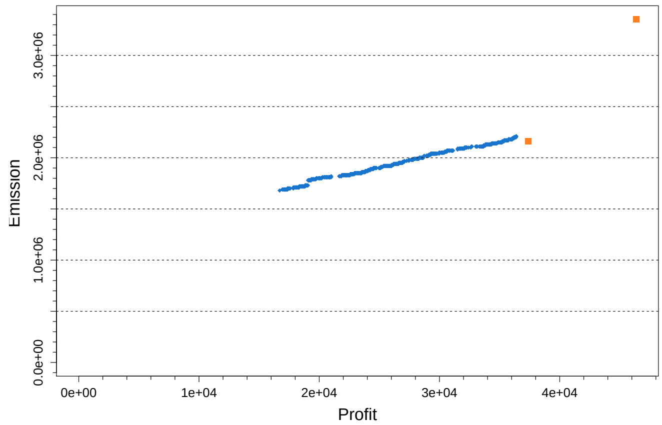


FIGURE A.11: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_11"

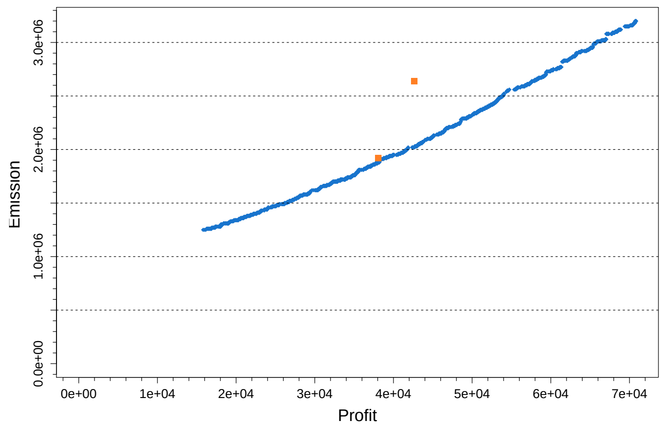


FIGURE A.12: Comparison between mono-objective and bi-objective single-commodity problem of the "Instance_12"

Appendix B

Illustrations of Tabu Search for the bi-level bi-objective multi-commodity problem

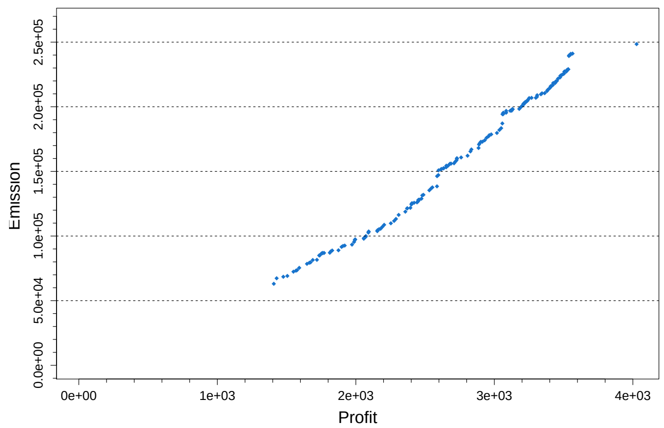


FIGURE B.1: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_1"

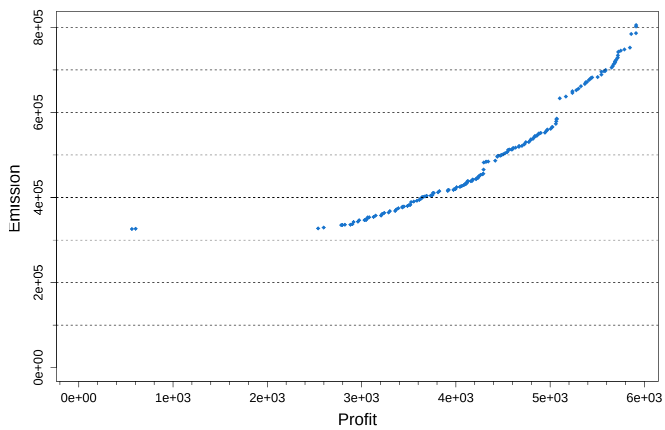


FIGURE B.2: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_2"

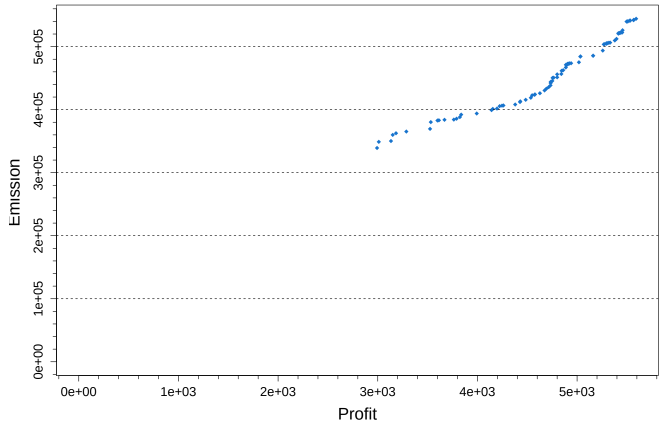


FIGURE B.3: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_3"

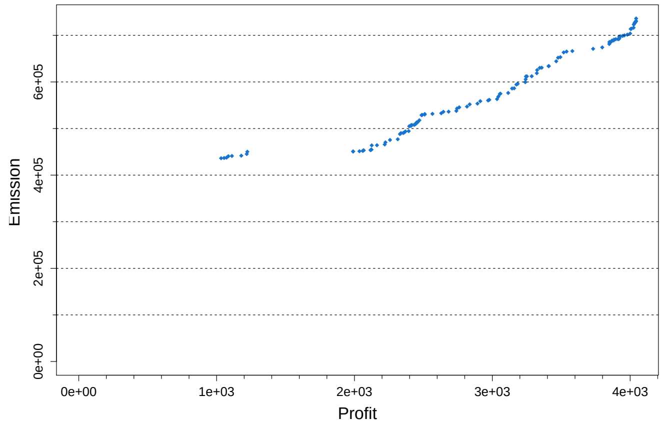


FIGURE B.4: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_4"

