Alma Mater Studiorum - Università di Bologna

DOTTORATO DI RICERCA IN

Automatica e Ricerca Operativa

Ciclo XXVIII

Settore concorsuale di afferenza: 01/A6 - RICERCA OPERATIVA

Settore scientifico disciplinare: MAT/09 - RICERCA OPERATIVA

Mathematical Optimization for Routing and Logistic Problems

Presentata da: Claudio Gambella

Coordinatore Dottorato
Prof. Daniele Vigo

Relatore
Prof. Daniele Vigo
Co-relatore
Prof. Andrea Lodi

Esame finale anno 2016

Contents

List of Figures

List of Tables

Keywords

- *Mathematical Optimization*
- *Second-Order Conic Programming*
- *Mixed-Integer Second-Order Conic Programming*
- *Path Planning*
- *Mission Planning*
- *Traveling Salesman Problem*
- *Vehicle Routing Problem*
- *Branch-and-Price*
- *Lagrangian Relaxation*
- *Lagrangian Decomposition*
- *Waste Management*
- *Stochastic Programming*
- *Two-Stage Multiperiod Stochastic Programming*
- *Electric Car-Sharing*

Acknowledgements

I am grateful to my supervisors Prof. Daniele Vigo and Prof. Andrea Lodi for their guidance and help in my research projects. I want to thank them also for the great mobility opportunities offered, which gave me personal enrichment and stimulated my Ph.D. activity.
Many thanks go to Dr. Bissan Ghaddar and Prof. Joe Naoum-Sawaya for supervising me during the internship in IBM Ireland. Their availability and listening skill have deeply stimulated me and made me feel part of a productive team.
Regarding the research on waste management of Chapter 5, I am thankful to Prof. Francesca Maggioni for her assistance, to the waste operator Herambiente SpA and the consulting company Optit Srl for sharing their real data.
I would like to thank the components of the group of Operations Research of the DEI, Dipartimento di Ingegneria dell'Energia elettrica e dell'Informazione of the University of Bologna, namely Prof. Paolo Toth, Prof. Enrico Malaguti, Prof. Michele Monaci, Dr. Valentina Cacchiani, Dr. Paolo Tubertini, Ph.D. students Maxence Delorme, Alberto Santini, Dimitri Thomopulos and the former member Dr. Tiziano Parriani. A particular mention goes to the Ph.D. student Sven Wiese, for his patience and presence during these years.
I would like also to thank the Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR) for the financial support given to my Ph.D. course.
Finally, I am indebted to my family for their constant care and loving support.

Chapter 1

Introduction

Mathematical optimization (also referred to as *mathematical programming*) is a branch of applied mathematics that requires to solve a minimization or maximization problem subject to a set of constraints. The general form of representation of a constrained optimization problem is

$$\min f(x) \tag{1.1}$$

$$\text{s.t.} \quad x \in X \subseteq \mathbb{R}^n, \tag{1.2}$$

where f is a real-valued function, called *objective function*, and the *feasible region* X is the subset of values of the *decision variables* x that satisfy the problem *constraints*, given in the form of equalities or inequalities. Note that every maximization problem can be equivalently converted in a minimization one.

Optimization problems can be viewed as mathematical formulations of decision problems. The applications of mathematical modeling invest several areas, such as economy, finance, engineering, scheduling, military, routing and logistic problems. The purpose of the optimization is to give a decision support system with quantitative tools, in contrast with qualitative criteria motivated by empirical experience and personal judgement.

Two main classes of solution approaches for problems of form (1.1)-(1.2) are *exact methods* and *heuristic algorithms*. The aim of an exact method is to select an *optimal solution*, namely a vector x of decision variables that belongs to set X and minimizes the objective function f (i.e., $f(x) \leq f(x') \quad \forall\, x' \in X$). When modeling a practical problem, the size of the optimization problem can be very large in terms of decision variables and constraints; in addition, a complete and accurate description of the set of the model entities may be an intractable task. In such situations, heuristics are adopted for finding solutions of good proven quality in a reasonable amount of time.

A minimal classification of optimization problems produces four relevant categories:

- *Linear Programming (LP)* (Dantzig [75]): problems with objective function and constraints expressed by linear functions;
- *Mixed-Integer Linear Programming (MILP)* (see, e.g., Smith and Taskın [230]): LPs in which (some) decision variables are required to assume integer values;
- *Non-Linear Programming (NLP)* (see, e.g., Bertsekas [36]): problems where objective function and constraints are represented by nonlinear functions;
- *Mixed-Integer Non-Linear Programming (MINLP)* (see, e.g., Belotti et al. [31]): NLPs in which (some) decision variables are subject to integrality requirements.

Amongst nonlinear problems, an important distinction is made between Convex Programming (see, e.g., Boyd and Vandenberghe [45]) and Non-Convex Programming (see, e.g., Burer and Letchford [50]). Non-convex problems have a non-convex feasible region or a non-convex continuous relaxation, if integrality constraints are present. A primary implication of the non-convexity is that the optimization problem may have multiple optimal solutions. In Non-Convex Programming, the optimizer is mainly interested in finding locally optimal solutions, because proving the global optimality of a candidate solution may be a very difficult challenge.
The problem type affects the choice of the applicable methods for finding optimal solutions of the mathematical model. For practically solving large-scale problems, an initial approach is to invoke *optimization solvers*. Such commercial or non-commercial software contain solution algorithms that proved to be efficient and effective for specific classes of optimization problems. The development of tailored algorithms on the specific optimization problem may be instead required for various reasons, such as lowering the computational time required and dealing with the scalability of the model.
In order to obtain information about the optimization problem, relaxed problems may be considered. *Relaxations* are modeling strategies that permit to consider substantially easier problems than the original one. A relaxation of (1.1)-(1.2) is an optimization problem

$$\min f_R(x) \tag{1.3}$$

$$\text{s.t.} \quad x \in X_R \subseteq \mathbb{R}^n \tag{1.4}$$

that satisfies the conditions:

1. $X_R \supseteq X$;
2. $f_R(x) \leq f(x) \; \forall \; x \in X$.

The two conditions ensure that solving a relaxation of a minimization problem provides a *lower bound* on the optimal solution value. In the case of *continuous relaxation* of a MILP (obtained by dropping the integrality requirements), the resulting bound is used, for instance, in branch-and-bound methods (Land and Doig [166], Nemhauser and Wolsey [198], Papadimitriou and Steiglitz [203]).

This chapter is meant for laying the main theoretical basis of the research projects developed in the thesis. To this end, Sections 1.1 and 1.2 introduce two relevant classes of programming paradigms, Section 1.3 describe a solution method developed on the basis of a problem relaxation, and Section 1.4 summarizes the forms in which the optimization solvers have been used in the projects of the thesis. The practical relevance of the modeling paradigms and solution techniques introduced in this chapter will become more apparent over Chapters 2, 3 and 5. Finally, an overview of the thesis is given in Section 1.5.

1.1 Conic Programming

Conic Programming is a relevant subcategory of Convex Programming. The accurate representation of an optimization model is a central issue in Non-Linear Programming. While general non-linear constraints can be particularly tricky to handle, recognizing the conic property of a function allows to adopt tailored solution algorithms in order to efficiently find optimal solutions.