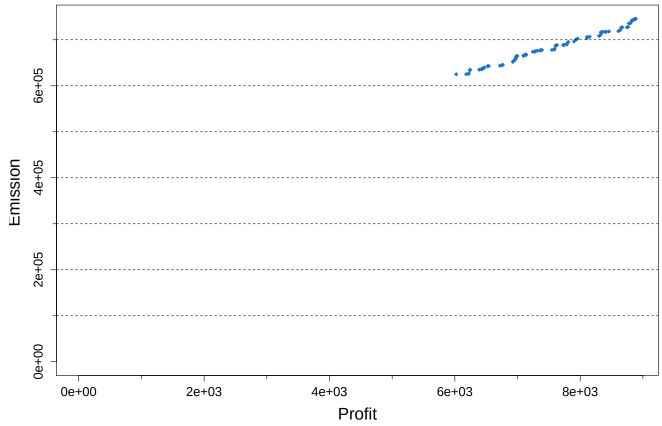


FIGURE B.5: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_5"

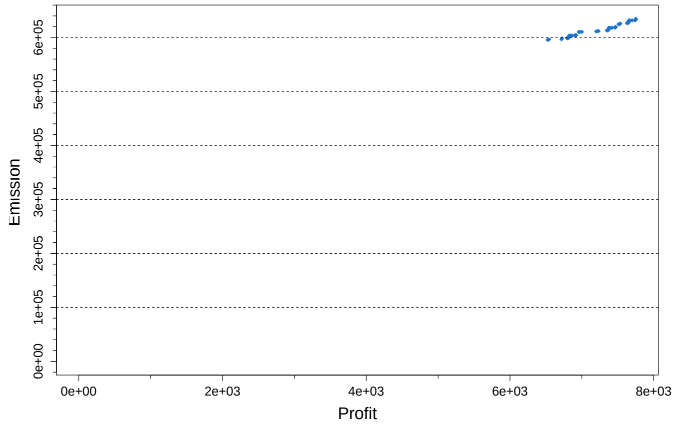


FIGURE B.6: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_6"

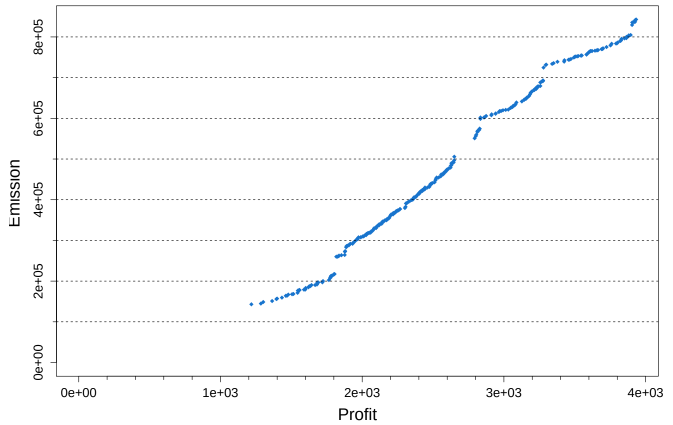


FIGURE B.7: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_7"

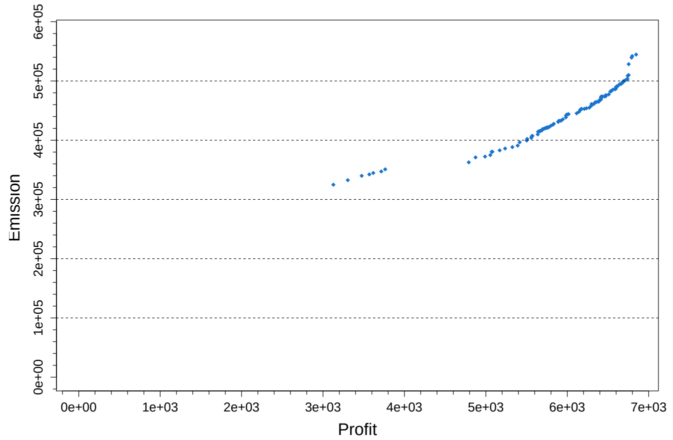


FIGURE B.8: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_8"

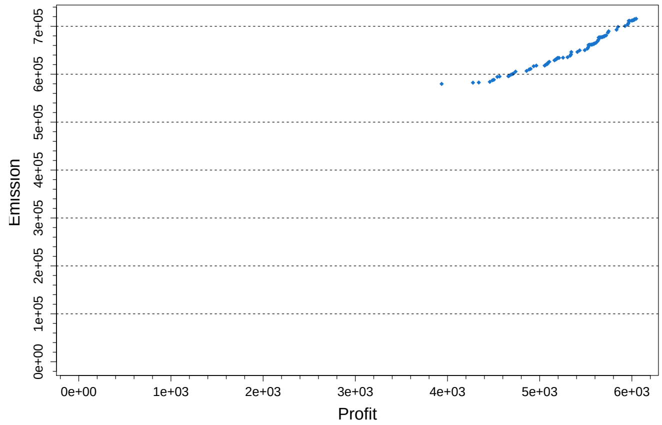


FIGURE B.9: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_9"

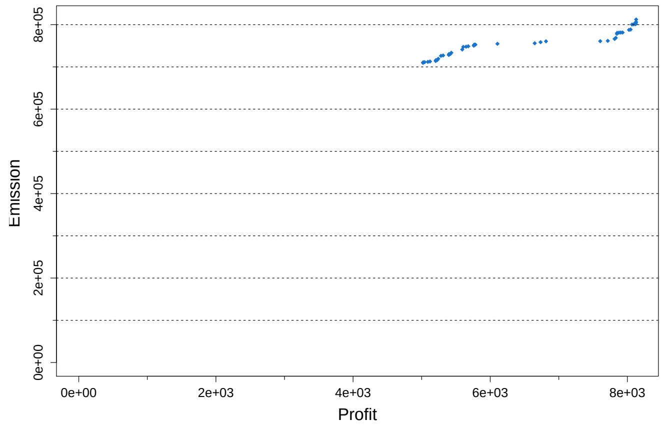


FIGURE B.10: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_10"