1.1.1 Conic Programs

Let $K \subset \mathbb{R}^n$ be a *cone* (i.e., closed under multiplication by positive scalars). The set K is said to be *regular* if it is convex, closed, it has a nonempty interior and it contains 0.
Let M be a $m \times n$ matrix, μ be a vector of $\mathbb{R}^m$ and γ be a vector of $\mathbb{R}^n$.
A *conic program on* K is an optimization problem of the form:

$$\min_{x \in \mathbb{R}^n} \{\gamma^T x : Mx - \mu \in K\}. \tag{1.5}$$

Conic programs are polynomially solvable when the associated cones are "computationally tractable" (i.e., admitting polynomial time membership/separation oracles) and the feasible region is appropriately bounded (Nemirovski [199]). In the case of problems on a cone from a family $\mathcal{K}$ of regular cones, fast interior point methods can be adopted: this motivates the interest in determining if a generic convex problem is representable as a conic problem, namely in a $\mathcal{K}$-representable form.
Representation (1.5) comprehends a wide variety of formulations, sharing the feature

of minimizing a linear objective function. Important and broad subclasses are given by the symmetric cones:

- $K = \mathbb{R}^m_+$, then problem (1.5) reduces to a Linear Programming problem (Dantzig [75])

$$\min_{x\in\mathbb{R}^n} \{\gamma^T x : Mx - \mu \geq 0\}. \tag{1.6}$$

- $K = \prod_{i=1}^{p} \mathcal{L}^{m_i}$, where each $\mathcal{L}^{m_i} = \{(y,t) \in \mathbb{R}^{m_i}, y \in \mathbb{R}^{m_i-1}, t \in \mathbb{R} : \|y\|_2 \leq t\}$ is a Lorentz cone. In this case, (1.5) is expressed in the form

$$\min_{x\in\mathbb{R}^n} \quad \gamma^T x \tag{1.7}$$

$$\|A_i x - b_i\|_2 \geq c_i^T x - d_i \quad i = 1,\dots,p, \tag{1.8}$$

where A_i is a $m_i \times n$ matrix, $b_i \in \mathbb{R}^{m_i}$, $c_i \in \mathbb{R}^n$ and $d_i \in \mathbb{R}$. Problem (1.7)-(1.8) is named as Conic Quadratic Programming (CQP) or Second-Order Conic Programming (SOCP) problem. Constraints (1.8) are called Conic Quadratic Inequalities (CQIs) or second-order cone constraints of dimension m_i.

- $K = \prod_{i=1}^{q} \mathcal{S}^{m_i}$, where each $\mathcal{S}^{m_i}$ belongs to the cone of positive semidefinite (i.e., $\succcurlyeq 0$) and symmetric $m_i \times m_i$ matrices with the Frobenius inner product $\langle A, B\rangle = Tr(AB)$.
Problem (1.5) therefore describes a Semi-Definite Programming (SDP) problem

$$\min_{x\in\mathbb{R}^n} \{\gamma^T x : \mathcal{A}_i x - B^i \equiv x_1 A_1^i + \cdots + x_n A_n^i - B^i \succcurlyeq 0 \quad i = 1,\dots,q\}, \tag{1.9}$$

where $A_j^i, B^i \in \mathcal{S}^{m_i}$. Each constraint of (1.9) is a Linear Matrix Inequality (LMI).

The three categories of symmetric cones are linked by the relationship: $LP \subset CQP \subset SDP$. Indeed, a one-dimensional CQI reduces to a linear inequality. Observe, in addition, that a Lorentz cone $(y,t) \in \mathcal{L}^m$is defined by the sdp matrix $\begin{pmatrix} t & y^T \\ y & tI_{m-1} \end{pmatrix}$.
In the following, we focus on the relevant case of Second-Order Conic Programming problems. They extend LPs and can also be considered as a special case of SDPs.

1.1.2 Second-Order Conic Programming

A Second-Order Conic Programming problem is defined as a conic problem with feasible region given by an intersection of affine spaces and finite direct product of Lorentz cones. Every SOCP can be represented as (1.7)-(1.8).
SOPCPs may be presented as subclasses of NLPs, with the remark that conic constraints are not globally differentiable, in general. Indeed, Euclidean norms are not differentiable in points at which they vanish. It worths mentioning that SOCPs comprehend quadratically-constrained problems, which are regular, as special cases: recognizing this situation can be particularly helpful for optimization solvers, since regular functions are easier to handle than non-smooth functions. Considering the SOCP (1.7)-(1.8), the i-th CQI becomes the quadratic constraint $\|A_i x - b_i\|_2^2 \geq d_i^2$ when $c_i = 0$ and $-d_i \geq 0$.

A relevant observation that enables to enlarge the set of SOCP-representable problems and functions is the following. By rotating the Lorentz cone $\mathcal{L}^m$ in the t, z plane through an angle of forty-five degrees, one obtains the *rotated quadratic cone*

$$\hat{\mathcal{L}}^m = \{(y, t, z) \in \mathbb{R}^m, y \in \mathbb{R}^{m_i - 2}, t \in \mathbb{R}, z \in \mathbb{R} : \|y\|_2^2 \leq 2tz\}.$$

For instance, this argument shows that *hyperbolic* constraints $\xi^T \xi \leq \lambda\mu, \lambda \geq 0, \mu \geq 0$ are SOCP-representable by the constraint $\left\| \begin{bmatrix} 2\xi \\ \lambda - \mu \end{bmatrix} \right\| \leq \lambda + \mu$ (Alizadeh and Goldfarb [6]).

A CQI can represent broad classes of nonlinear functions, such as: Euclidean norms ($f(x) = \|x\|_2$), convex quadratic forms ($f(x) = x^T A^T A x + b^T x + c$), univariate rational power functions, power monomials ($\prod_{i=1}^m x_i^{p_i}$, with $x_i \geq 0$ and rational exponentials $p_i \geq 0$ such that $\sum_i p_i \leq 1$. Hence, SOCPs can represent a wide variety of engineering and finance problems, such as filter design, antenna array weight design, truss design, portfolio optimization, equilibrium condition of mechanical systems (all described in Lobo et al. [178]), location-aided routing in mobile ad-hoc networks (Maggioni et al. [182]), sensor network localization (Tseng [242]), image restoration (Goldfarb and Yin [119]) and design of robust classifiers in machine learning (Shivaswamy et al. [227]). In addition, a relevant application of SOCP can arise in the context of *Robust Linear Programming*. In real-life optimization problems, the decision maker is often required to consider LPs in which (some) problem data are affected by uncertainty (Dantzig [74]). In such situations, a deterministic optimization problem may consider optimal solutions as infeasible solutions, because of even “small” errors in the determination of LP parameters. A possibility for limiting the effects of uncertainty is to consider the *Robust Counterpart* (RC) of the LP. Having an *uncertainty set* $\mathcal{U}$ in which (some of the) problem parameters may vary, then a candidate solution is said to be *robust feasible* if it satisfies the problem constraints in a worst-case framework, i.e., regardless the actual realization of the uncertain data in $\mathcal{U}$. The RC of the LP is the problem of minimizing the value of the objective function over robust feasible solutions. When $\mathcal{U}$ is CQP-representable (e.g., $\mathcal{U}$ is an intersection of boxes and ellipsoids), then the resulting RC is a CQP, which is therefore computationally tractable. For example, the RC of the least square problem can also be formulated by means of conic functions (see, e.g., El Ghaoui and Lebret [86] and Chandrasekaran et al. [57]).

Algebraic structure, duality theory, complementarity theory and primal-dual interior point methods for SOCPs are covered in detail in Alizadeh and Goldfarb [6]. Primal-dual algorithms for SOCP prove to be more effective than primal- or dual-only approaches. While primal-only or dual-only methods for LPs can be adapted for solving SOCPs with limited effort, natural extensions of primal-dual methods for LPs to SOCPs must face non-commutativity problems (see Section 7.1 of Alizadeh and Goldfarb [6]), whether they are path-following (Gonzaga [120]) or potential-reduction algorithms (Todd [237]). One of the most effective methods for solving SOCPs is the primal-dual potential reduction method of Nesterov and Nemirovski [200]. Other tailored algorithms suitable for solving SOCPs are given, for instance, by a simplex method for conic problems (Goldfarb [118]) and by methods based on polyhedral reformulations of the second-order cone constraints (Ben-Tal and Nemirovski [32]). The lack of regularity of CQI prevents the direct applicability of some NLP algorithms (e.g., interior point methods requiring the second-order differentiability of objective function and constraints). Taking advantage of the fact that the non-global differentiability of the Euclidean norm is an issue for a generic NLP algorithm only if it is present in an optimal solution, an SOCP can be solved as a special case of NLP after considering some reformulations of Lorentz cones. Such techniques are summarized in Section 3.2 of Benson and Saglam [35]. However, the precision of the solution obtained after manipulating the problem constraints is an issue to be considered.

1.1.3 Mixed-Integer Second-Order Conic Programming

A Mixed-Integer Second-Order Conic Programming (MISOCP) problem is an SOCP problem in which (some of) the decision variables are subject to integrality requirements. Hence, every MISOCP can be presented in the following form:

$$\min_{x \in \mathbb{R}^n} \quad \gamma^T x \tag{1.10}$$

$$\|A_i x - b_i\|_2 \geq c_i^T x - d_i \quad i = 1, \ldots, p, \tag{1.11}$$

$$x_j \in \mathbb{Z} \quad j \in J, \tag{1.12}$$

where problem data $\gamma, A_i, b_i, c_i, d_i$ have suitable dimensions and $J \subseteq \{1, \ldots, n\}$. Clearly, when J is empty, problem (1.10)-(1.12) reduces to an SOCP.