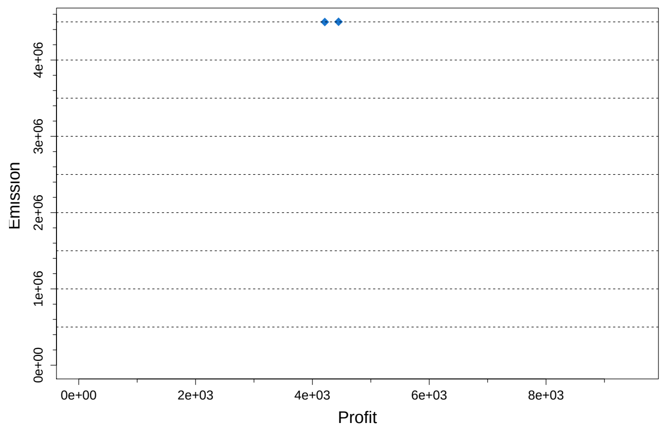


FIGURE B.11: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_11"

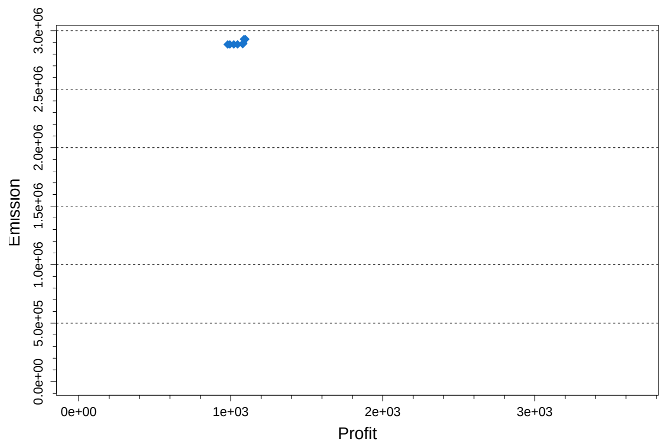


FIGURE B.12: Graphic of Tabu Search for the bi-level bi-objective multi-commodity problem of the "Instance_12"

Appendix C

Graphics of Co-evolutionary algorithm based on a random keys for the bi-level bi-objective multi-commodity problem

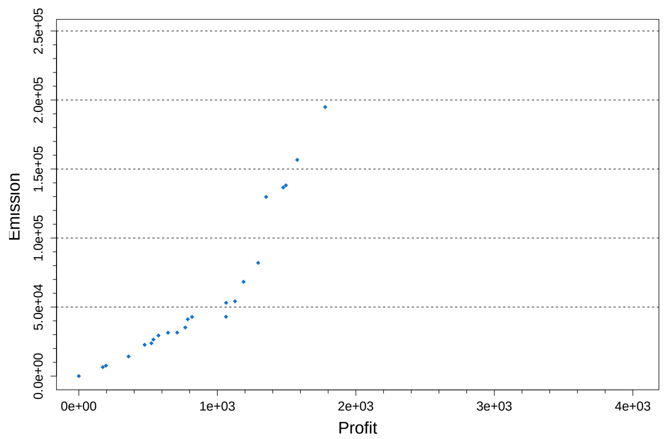


FIGURE C.1: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_1"

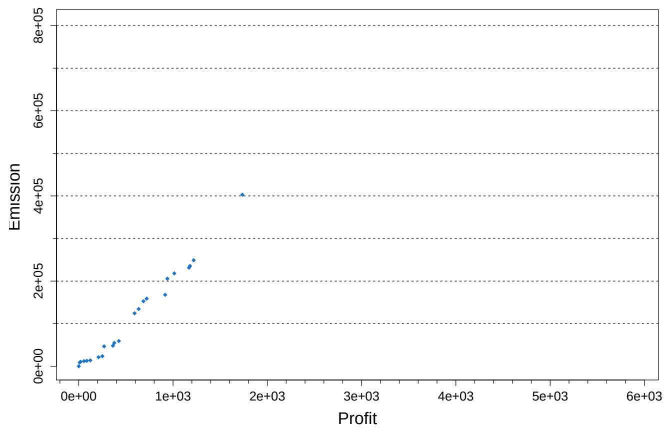


FIGURE C.2: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_2"

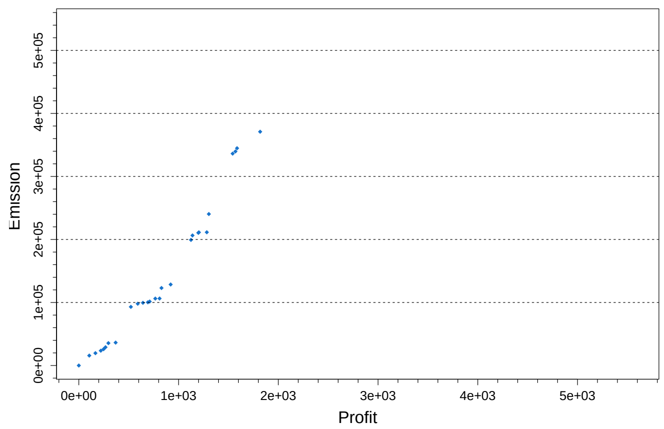


FIGURE C.3: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_3"

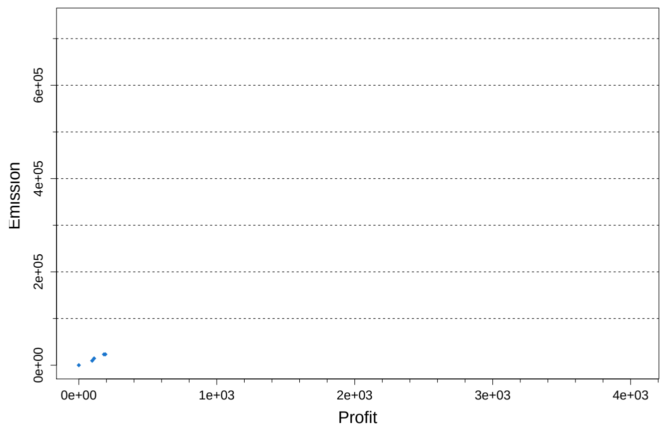


FIGURE C.4: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_4"

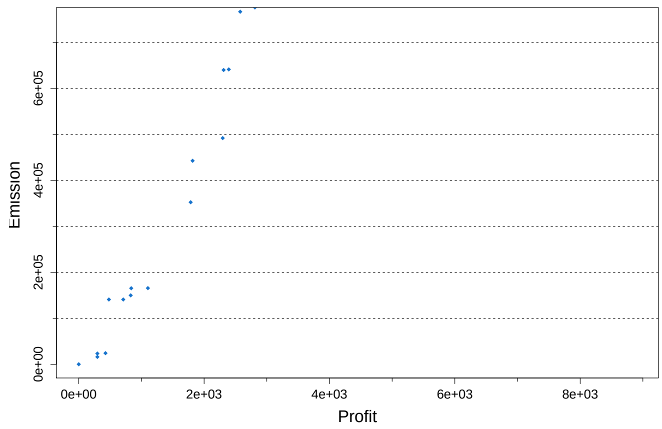


FIGURE C.5: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_5"

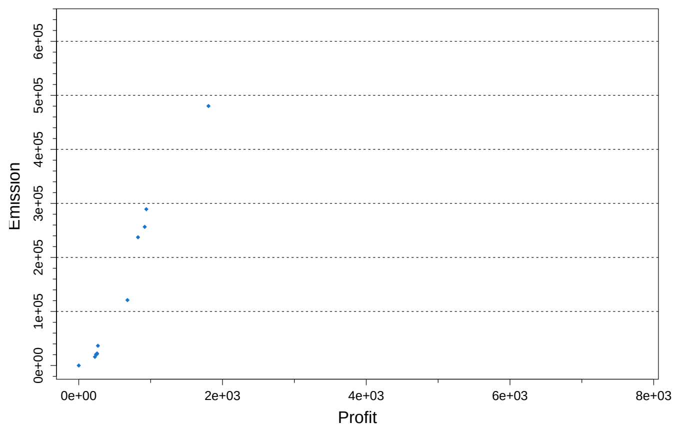


FIGURE C.6: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_6"

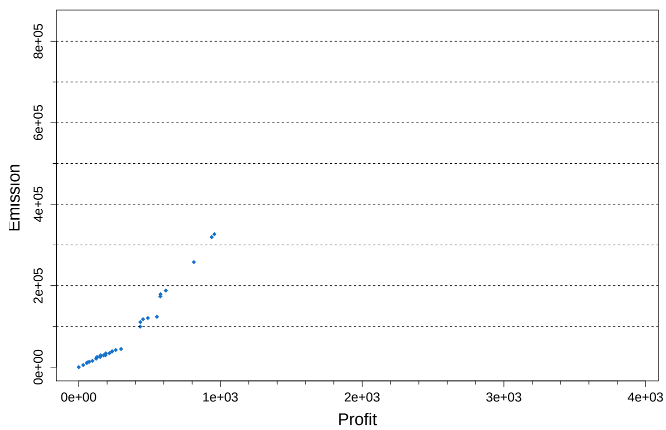


FIGURE C.7: Graphics of Co-evolutionary algorithm the bi-level bi-objective multi-commodity problem of the "Instance_7"

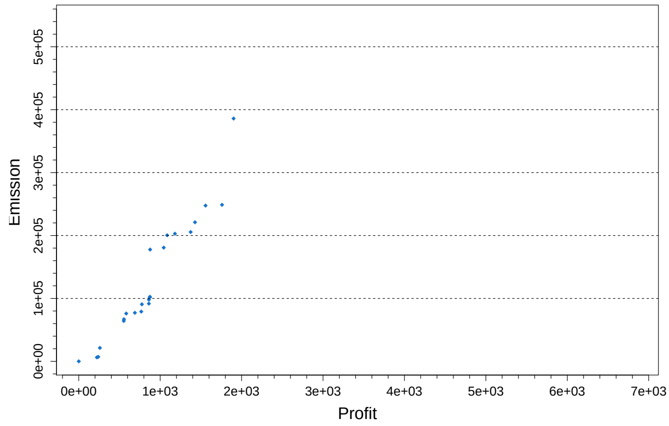


FIGURE C.8: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_8"

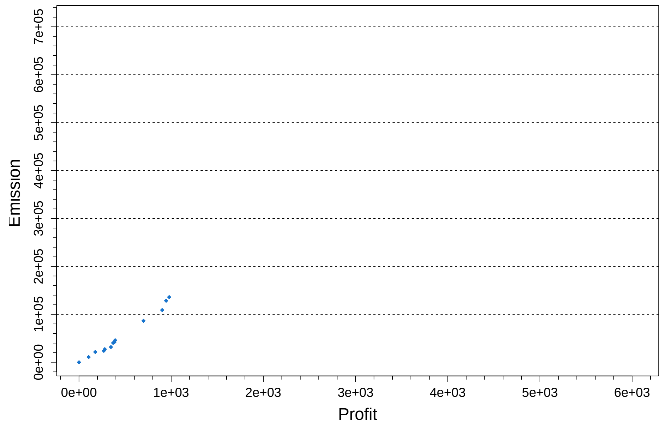


FIGURE C.9: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_9"

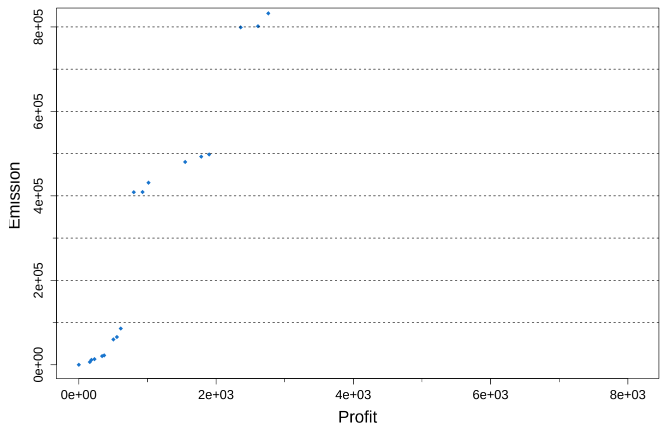


FIGURE C.10: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_10"