In MISOCP applications, the variety of use of binary or integer variables can be very wide: for example, investment options in portfolio optimization (see, e.g., Bonami and Lejeune [40]), options pricing in a financial market under uncertainty (Pınar [208]), network design in telecommunication networks (see, e.g., Cheng et al. [61] and Hijazi et al. [138]), Euclidean k-center problem Brandenberg and Roth [46], facility location and inventory management (Atamtürk et al. [17]).

The method used for solving the continuous SOCP relaxation of an MISOCP plays a fundamental role in building an efficient solution algorithm for the mixed-integer problem. MISOCP algorithms can be essentially divided into two groups: extension of MILP methods, motivated by the relationship between second-order and linear cones, or tailored MINLP approaches that consider the SOCP subproblems as special NLPs. The first group is constituted by branch-and-bound algorithms, in analogy with the methods for MILPs. In such methods, the nodes of the solution tree are constituted by SOCP problems. Hence, an SOCP solver capable of warmstarting and detecting infeasible subproblems would be particularly competitive for the overall MISOCP method. For improving the performance of the branch-and-bound algorithms, valid inequalities such as Gomory cuts (Çezik and Iyengar [56]) and mixed-integer rounding cuts (Atamtürk and Narayanan [16]) for MISOCP may be considered.
The latter set comprehends algorithms based on outer approximation (Duran and Grossmann [83]), extended cutting-plane methods (Westerlund and Pettersson [252]), LP/NLP-based branch-and-bound (Quesada and Grossmann [212]) and generalized Benders decomposition (Geoffrion [108]). Such methods rely on polyhedral relaxations of the second-order conic constraints and may consider SOCP subproblems for improving upper and lower bounds, fathom nodes in the branch-and-bound algorithm or also as a local search procedure. As mentioned in Section 1.1.2, the main issue in applying NLP-based methods to SOCPs is the non-global differentiability of second-order conic constraints, which prevents the use of gradient-based cuts for solving MISOCP. Possible alternatives are to consider subgradients cuts (for instance, as proposed in the branch-and-cut and hybrid branch-and-bound/outer approximation methods of Drewes and Ulbrich [81]), or lifted polyhedral relaxations (Ben-Tal and Nemirovski [32], Glineur [116]) that also help to tighten lower bounds.

1.1.4 Solvers for (MI)SOCPs

In order to find optimal solutions for (MI)SOCP, an optimizer may be interested in using commercial and non-commercial optimization solvers. Recognizing the conic structure of the optimization problem is crucial to choose the appropriate solver and algorithm tailored for (MI)SOCP.

For an overview of Conic Programming solvers, the reader is referred to Mittelmann [193] and Conic Programming Solvers [67]. A brief description of the solvers applied to MISOCP formulations of Chapters 2 and 3 of the thesis is here reported:

- Developed by IBM ILOG, Cplex [70] contains an optimization suite of state-of-the-art solvers for linear programming, mixed-integer programming, (mixed-integer) quadratic programming, and (mixed-integer) quadratically-constrained programming problems. In particular, an MISOCP can be handled with an NLP branch-and-bound algorithm (i.e., solving a quadratic relaxation at each node) or in an outer approximation scheme (i.e., solving a linear relaxation at each node).

- The Gurobi Optimizer [128] acquires its name from the founders Zonghao Gu, Edward Rothberg and Robert Bixby. Similarly as Cplex, Gurobi offers mathematical programming solvers for handling major problem types.

- SCIP (Solving Constraint Integer Programs) (Achterberg [4]) is a mixed-integer programming and mixed-integer nonlinear programming solver and a framework for branch-and-cut and branch-and-price developed at Zuse Institute Berlin. SCIP is based on the notion of Constraint Integer Programming, which is a framework for integrating constraint programming (see, e.g, Apt [10], Wallace [249]) and mixed-integer programming modeling and solving techniques.

- Maintained in MOSEK ApS, MOSEK (Andersen and Andersen [8]) is a software to solve mathematical problems such as linear programs, quadratic and quadratically constrained programs, conic problems and mixed-integer problems. The strong point of MOSEK is its state-of-the-art interior-point optimizer for continuous linear, quadratic and conic problems.

- Xpress [256] is an optimization suite for linear programming, mixed-integer linear programming, convex quadratic programming, convex quadratically-constrained quadratic programming , second-order cone programming and their mixed integer counterparts. In addition to the Optimizer, Xpress includes the general purpose nonlinear solver Xpress-NonLinear and the modeling language Xpress-Mosel ([255]). Xpress was originally developed by Dash Optimization and later acquired by FICO.

Apart from MOSEK, all solvers require the CQIs of a MISOCP to be written in the quadratic form.

1.2 Stochastic Programming

The mathematical formulation of a real-life decision problem may not be accurately represented by a *deterministic* optimization problem, namely a model in which all data are known with certainty. For instance, uncertainty can be identified in customers demand in transportation, energy and finance problems, or in cost and prices parameters in logistics and engineering applications. In case the values of (some of) the problem data are not known at the moment of making a decision, a wrong estimation of the uncertain parameter may lead the model to certify solutions that are of poor quality or infeasible as optimal solutions. The deterministic model, in which the future is supposed to be fully and perfectly known, may therefore be misleading. This urges to adopt mathematical formulations which take into account the uncertainty affecting the problem data, in the spirit of *decision-making under uncertainty*.

In Section 1.1.2, we mentioned the *Robust Counterpart* of LP for obtaining solutions that are feasible for a range of uncertain values of problem parameters. Robust Programming paradigm is a conservative manner of considering the variability of the problem data. Another paradigm for modeling uncertainties is the *Chance-Constrained Programming* framework (see Charnes and Cooper [58], Heilmann [131]), which expresses the requirement of satisfying constraints under confidence levels. Chance-constrained formulations accept the risk of obtaining infeasible solutions by means of probabilistic constraints (Prekopa [210]).

The value of *Stochastic Programming* approaches lies in the explicit evaluation of flexible solutions against uncertainty. The principle is that it is impossible to make decisions that are optimal in all circumstances, namely for every realization of the random quantities of the problem. In the Stochastic Programming paradigm, data randomness is represented by random variables. It is assumed that the information on the stochastic nature of the problem enables a description of the random variables ψ, under the form of the probability distributions, densities or, more generally, probability measures. An outcome of the random variables is denoted by $\omega \in \Omega$, i.e., $\psi = \psi(\omega)$. This description of the random variables is relevant for the phase of *scenario generation*. Scenarios are a finite set of representative outcomes of the uncertain data. Each scenario is associated with a probability of realization (*scenario probability*). If the scenario representation is accurate, in the long run one expects to observe all scenarios with the occurrence given by the probability. Hence, the stochastic models aim at taking decisions balanced or hedged against the various scenarios.

The introduction of quantified uncertainty in stochastic optimization problems largely increases the size of the resulting optimization problem. However, stochastic programming formulations possess special structure, which is exploited in algorithms for handling large-scale models. Among the methods of this class, we mention the L-shaped method (Van Slyke and Wets [244]), the Regularized Decomposition method (Ruszczyński [217]) for two-stage problems and Nested Decomposition procedures (Louveaux [179], Noël and Smeers [201], Pereira and Pinto [206]) for multistage formulations.

The remainder of the section gives an introduction to the Stochastic Programming paradigm. Stochastic formulations are given in Chapter 5 for a waste management problem. For more in-depth introductions to the topic, the reader is referred to the books of Birge and Louveaux [38] and Kall and Wallace [151]. The Two-Stage Stochastic Programming paradigm is presented in Section 1.2.1, while the extension to the Multistage setting is developed in Section 1.2.2. In Sections 1.2.3 and 1.2.4 standard stochastic measures and bounds for validating a stochastic model are provided: their purpose is to give quantitative information on the impact of the uncertainty in the decision problem.