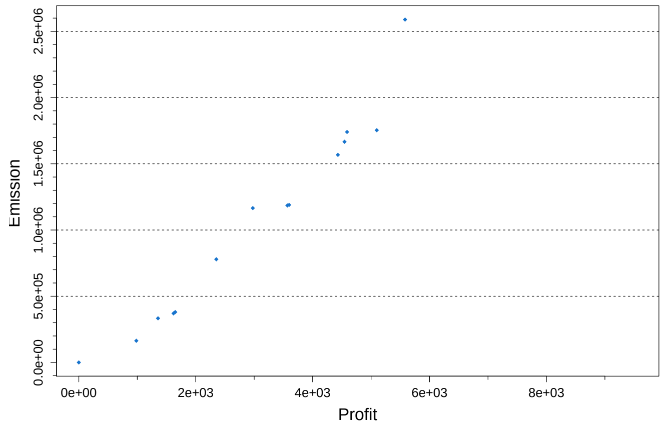


FIGURE C.11: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_11"

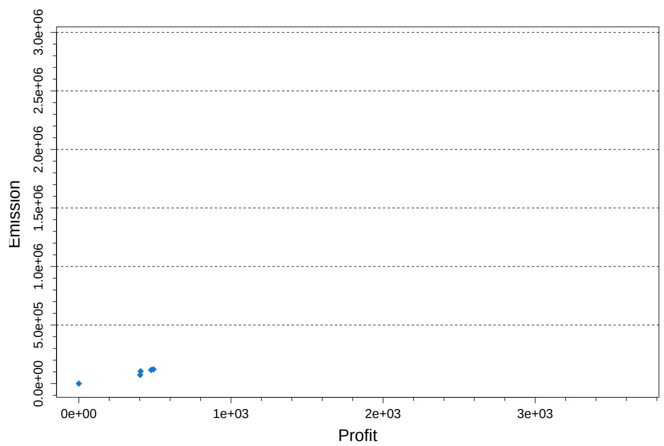


FIGURE C.12: Graphics of Co-evolutionary algorithm for the bi-level bi-objective multi-commodity problem of the "Instance_12"

Appendix D

Analyzing ND solutions lists for the bi-level bi-objective multi-commodity problem

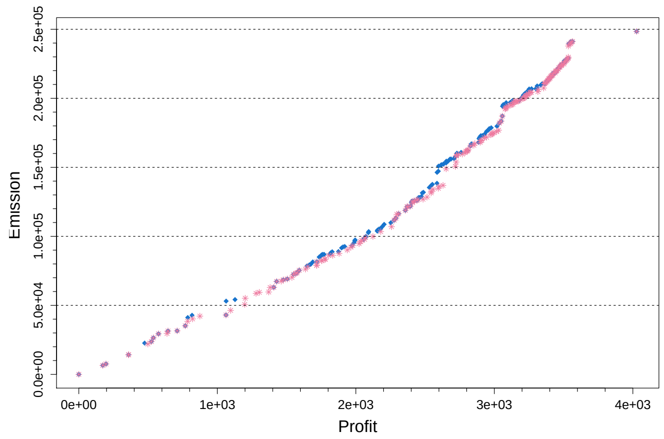


FIGURE D.1: Comparison between the merged ND solutions lists and path relinking of the "Instance_1"

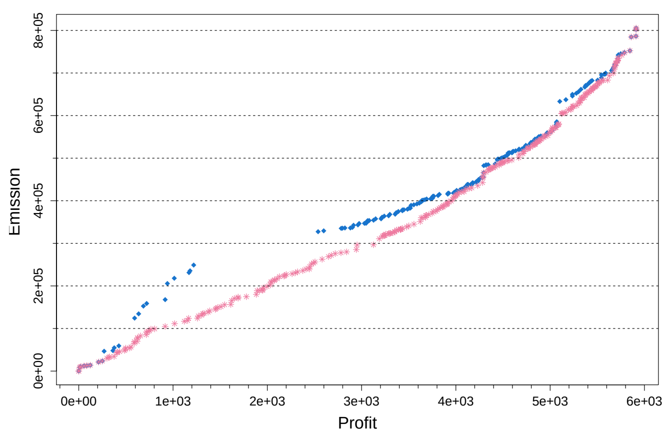


FIGURE D.2: Comparison between the merged ND solutions lists and path relinking of the "Instance_2"

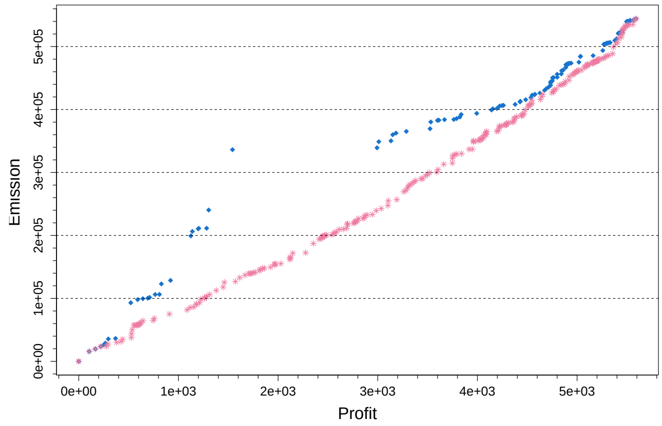


FIGURE D.3: Comparison between the merged ND solutions lists and path relinking of the "Instance_3"

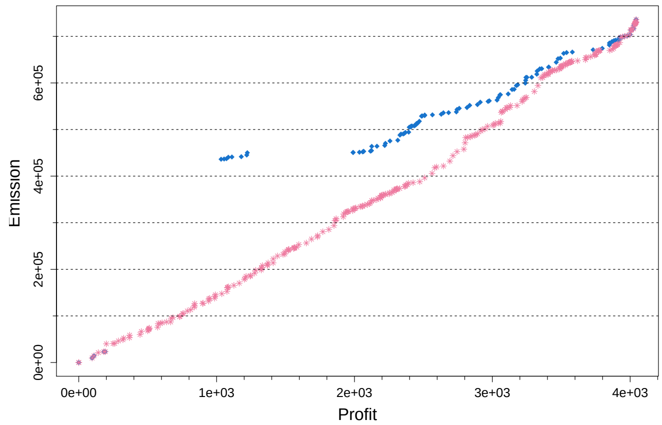


FIGURE D.4: Comparison between the merged ND solutions lists and path relinking of the "Instance_4"

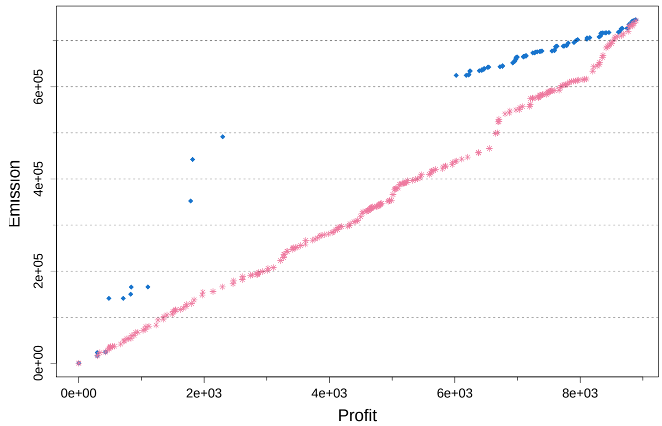


FIGURE D.5: Comparison between the merged ND solutions lists and path relinking of the "Instance_5"

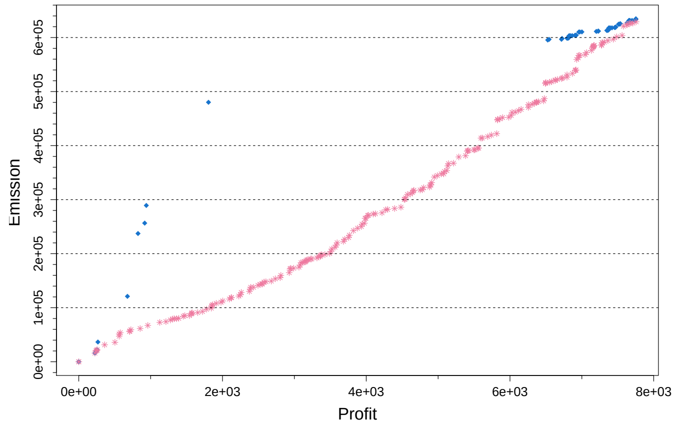


FIGURE D.6: Comparison between the merged ND solutions lists and path relinking of the "Instance_6"

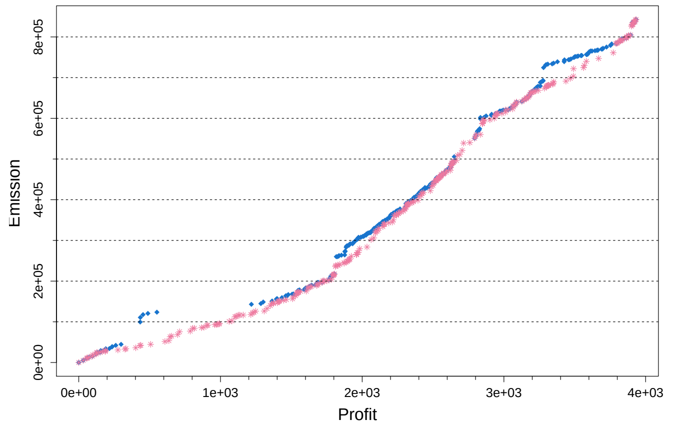


FIGURE D.7: Comparison between the merged ND solutions lists and path relinking of the "Instance_7"

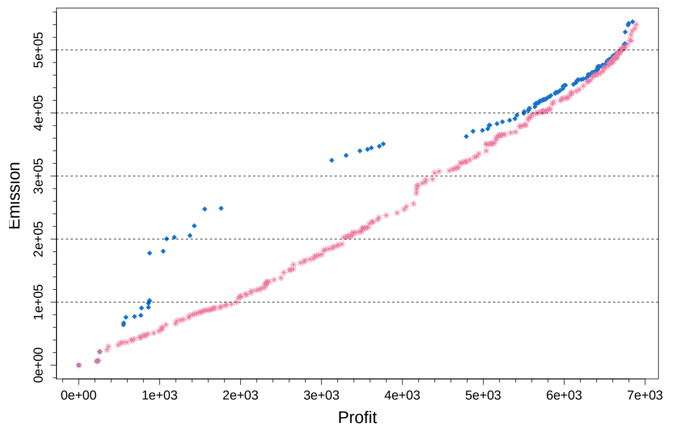


FIGURE D.8: Comparison between the merged ND solutions lists and path relinking of the "Instance_8"

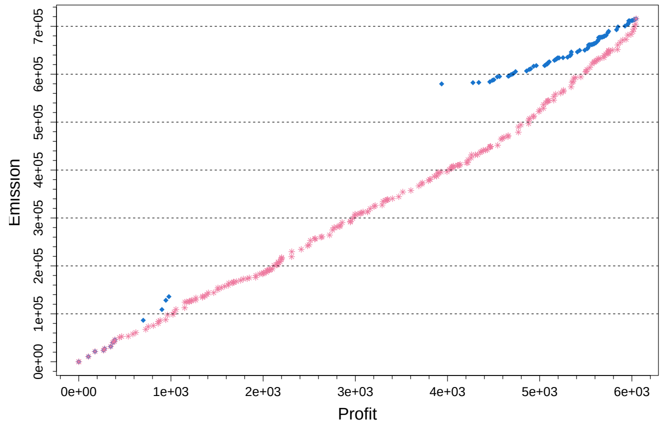


FIGURE D.9: Comparison between the merged ND solutions lists and path relinking of the "Instance_9"

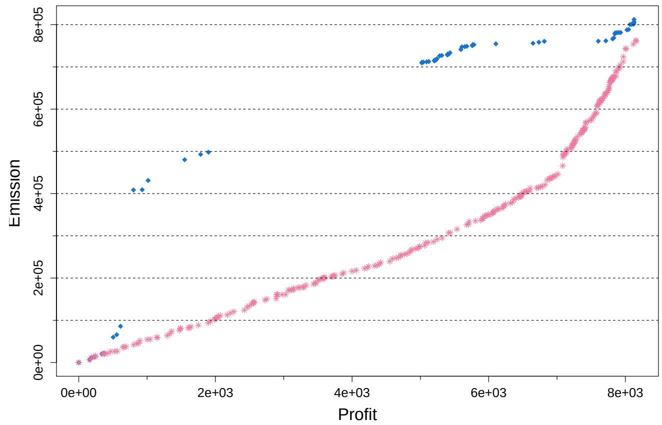


FIGURE D.10: Comparison between the merged ND solutions lists and path relinking of the "Instance_10"