1.2.1 Two-Stage Stochastic Programming

Two-Stage Stochastic Linear Programming is about taking *recourse actions* in optimization problems with uncertain data. The recourse is a corrective action in response to the disclosure of the uncertainty.
The set of decision variables is then divided into *first-stage decisions* and *second-stage decisions*. First-stage variables x represent the decision to take before the outcome ω of the random variables ψ becomes known: this period is called the *first stage*. After the disclosure of ψ, second-stage decisions y can be taken in the so-called *second stage* period. A schematic representation of the realization of uncertainty and the decisions to take in a Two-Stage Stochastic Programming model is thus

$$x \longrightarrow \psi(\omega) \longrightarrow y(\omega, x).$$

First-stage decisions are also referred to as *nonanticipative* decisions, because they are taken at the moment in which it is not possible to anticipate every possible realization of the uncertain data. In logistic problems of planning under uncertainty, such as facility location problems (Balinski [20], Silva and De la Figuera [228]), the first-stage decisions are usually strategic decisions which are not fully alterable (e.g., deciding which resources are active or which plants are open), while short-term modifications are considered in the second stage variables (e.g., determining the transportation plan). An example of stochastic facility location problem is considered in Louveaux and Peeters [180].
We provide a mathematical formulation for the classical *two-stage stochastic linear program with fixed recourse* (introduced by Dantzig [74] and Beale [26]), also called *recourse problem* (RP)

$$RP := \quad \min c^T x + \mathbb{E}_\psi[\min q(\omega)^T y(\omega)] \tag{1.13}$$

$$\text{s.t.} \quad Ax = b, \tag{1.14}$$

$$T(\omega)x + Wy(\omega) = h(\omega), \tag{1.15}$$

$$x \geq 0, y(\omega) \geq 0. \tag{1.16}$$

The model components are divided between the first-stage and the second-stage period. The first-stage vectors $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$ and the real matrix A of size $m_1 \times n_1$ are associated with the first-stage decisions $x \in \mathbb{R}^{n_1}$. The second-stage data $(q(\omega) \in$

$\mathbb{R}^{n_2}, h(\omega) \in \mathbb{R}^{m_2}$ and the real matrix $T(\omega)$ of size $m_2 \times n_1$) are dependent on the outcome ω of uncertainty. The recourse matrix W of size $m_2 \times n_2$ is independent of the realization ω, hence problem (1.13)-(1.16) is with *fixed recourse*. When the random event ω is realized, the data are known, and the second-stage decisions $y(\omega) \in \mathbb{R}^{n_2}$ can be taken by solving a linear program. The objective function (1.13) is composed by a deterministic term $c^T x$ and by the *recourse term*, given by the expectation $\mathbb{E}_\psi$ of the second-stage objective $q(\omega)^T y(\omega)$ over ψ. The recourse cost $q(\omega)$ can be thought as a penalty for the errors in the constraints with respect to the possible realization of the random variables. Two-stage stochastic *mixed-integer* formulations can be considered as an extension of (1.13)-(1.16) by replacing constraints (1.16) with $x \in X, y(\omega) \in Y$, where $X \subset \mathbb{Z}_+^{n_1}$ and $Y \subset \mathbb{Z}_+^{n_2}$.

For practically solving RP, the uncertainty is discretized by means of a finite set $\mathcal{S}$ of representative scenarios s. Figure 1.1 displays a scenario tree for a two-stage formulation with 6 scenarios.

Problem (RP) can be reformulated in a *extensive form*:

$$RP = \quad \min c^T x + \sum_{s \in \mathcal{S}} p^s (q^s)^T y^s \tag{1.17}$$

$$\text{s.t.} \quad Ax = b, \tag{1.18}$$

$$T^s x + W y^s = h^s \qquad \forall\ s \in \mathcal{S}, \tag{1.19}$$

$$x \geq 0,\ y^s \geq 0 \qquad \forall\ s \in \mathcal{S}, \tag{1.20}$$

where $q^s, T^s, h^s, \forall\ s \in \mathcal{S}$ are scenario-dependent parameters, and $p^s, \forall\ s \in \mathcal{S}$ are the scenario probabilities, which respect the condition $\sum_{s \in \mathcal{S}} p^s = 1$.

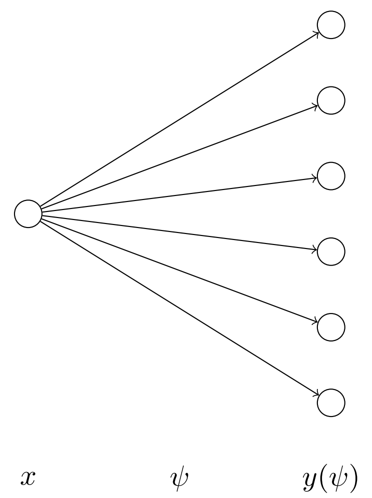


FIGURE 1.1: Two-stage scenario tree with $|\mathcal{S}| = 6$

1.2.2 Multistage Stochastic Programming

In Section 1.2.1 the stochastic formulation involves two stages, meaning that the decisions are referred to two time periods separated by the complete disclosure of the uncertainty. In many decision-making problems, the outcome of random variables is instead gradually revealed over multiple stages and the decisions are distributed over multiple periods.

In this situation, the uncertain parameters are modeled with a *random process* over $H-1$ stages:

$$\boldsymbol{\psi}^{H-1} = (\psi^1, \psi^2, \dots, \psi^{H-1}),$$

where each $\psi^t,\ t = 1, \dots, H-1$ is a random variable. The vector $\boldsymbol{\psi}^t = (\psi^1, \psi^2, \dots, \psi^t)$ denotes the random variables related to periods until the t-th one.

The decision vector

$$x = (x^1, x^2(\psi^1), \dots, x^{H-1}(\boldsymbol{\psi}^{H-2}), x^H(\boldsymbol{\psi}^{H-1}))$$

groups the decision variables according to the period $t = 1, \dots, H$ to which they are referred. The relationship between decision variables and the disclosure of uncertainty in the multistage setting is as follows:

$$\text{decision}(x^1) \to \text{realization}(\psi^1) \to \text{decision}(x^2) \to \text{realization}(\psi^2) \to \cdots$$
$$\cdots \to \text{decision}(x^{H-1}) \to \text{realization}(\psi^{H-1}) \to \text{decision}(x^H).$$

Decisions x^t at time period t depend on the history up to time t. The set of nonanticipative decisions is represented by variables x^1.

The nested formulation for a Multistage Stochastic Programming formulation with fixed recourse is then given by

$$\begin{aligned}
RP := & \min_x \mathbb{E}_{\boldsymbol{\psi}^{H-1}} z(x, \boldsymbol{\psi}^{H-1}) = \\
& = \min_{x^1} c^1 x^1 + \mathbb{E}_{\psi_1} \big[\min_{x^2} c^2(\psi^1) x^2(\psi^1) + \mathbb{E}_{\psi^2} \big[\dots \\
& \quad \cdots + \mathbb{E}_{\psi^{H-1}} \big[c^H(\psi^{H-1}) x^H(\boldsymbol{\psi}^{H-1}) \big]\big]\big] && (1.21) \\
\text{s.t. } & Ax^1 = h^1, && (1.22) \\
& T^1(\psi^1)x^1 + W^2 x^2(\psi^1) = h^2(\psi^1), && (1.23) \\
& \vdots \\
& T^{H-1}(\boldsymbol{\psi}^{H-1}) x^{H-1}(\boldsymbol{\psi}^{H-2}) + W^H x^H(\boldsymbol{\psi}^{H-1}) = h^H(\boldsymbol{\psi}^{H-1}), && (1.24) \\
& x^1 \geq 0,\ x^t(\boldsymbol{\psi}^{t-1}) \geq 0,\ \forall\, t = 2, \dots, H, && (1.25)
\end{aligned}$$

where: the transpose sign ^{T} is omitted when clear from the context; $\mathbb{E}_{\psi^t}$ indicates the expectation with respect to random variable ψ^t and ψ^t both denotes a random vector and its particular realization. The set of parameters of model (1.21)-(1.25) is constituted by

- known vectors $c^1 \in \mathbb{R}^{n_1}$, $h^1 \in \mathbb{R}^{m_1}$ and real matrix A of size $m_1 \times n_1$ are known (i.e., deterministic parameters);
- vectors $h^t \in \mathbb{R}^{m_t}, c^t \in \mathbb{R}^{n^t}$ and real matrices T^{t-1} of size $m_{t-1} \times n_{t-1}$ and W^t of size $m_t \times n_t$, with $t = 2, \dots, H$, are stochastic parameters.

For $H = 2$, the formulation represents a two-stage model.