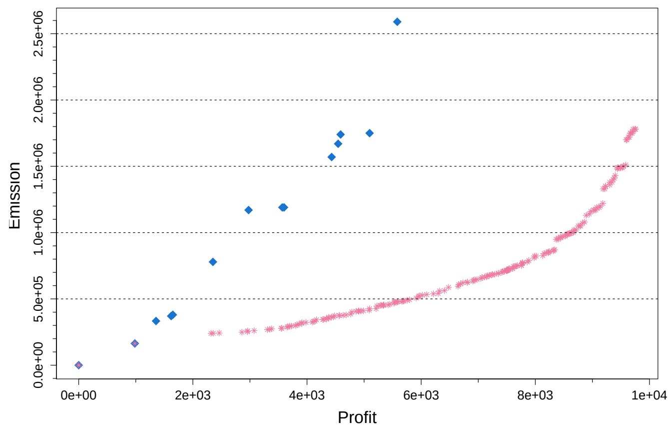


FIGURE D.11: Comparison between the merged ND solutions lists and path relinking of the "Instance_11"

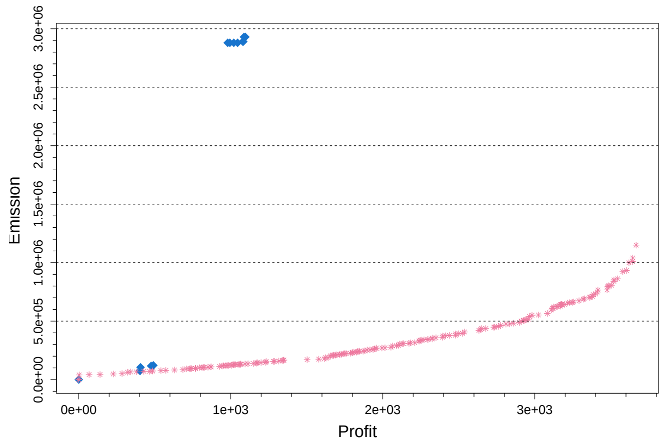


FIGURE D.12: Comparison between the merged ND solutions lists and path relinking of the "Instance_12"

Bibliography

Arora, V. K. et al. (2011). "Carbon emission limits required to satisfy future representative concentration pathways of greenhouse gases". In: *Geophysical Research Letters* 38.5.

Beamon, B. M. (1999). "Designing the green supply chain". In: *Logistics information management* 12.4, pp. 332–342.

Calvete, H.I., C. Galé, and M-J. Oliveros (2011). "Bilevel model for production-distribution planning solved by using ant colony optimization". In: *omputers and Operations Research* 38, pp. 320–327.

Cordeau, J.F. et al. (2002). "A guide to vehicle routing heuristics". In: *Journal of the Operational Research society* 53.5, pp. 512–522.

Dekker, R., J. Bloemhof, and I. Mallidis (2012). "Operations Research for green logistics–An overview of aspects, issues, contributions and challenges". In: *European Journal of Operational Research* 219.3, pp. 671–679.

Dempe, S. (2002). "Foundations of bilevel programming". In: *Springer Science & Business Media*.

Diabat, A. and D. Simchi-Levi (2009). "A carbon-capped supply chain network problem". In: *In 2009 IEEE International Conference on Industrial Engineering and Engineering Management* IEEE, pp. 523–527.

Diario Oficial de la Federación (1971). *Ley Federal para prevenir y controlar el medio ambiente*.

— (1988a). *Ley General del Equilibrio Ecológico y la Protección del Medio Ambiente*. DOF 13-05-2016.

— (1988b). *Reglamento de la Ley General del Equilibrio Ecológico y la protección al ambiente en materia de prevención y control de la contaminación de la atmósfera*. DOF 31-10-2014.

— (1993). *Norma Oficial Mexicana NOM-043-SEMARNAT-1993*.

— (1999a). *Norma Oficial Mexicana NOM-041-ECOL-1999*.

— (1999b). *Norma Oficial Mexicana NOM-042-ECOL-1999*.

Dorronsoro, B. et al. (2013). "Achieving super-linear performance in parallel multiobjective evolutionary algorithms by means of cooperative coevolution". In: *Computers and Operations Research* 40.6, pp. 1552–1563.

Erdoğan, S. and E. Miller-Hooks (2012). "A green vehicle routing problem". In: *Transportation Research Part E: Logistics and Transportation Review* 48.1, pp. 100–114.

Forster, Piers et al. (2007). "Changes in Atmospheric Constituents and in Radiative Forcing". In: *Climate Change 2007: The Physical Science Basis. Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*. Ed. by D. Qin M. Manning-Z. Chen M. Marquis K.B. Averyt M.Tignor Solomon S. and H.L. Miller. Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press. Chap. 2, pp. 129–234.

Gimenez, C., V. Sierra, and J. Rodon (2012). "Sustainable operations: Their impact on the triple bottom line." In: *International Journal of Production Economics* 140.1, pp. 149–159.

Glover, F. (1986). "Future paths for integer programming and links to artificial intelligence". In: *Computers and operations research* 13.5, pp. 533–549.

Glover, F. and M. Laguna (1993). "Modern heuristic techniques for combinatorial problems". In: *Blackwell, London*.

Gonçalves, J.F. and M.G.C. Resende (2011). "Biased random-key genetic algorithms for combinatorial optimizatio". In: *J Heuristics* 17, p. 487.

Hansen, J. et al. (1981). "Climate impact of increasing atmospheric carbon dioxide. Science". In: *Science* 213.4511, pp. 957–966.

Harris, I., C. Mumford, and M. Naim (2009). "The multi-objective uncapacitated facility location problem for green logistics". In: *In 2009 IEEE Congress on Evolutionary Computation* IEEE, pp. 2732–2739.