In a multistage framework, the discretization of the set of possible outcomes of the random vectors ψ^t is represented in a *scenario tree*. An example of implicit representation of a scenario tree with 4 stages and 2 branches per stage is given in Figure 1.2. Each scenario is associated with a path from the root to a leaf node. The root node corresponds to the situation in which no outcome of randomness has been observed. Every branch in the tree corresponds to the disclosure of the random vector in a specific time period. Each stage corresponds to a decision time, while each period t is the term between two consecutive stages t and $t+1$ in which the random vector ψ^t is revealed.

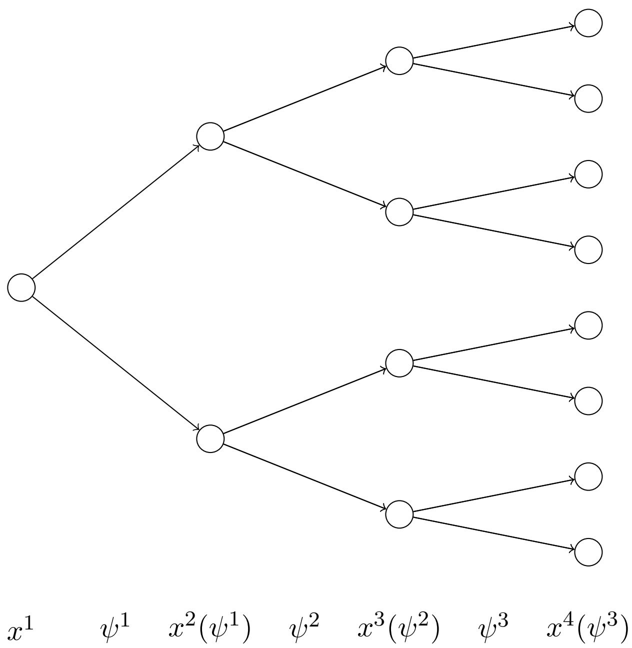


FIGURE 1.2: Multistage scenario tree with 4 stages and 2 branches per stage, $|\mathcal{S}| = 8$

The choice of the *scenario-generation method* to adopt is a relevant phase of the modeling process and is typically problem-dependent. A "good" scenario generation method should influence the solution as little as possible and be such that the scenario-based solution converges to the true optimal solution when the number of scenarios increases. Since a large size of the scenario set carries a computational load, the mathematical modeler should also estimate the number s^* for which the scenario representation of the uncertain parameters can be considered as an acceptable description of the future. Several scenario generation methods may be applied to the stochastic formulation and then test which one works better. For an overview of scenario-generation methods and applications, the reader is referred to the Ph.D. thesis of Kaut [153] and the papers of Høyland and Wallace [141], Kaut and Wallace [154], Pflug [207].

The particular case in which the uncertainty in revealed only once along a multiperiod problem gives rise to a two-stage multiperiod formulation, called *Two-stage relaxation Problem* (TP) (Maggioni et al. [184]). The TP framework models the case study of Chapter 5.

1.2.3 Measures and Bounds for Two-Stage Stochastic Programming

Consider the deterministic optimization problem associated with a scenario ψ,

$$\min z(x,\psi) = \quad c^T x + \min_{y\geq 0} q^T y \tag{1.26}$$

$$\text{s.t.} \qquad Ax = b, \tag{1.27}$$

$$Tx + Wy = h, \tag{1.28}$$

$$x \geq 0. \tag{1.29}$$

We assume that the optimal solution value of (1.26)-(1.29) is finite $\forall\, \psi$. The optimal solutions $\bar{x}(\psi)$ of (1.26)-(1.29) are chosen in the ideal situation of knowing the realization of the random variable in advance. They correspond to the case in which the decision maker is allowed to postpone all his decisions after the disclosure of uncertainty. The *Wait-and-See* (*WS*) solution is then defined as

$$WS = \mathbb{E}_\psi[\min_x z(x,\psi)] = \mathbb{E}_\psi z(\bar{x}(\psi),\psi), \tag{1.30}$$

The *WS* can be compared with the optimal solution value *RP*, which is obtained in the case in which only a statistical information about the distributions of the random variable is available, namely $RP = \min_x \mathbb{E}_\psi z(x,\psi)$. The difference between the *WS* approach and the stochastic solution is called the *Expected Value of using Perfect Information* (*EVPI*), namely:

$$EVPI = RP - WS. \tag{1.31}$$

EVPI was introduced in Avriel and Williams [18] and represents the increase of costs due to the random components of the problem. Large values of *EVPI* indicate a strong presence of variability in the problem data and hence suggest to adopt stochastic solutions rather than deterministic solutions. *EVPI* can be viewed as the money that the decision maker would be willing to pay in order to obtain a perfect knowledge of the

future.
Another method to evaluate the importance of the stochastic solution is to compare it with the solution of a much simpler problem. Consider the expected value $\bar{\psi} = \mathbb{E}(\psi)$ of random variables ψ. The *Expected Value problem* or *Mean Value problem* $EV = \min_x z(x, \bar{\psi})$ yields an optimal solution $\bar{x}(\bar{\psi})$ that completely neglects the variability of ψ, because stochastic parameters have been replaced by their mean values. Introducing the *Expected result of using the EV solution* to be $EEV = \mathbb{E}_{\psi} z(\bar{x}(\bar{\psi}), \psi)$, one can evaluate how well the second-stage decisions are taken, after fixing the first-stage variables as prescribed by the EV problem. The *Value of the Stochastic Solution* (*VSS*) (suggested in Birge [37]) measures how the decision $\bar{x}(\bar{\psi})$ behaves in terms of *RP*, in the following way:

$$VSS = EEV - RP. \tag{1.32}$$

VSS represents the possible gain obtained from solving the stochastic model.
To sum up, *EVPI* measures the value of knowing the future with certainty (assuming the scenario tree chosen is a good approximation of the reality), while *VSS* assesses the value of knowing and using distributions of future outcomes.
The relationships

$$WS \leq RP \leq EEV \tag{1.33}$$

assure that both WS and $EVPI$ are non-negative. Inequalities (1.33) are proven in [38].
The EEV can be an infeasible problem if the second-stage variables are not able to compensate the bad choices made by the EV solutions. In this case, a generalization of the *VSS* measure can be more relevant and permits to consider refined bounds.
Consider the extensive formulation (1.17)-(1.20) of *RP* in which only the right-hand side is stochastic ($\psi = h(\omega)$). Consider a *reference* scenario ψ_r; classical choices are given by $\bar{\psi}$ and the worst-case scenario. Choosing $k \in \mathcal{S}$, the *pair subproblems* $PAIRS(\psi^r, \psi^k)$ is:

$$\begin{aligned}
\min z^p(x, \psi^r, \psi^k) = \quad & c^T x + p^r q^T y(\psi^r) + (1 - p^r) q^T y(\psi^k) && (1.34)\\
\text{s.t.} \quad & Ax = b, && (1.35)\\
& Wy(\psi^r) = h(\psi^r) - Tx, && (1.36)\\
& Wy(\psi^k) = h(\psi^k) - Tx, && (1.37)\\
& x, y \geq 0. && (1.38)
\end{aligned}$$

Consider also the *Sum of Pairs Expected Values (SPEV)*

$$SPEV = \frac{1}{1 - p^r} \sum_{\substack{k \in \mathcal{S} \\ k \neq r}} p^k \min z^p(x, \psi^r, \psi^k). \tag{1.39}$$

The definition of the *PAIRS* problem is valid even for reference scenarios not belonging to $\mathcal{S}$. In such a case, $SPEV$ reduces to be equal to WS. In the general case, WS and

$SPEV$ are related to RP, by the inequalities

$$WS \leq SPEV \leq RP \tag{1.40}$$

(the proof can be found in [38]).
In order to develop additional bounds on RP related to the pairs subproblems, a generalization of the *VSS* measure is considered. Let $z(x, \psi^r)$ be the optimization problem associated with a reference scenario ψ^r, and $\bar{x}^r$ an optimal solution of $\min_x z(x, \psi^r)$. Computing the *Expected Value of the Reference Scenario* (*EVRS*)

$$EVRS = \mathbb{E}_\psi z(\bar{x}^r, \psi), \tag{1.41}$$

the *VSS* is then redefined as $VSS = EVRS - RP$. Either $\bar{x}^r$ is feasible for RP or infeasible for RP (in the latter case $EVRS := +\infty$), the *VSS* is still nonnegative.
An additional measure is introduced for refining the bounds on RP and *VSS*. Let $(\bar{x}^k, \bar{y}^k, y(\psi^k))$ the optimal solution of $PAIRS(\psi^r, \psi^k)$. Then, the *Expectation of Pairs Expected Value* (*EPEV*) is defined as

$$EPEV = \min_{k \in \mathcal{S} \cup \{r\}} \mathbb{E}_\psi z(\bar{x}^k, \psi). \tag{1.42}$$

The following bounds for RP and *VSS* then hold:

$$RP \leq EPEV \leq EVRS, \tag{1.43}$$

$$0 \leq EVRS - EPEV \leq VSS \leq EVRS - SPEV \leq EVRS - WS. \tag{1.44}$$

1.2.4 Measures and Bounds for Multistage Stochastic Programming

In a multistage setting, the *WS* definition is easily extended from the two-stage case and *EVPI* is computed as in equation (1.31). Regarding the *VSS*, the definition needs to be addressed to the particular stage considered.
In analogy with the two-stage case, the *Expected Value problem* (*EV*) associated to multistage formulation (1.21)-(1.25) is obtained by substituting the random variables $\boldsymbol{\psi}$ with their expected values $\bar{\boldsymbol{\psi}} = (\mathbb{E}_{\psi^1}, \mathbb{E}_{\psi^2}, \ldots, \mathbb{E}_{\psi^{H-1}}) = (\bar{\psi}^1, \bar{\psi}^2, \ldots, \bar{\psi}^{H-1})$, namely

$$
\begin{aligned}
EV := & \min_x (z(x, \bar{\boldsymbol{\psi}})) = \\
& = \min_{x^1, \ldots, x^H} c^1 x^1 + c^H x^H && (1.45)\\
& \text{s.t. } Ax^1 = h^1, && (1.46)\\
& \quad T^1(\bar{\psi}^1) x^1 + W^2 x^2 = h^2(\bar{\psi}^1), && (1.47)\\
& \quad \vdots \\
& \quad T^{H-1}(\bar{\psi}^{H-1}) x^{H-1} + W^H x^H(\bar{\psi}^{H-1}) = h^H(\bar{\psi}^{H-1}), && (1.48)\\
& \quad x^t \geq 0, \ \forall\, t = 1, \ldots, H. && (1.49)
\end{aligned}
$$

In order to introduce the concept of *VSS* in multistage setting, we consider the *Expected result at stage t of using the Expected Value solution* EEV^t (Escudero et al. [89]). The EEV^t is given by the optimal solution value of the *RP* model where the decision variables $x^1, \dots, x^t$ until stage t are fixed at the optimal values obtained in the *EV* problem. Note that every EEV^t may be an infeasible subproblem, as happens for the stochastic formulations of Chapter 5.
The *Value of the Stochastic Solution at stage t*, VSS^t, is then defined as follows:

$$VSS^t = EEV^t - RP, \qquad \forall\, t = 1, \dots, H-1. \tag{1.50}$$

For multistage linear stochastic programs, we only mention the following bounds:

$$EV \leq WS, \tag{1.51}$$
$$VSS^t \leq EEV^t - EV \qquad \forall\, t = 1, \dots, H-1. \tag{1.52}$$

Proofs of (1.51), (1.52), further bounds and measures in multistage linear programs are discussed in Maggioni et al. [184].

1.3 Lagrangian Relaxation and Lagrangian Decomposition

The *Lagrangian Relaxation* (LR) is a relaxation method particularly suited for optimization models that exhibit a special structure (Geoffrion [109], Held and Karp [132, 133]). The present section considers problems with linear constraints only, however also nonlinear constraints can be relaxed in a Lagrangian fashion in MINLPs (see, e.g., Nowak [202]). *Lagrangian Decomposition* (Guignard and Kim [125, 126]) is an additional way of exploiting the specific structure of the optimization problem via Lagrangian relaxation.

1.3.1 Lagrangian Relaxation and Lagrangian Dual Problem

Consider the Integer Linear Programming (ILP) problem (P):

$$(P): \quad \min \gamma^T x \tag{1.53}$$
$$\text{s.t.} \quad Ax = b, \tag{1.54}$$
$$Cx = d, \tag{1.55}$$
$$x \in \mathbb{N}^n, \tag{1.56}$$

where $\gamma \in \mathbb{R}^n$, A is a $m \times n$ matrix, $b \in \mathbb{R}^m$, C is a $p \times n$ matrix and $d \in \mathbb{R}^p$.
Assume that (1.54) are "hard" constraints, in the sense that problem (1.53),(1.55),

(1.56) is easily solvable in comparison to the original problem (P). An example of such situation is found when (P) can be split into several independent subproblems if "linking" constraints (1.54) are omitted (Frangioni [97]). The idea of LR is to remove the complicating constraints from the constraint set and consider their violation in the objective function. Introducing Lagrange multipliers $\lambda \in \mathbb{R}^m$, the family of LRs of (P) with respect to (1.54) constraints is given by

$$(LR(\lambda)): \quad \min \gamma^T x + \lambda^T (b - Ax) \tag{1.57}$$

$$Cx = d, \tag{1.58}$$

$$x \in \mathbb{N}^n. \tag{1.59}$$

For every value of $\lambda \in \mathbb{R}$, $(LR(\lambda))$ satisfies conditions (1) and (2) for being a relaxation of (P). Note that, in case inequality constraints are relaxed, an appropriate imposition of the Lagrange multipliers sign is needed to respect requirement (1). Due to the structure of (P), solving $(LR(\lambda))$ is computationally viable.
Being a relaxation, the optimal solution $v(LR(\lambda))$ is not greater than the optimal solution value $v(P)$ of (P) for every choice of λ. The interest in finding the best possible Lagrange relaxation lies in the determination of tight lower bounds. Such problem is denoted as *Lagrangian Dual* (LD) problem:

$$(LD): \quad \max_{\lambda \in \mathbb{R}^m} v(LR(\lambda)). \tag{1.60}$$

Algorithms for obtaining practical solutions of (1.60) are introduced in Section 1.3.2.

1.3.2 Iterative Methods for Solving the Lagrangian Dual Problem

A popular and relatively simple solution approach for (LD) is the *sub-gradient optimization method* (Boyd and Mutapcic [44]), which is an iterative algorithm for maximization problems with concave and not globally differentiable objective function. Since its convergence tends to be slow in practical cases, the sub-gradient optimization is mainly adopted with an iteration limit in a heuristic framework. As an attempt to accelerate the convergence to the optimal solution of (LD), *bundle methods* can be considered (Belloni et al. [29], Crainic et al. [71], Zhao and Luh [262]). Such iterative methods prove also to be quite robust with respect to the tuning of the algorithm parameters; however, their possible drawback is the need to solve a quadratic programming problem at each iteration, causing an increase of computational complexity.

A reformulation of (LD) with a differentiable objective function is now considered to introduce an iterative method for solving (LD). The method is a *Kelley's cutting-plane method* (Kelley Jr. [159]) in the dual viewpoint, while in the primal prospective it is a *Column Generation* (CG) algorithm (Desaulniers et al. [79]) that uses *Dantzig-Wolfe*

Decomposition (DWD) (Dantzig and Wolfe [77]).
Let X be the feasible region of (LR), i.e., $X = \{x \in \mathbb{N}^n : Cx = d\}$. Suppose, for simplicity, that its convex hull $Conv(X)$ (i.e., the boundary of the smallest convex polygon containing X) is bounded. With this assumption, the set of extreme points x_f of $Conv(X)$ has finite cardinality F. Hence, (LD) can be equivalently rewritten as the *Lagrangian Master Problem* (LMP)

$$(LMP): \quad \max \theta \tag{1.61}$$
$$\theta \leq \gamma^T x_f + \lambda^T(b - Ax_f) \quad \forall f \in F, \tag{1.62}$$
$$\theta, \lambda \text{ free.} \tag{1.63}$$

Since (LMP) may have a large number of constraints (1.62), a constraint (cut) generation approach is devised. The Kelley's cutting plane method considers restricted (LMP) problems iteratively built by adding violated cuts, until the separation problem does not find additional violated constraints. The initial constraint set must ensure that (LMP) has a finite optimal solution.
The algorithm may also be motivated in the primal viewpoint by considering $(DLMP)$, the dual problem of (LMP)

$$(DLMP): \quad \min \sum_{f=1}^{F} (\gamma^T x_f)\alpha_f \tag{1.64}$$
$$\sum_{f=1}^{F} \alpha_f A x_f = b, \tag{1.65}$$
$$\sum_{f=1}^{F} \alpha_f = 1, \tag{1.66}$$
$$\alpha_f \geq 0 \qquad \forall f = 1, \ldots, F. \tag{1.67}$$

Each variable of $(DLMP)$ corresponds to a constraint of (LMP). The potential exponential cardinality of the variable set of $(DLMP)$ suggests a CG solution method. CG considers an initial pool of columns α_f, which constitute a restricted version of $(DLMP)$. In each iteration, the solution of (LMP) is added (as a column) in $(DLMP)$ if its reduced cost is negative. The solution of (P) is then retrieved as a convex combination of the x_f points with α_f coefficients.
Note that $(DLMP)$ can also be obtained from (P) by applying the Dantzig-Wolfe Decomposition. The principle of DWD is to reformulate (P) by replacing the variables x with the convex combination of the extreme points of $Conv(X)$: this lead to $(DLMP)$, also called master problem of DWD (Létocart et al. [171]).