Harris, I., C.L. Mumford, and M. Naim (2014). "A hybrid multi-objective approach to capacitated facility location with flexible store allocation for green logistics modeling". In: *Transportation Research Part E: Logistics and Transportation Review* 66, pp. 1–22.

Hizir, A. E. (2006). "Using emission functions in mathematical programming models for sustainable urban transportation: An application in bilevel optimization". In: *Doctoral dissertation, Sabancı University*.

Ho, S.C. and M. Gendreau (2006). "Path relinking for the vehicle routing problem". In: *Journal of Heuristics* 12.1-2, pp. 55–72.

Huppes, Gjalt and Masanobu Ishikawa (2005). "A framework for quantified eco efficiency analysis". In: *Journal of Industrial Ecology* 9.4, pp. 25–41.

Kalashnikov, V. V. et al. (2015). "Bilevel programming and applications". In: *Mathematical Problems in Engineering* 2015.

Kannan, D. et al. (2013). "Integrated fuzzy multi criteria decision making method and multi-objective programming approach for supplier selection and order allocation in a green supply chain". In: *Journal of Cleaner Production* 47, pp. 355–367.

Konak, A., D.W. Coit, and A.E. Smith (2006). "Multi-objective optimization using genetic algorithms: A tutorial". In: *Reliability Engineering and System Safety,* 91.9, pp. 992–1007.

Kulturel-Konak, S., A.E. Smith, and B.A. Norman (2006). "Multi-objective tabu search using a multinomial probability mass function". In: *European Journal of Operational Research* 169.3, pp. 918–931.

Labuschagne, Carin, Alan C. Brenta, and Ron P.G. van Erck (2005). "Assessing the sustainability performances of industries". In: *Journal of cleaner production* 13.4, pp. 373–385.

Lai, K. H. and C. W. Wong (2012). "Green logistics management and performance: Some empirical evidence from Chinese manufacturing exporters". In: *Omega* 40.3, pp. 267–282.

Legillon, F., A. Liefooghe, and E.G. Talbi (2012). "Cobra: A cooperative coevolutionary algorithm for bi-level optimization". In: *2012 IEEE Congress on Evolutionary Computation* IEEE, pp. 1–8.

Lenir, P., R. Govindarajan, and S.S. Nemawarkar (1992). "Exploiting instruction-level parallelism: the multithreaded approach". In: *ACM SIGMICRO Newsletter ,IEEE Computer Society Press* 23.1-2, pp. 189–192.

Li, F. et al. (2008). "Distribution center location for green supply chain." In: *Service Operations and Logistics, and Informatics, 2008. IEEE/SOLI 2008* 2, pp. 2951–2956.

Linton, J. D., R. Klassen, and V. Jayaraman (2007). "Sustainable supply chains: An introduction". In: *Journal of operations management* 25.6, pp. 1075–1082.

Martí, Rafael, José Luis González-Velarde, and Abraham Duarte (2009). "Heuristics for the bi-objective path dissimilarity problem". In: *Computers & Operations Research* 36.11, pp. 2905–2912.

Mathew, T. V. and S. Sharma (2006). "Continuous network design with emission pricing as a bi-level optimization problem". In: *In Proceedings of the Ninth International Conference on Applications of Advanced Technology in Transportatio*, pp. 804–809.

Morán-Mirabal, L. F., J.L. González-Velarde, and M.G.C. Resende (2014). "Randomized heuristics for the family traveling salesperson problem". In: *International Transactions in Operational Research* 21.1, pp. 41–57.

Noronha, T.F., M.G.C. Resende, and C.C. Ribeiro (2011). "A biased random-key genetic algorithm for routing and wavelength assignment". In: *Journal Global Optimization* 50, p. 503.

Oduguwa, V. and R. Roy (2002). "Bi-level optimisation using genetic algorithm. In Artificial Intelligence Systems, 2002.(ICAIS 2002)". In: *2002 IEEE International Conference* IEEE, pp. 322–327.

Potter, M.A. and K.A. De-Jong (1994). "A cooperative coevolutionary approach to function optimization". In: *In International Conference on Parallel Problem Solving from Nature* Springer Berlin Heidelberg. Pp. 249–257.

Sakawa, M. and I. Nishizaki (2002). "Interactive fuzzy programming for two-level nonconvex programming problems with fuzzy parameters through genetic algorithms". In: *Fuzzy Sets and Systems* 127.2, pp. 185–197.

Sawaragi, Y., H. Nakayama, and T. Tanino (1985). "Theory of multiobjective optimization". In: *Elsevier* 176.

Sbihi, A. and R. W. Eglese (2010). "Combinatorial optimization and green logistics". In: *Annals of Operations Research* 175.1, pp. 159–175.

Sinha, A., P. Malo, and K. Deb (2016). "Solving Optimistic Bilevel Programs by Iteratively Approximating Lower Level Optimal Value Function". In:

Szekely, F. and M. Knirsch (2005). "Responsible leadership and corporate social responsibility: Metrics for sustainable performance". In: *European Management Journal* 23.6, pp. 628–647.

Ubeda, S., F. J. Arcelus, and J. Faulin (2011). "Green logistics at Eroski: A case study". In: *International Journal of Production Economics* 131.1, pp. 44–51.

Wang, G.M., L.M. Ma, and L.L. Li (2011). "An application of bilevel programming problem in optimal pollution emission price". In: *Journal of Service Science and Management* 4.3.

Wang, Jian et al. (2014). "Urban Traffic Congestion Pricing Model with the Consideration of Carbon Emissions Co". In: *Sustainability* 6.2, pp. 676–691.

Yeh, W. C. and M.C. Chuang (2011). "Using multi-objective genetic algorithm for partner selection in green supply chain problems". In: *Expert Systems with applications* 38.4, pp. 4244–4253.

Yin, P. Y., S.R. Lyu, and Y.L. Chuang (2016). "Cooperative coevolutionary approach for integrated vehicle routing and scheduling using cross-dock buffering". In: *Engineering Applications of Artificial Intelligence* 52, pp. 40–53.

Zitzler, Eckart and Lothar Thiele (1999). "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach". In: *IEEE transactions on Evolutionary Computation* 3.4, pp. 257–271.