1.3.3 Considerations on the Lagrangian Dual Problem Solution

Solving (LD) to optimality does not guarantee to find a feasible solution of (P), since the difficult constraints (1.54) are not part of the feasibility region of (LD). However, the values of Lagrangian multipliers have empirically proven to give indications for feasible solutions of good quality. This inspires the development of algorithms to render an optimal solution of (LD) feasible for (P): such methods are referred to as *Lagrangian heuristics* (see, e.g., Boschetti and Maniezzo [42], Caprara et al. [52], Holmberg and Yuan [139]).

An optimal solution of (LD) that is also a feasible solution for the original problem (P) is an optimal solution of (P), in case the dualized constraints are equalities. If inequalities are instead dualized, a Lagrangian dual solution may be non-optimal for (P), because the complementary slackness conditions are not automatically satisfied in this case (Guignard [124]). In any case, the optimal solution of the Lagrangian Dual problem gives a lower bound on minimization problem (P). Hence, this bound may be used in place of the bound given by linear programming relaxations in a branch-and-bound algorithm for MILPs, with the hope of being a tighter bound and improving the algorithm performance (Ribeiro and Minoux [214]). A crucial requirement for a Lagrangian-based branch-and-bound scheme is that the branching constraints do not significantly increase the computational complexity of children nodes in comparison to that of the root node (Fisher [94]). In Chapter 3, the Lagrangian Dual bound is used in a branch-and-price framework. In every node of the branch-and-price tree, a Lagrangian Dual problem with branching constraints appended is solved at optimality thanks to the cutting plane/CG method mentioned in Section 1.3.2.

1.3.4 Lagrangian Decomposition

Lagrangian Decomposition is an extension of the Lagrangian Relaxation that introduces a staircase structure in problem (P). The Lagrangian Decomposition bound dominates the Lagrangian Relaxation bound; the dominance can be strict under particular conditions (Guignard and Kim [125]).
In the Lagrangian Decomposition approach applied to formulation (P), a set of additional variables y subject to copy constraints (1.71) added to (P), namely:

$$\min \gamma^T x \tag{1.68}$$
$$\text{s.t.} \quad Ax = b, \tag{1.69}$$
$$Cx = d, \tag{1.70}$$
$$x = y, \tag{1.71}$$
$$x \in \mathbb{N}^n, \tag{1.72}$$
$$y \in \mathbb{N}^n. \tag{1.73}$$

The LR of (1.71) with Lagrange multipliers $\mu \in \mathbb{R}^n$ is

$$
\begin{aligned}
(LR_{xy}(\mu)): \quad & \min \gamma^T x + \mu^T (y - x) && (1.74)\\
\text{s.t.} \quad & Ay = b, && (1.75)\\
& Cx = d, && (1.76)\\
& x \in \mathbb{N}^n, && (1.77)\\
& y \in \mathbb{N}^n, && (1.78)
\end{aligned}
$$

which is decomposable into the two independent subproblems

$$
\begin{aligned}
(LR_x(\mu)): \quad & \min (\gamma - \mu)^T x && (1.79)\\
\text{s.t.} \quad & Cx = d, && (1.80)\\
& x \in \mathbb{N}^n. && (1.81)
\end{aligned}
$$

$$
\begin{aligned}
(LR_y(\mu)): \quad & \min \mu^T y && (1.82)\\
\text{s.t.} \quad & Ay = b, && (1.83)\\
& y \in \mathbb{N}^n. && (1.84)
\end{aligned}
$$

The *Lagrangian Decomposition Dual* (LDD) problem amounts to determine the best bound given by $(LR_{xy}(\mu))$:

$$
\begin{aligned}
(LDD): \quad & \max_{\mu \in \mathbb{R}^n} v(LD_{xy}(\mu)) = && (1.85)\\
& = \max_{\mu \in \mathbb{R}^n} \big(\min\{(\gamma - \mu)^T x : x \in X\} + \min\{\mu^T y : y \in Y\} \big), && (1.86)
\end{aligned}
$$

where $X = \{x \in \mathbb{N}^n : Cx = d\}$ and $Y = \{y \in \mathbb{N}^n : Ay = b\}$.
It is worth mentioning that in some cases, the original problem structure may directly lead to Lagrangian Decomposition when dualizing a set a linking constraints; namely, the LR problem decomposes into independent subproblems without the introduction of artificial variables. For instance, this situation arises in Vehicle Routing Problems (VRPs) (Dantzig and Ramser [76]) in which the constraints relaxed in a Lagrangian fashion are the only ones which involve more than one vehicle (see, e.g. Kohl and Madsen [164]); in other words, each constraint of LR problem is vehicle dependent and a Lagrangian Decomposition into the set of vehicles is naturally applicable. This structure is also exhibited in the MISOCP formulation considered in Chapter 3 for a class of VRPs.

1.4 Invoking Optimization Solvers

In this section, we describe the modality in which the optimization solvers have been invoked for solving a MISOCP model in Chapters 2 and 3 and linear stochastic formulations in Chapter 5.

The ways in which expressing an optimization problem and submitting it to an optimization solver may be solver dependent. If the optimization model is expressed in a programming language (e.g., C, C++, Java or Phyton), it is generally possible to embed an optimizer Application Programming Interface (API) in the code in order to call a specific solver. Solvers may also offer their own language for the implementation of models and algorithms: an example is given by the modeling/programming language Mosel [255] provided by Xpress. An alternative is to consider a solver as a stand-alone tool and call it from the command line, according to a specific syntax.

The difficulties of considering different interfaces and programming languages can be reasonably overcome by adopting modeling languages such as Algebraic Modeling Languages (AMLs) instead of programming languages. Such languages have been of large help in the MISOCP solvers comparison performed in Chapter 2. Modeling languages are essentially characterized by the following features: letting the user store the mathematical model and problem data in structures that are easily accessible from the solver; interfacing with several solvers in a simple and compact way and presenting the solver solution in a format easily understandable by the user. The solver is a tool external to the AML.

Two widely used AMLs are AMPL (Fourer et al. [95]) and GAMS (Rosenthal [216]). They both posses a well-defined syntax and have semantics very close to mathematical notation, which makes them suitable to adopt even for users with modest programming skills. In Chapter 2, GAMS has been preferred over AMPL for solving the MISOCP problem, because it can invoke a larger number of dedicated solvers.

The use of entities such as sets and indexes largely enhances the flexibility of AMLs and enables to develop even complex algorithms that use optimization solvers as black-boxes; this feature has been used to compute the stochastic measures of Chapter 5. However, when the computational time is a concern of the code developer, a programming language is generally preferred, because algorithmic instructions (such as loops on indexed sets) can be very time-consuming in an AML execution. Moreover, it should be noted that not all solver parameters have necessarily a clear equivalent in AMPL and GAMS parameters. This may be another reason for considering solver-dependent frameworks and languages for expressing optimization problems. The C programming language has been used in Chapter 2 for implementing the Benders Enumeration Algorithm and in Chapter 3 for both testing the MISOCP formulations introduced in Section 3.3 and Section 3.4 with Cplex and coding the branch-and-price algorithm described in Section 3.5. The MOSEK and Cplex APIs for C have been called in the two algorithms, respectively.

A relevant advantage of AMPL and GAMS is the availability of handlers for directly importing data from spreadsheets or databases. This is in contrast with many programming languages, such as C, for which the user is forced to express data in a text format (e.g., .txt, .csv). More precisely, AMPL under a Microsoft Windows platform is able to read parameters of the problem from a Microsoft Excel file. This feature has been used for the computational experiments on the stochastic models of Chapter 5.

1.5 Thesis Overview

This section summarizes the contributions contained in the thesis.
Chapter 2 (Gambella et al. [99]) is devoted to the study of a path and mission planning problem arising from the usage of a system of heterogeneous vehicles called Carrier-Vehicle (CV) system. The two vehicles are differing for operational capabilities and limitations in autonomy. The interaction and synchronization among them poses interesting and challenging optimization requests. The problem of visiting a set of static locations in shortest time by using the CV system is known as CV Traveling Salesman Problem (CVTSP). The chapter presents a Mixed-Integer Second Order Conic Programming (MISOCP) model for CVTSP, which is used for developing a Benders-like enumeration algorithm. Computational results compare the solutions obtained with several MISOCP solvers against the enumerative procedure. The work of the chapter was presented at the *VeRoLog 2015*.
Chapter 3 (Gambella et al. [100]) is the outcome of the internship I served in IBM Research Ireland under the supervision of Dr. Bissan Ghaddar and Prof. Joe Naoum-Sawaya. The internship project focused on a class of routing problems, called Interceptor Vehicle Routing Problems (IVRPs), which has a number of relevant applications in target tracking problems, both in civilian and military contexts and in ride-sharing or carpooling systems. The chapter presents novel mathematical formulations for IVRPs which are classified as MISOCP models. Valid inequalities and symmetry breaking constraints are introduced for helping to lower the resolution times. The main contribution of the project is constituted by a branch-and-price algorithm based on a Lagrangian relaxation of the vehicle-assignment constraints. Computational tests show the effectiveness of the branch-and-price approach over the MISOCP resolution with Cplex.
In Chapter 4 (Gambella et al. [102]) a mathematical formulation for the strategic problem of waste flow allocation in a deterministic version is presented. Original constraints are developed in order to tackle realistic requirements, such as the modeling of the operative cycle of digester facilities and the logic conditions on incoming and outgoing flow in non-disposal facilities. The resulting Mixed-Integer Linear Programming formulation is the building block of an optimization tool that is effectively used by the consulting company Optit Srl as a decision support system for Herambiente SpA, which is largest company in the waste treatment sector in Italy. Operations research techniques are fundamental for achieving cost savings and allow what-if (statistical) analysis for the considered problem. The work of the chapter was presented by Matteo

Pozzi in a preliminary version under the name *Optimization of Large-Scale Waste Flow Management at HerAmbiente* at the *AIRO 2015*.
Chapter 5 (Gambella et al. [101]) describes a problem of waste flow allocation in which the uncertainty in the waste generation amounts is explicitly considered in a Two-Stage Multiperiod Stochastic Programming formulation. The study is motivated by the availability of historical data of waste generated in some cities under the responsibility of Herambiente SpA. Optit Srl also provided the main features of the waste management network of Emilia-Romagna region. The proposed stochastic models consider a monthly waste flow allocation in a yearly planning horizon. Preliminary computational results are referred to a limited set of scenarios obtained by historical data. Standard stochastic measures such as Expected Valued of Perfect Information and Value of Stochastic Solution are reported.
Chapter 6 (Brandstätter et al. [47]) is a survey on the optimization challenges arising in electric car sharing systems. Nowadays, services of shared mobility are gaining an increasing interest and popularity. This is due both to the possibility of decreasing dangerous gas emission and to lower transport expenses. The attention on the sustainability aspect is particularly relevant in car-sharing systems that use electrical cars, giving rise to Ecar-sharing systems, opposed to conventional car-sharing services. The chapter summarizes the most relevant strategical and tactical decisions to take in building and managing an Ecar-sharing system.

Chapter 2

Exact Solutions for the Carrier-Vehicle Traveling Salesman Problem

2.1 Introduction

Path and mission planning problems (Bortoff [41], Griggs et al. [123]) are of remarkable importance in many operational scenarios with multi-vehicle systems. In challenging applications, such as environmental sampling, planetary exploration and rescue missions, single vehicle systems are typically not suitable for fulfilling complex tasks because of their limited autonomy, for example. This motivates the adoption of multi-vehicle systems (Murray [194]), in which each one has specialized skills that should be exploited in order to achieve the desired goal.

The two-vehicle system called Carrier-Vehicle (CV) system has received some attention in the latest years for the development of planning and control algorithms. This is both due to the simplicity of its representation and its wide practical applicability. The Carrier is a slow vehicle with very large travel autonomy and able to transport, deploy, recover, and service a faster Vehicle. Such a vehicle has a limited operational autonomy and is therefore subject to limitations in the stand-alone (i.e., traveling not on the carrier deck) operating time. An example of CV systems in maritime applications is a system constituted by a ship that is a service base for a helicopter or a unmanned aerial vehicle.
The path-planning Carrier-Vehicle problems generally considered in the literature require to determine a minimum-time path (also called *trajectory* hereafter) in which the CV visits a given set of points (called *target points*) by following a sequence of i) take-off, ii) target point visit and iii) landing operations. Starting and ending points of the CV route are often considered to be coincident with a point, corresponding to

the CV base.
At *take-off* points, the vehicle departs from the carrier deck and heads towards a target point. After visiting one or more target locations, the two vehicles meet at a *landing* point. Different tasks for completing a mission can be required; one of the most commonly adopted is to make the CV system reaching a carrier base after having visited all target locations. Figure 2.1 displays a feasible solution for the problem in which the starting and ending points of the CV path are different.

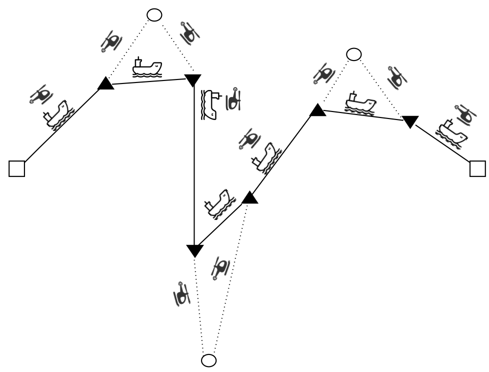


FIGURE 2.1: Schematic representation of a Carrier-Vehicle route. Squares represent the start and end location, circles are the target points, triangles are take-off and landing positions. Solid lines are the carrier paths and dotted lines are the vehicle ones.

In Garone et al. [103], the authors formulate the Carrier Vehicle Problem (CVP) in which the target point visiting sequence is determined a-priori and the vehicle can only visit a single target location in a given path between take-off and landing points. We denote such paths as *take-off/landing processes*. In fast rescue missions, the vehicle is usually required to return to the carrier after visiting a unique target point.
The CVP can be efficiently solved as a continuous convex problem. By removing the assumption for which the target visiting order is known, a different problem variant called Carrier-Vehicle Traveling Salesman Problem (CVTSP) arises. The authors provide analytical lower bounds and heuristics for CVP and CVTSP: CVTSP bounds are obtained by exploiting well-known properties of the Traveling Salesman Problem (TSP) (Dantzig et al. [73]). Conditions under which the TSP optimal solution coincides with the CVTSP optimal solution are also illustrated.
As stated in Garone et al. [104], the exact solution of CVTSP can be practically computed for instances with around 5 target points. In such cases, an exact procedure would explore every possible target visiting sequence and then compute the associated CVP cost. The CVTSP solution would be the minimum cost sequence.

Another relevant variant in the class of carrier-vehicle problems is the Generalized Carrier-Vehicle Traveling Salesman Problem (GCVTSP), introduced in Garone et al. [105]. This problem generalizes the CVTSP in the sense that the number of points to be visited in each take-off/landing process is not known in advance. A mixed-integer nonlinear convex model is formulated to solve small-sized instances (from 5 to 7 target points). In Garone et al. [106], a three-phase heuristic is presented in order to practically obtain good-quality solutions for instances with up to 100 target points. Multiple targets visits in a take-off/landing process are also considered in the recent contribution of [163], for the case in which the target visiting sequence is fixed. The authors present a Mixed-Integer Second-Order Conic Programming (MISCOP) model. The MISOCP resolution with the state-of-the-art solver Gurobi requires computational times of the order of $10^3 - 10^4 s$ to solve instances with a number of target points varying from 30 to 100.

The present work aims at solving the CVTSP to optimality. A preliminary version of this algorithm was presented at the VeRoLog 2013 Conference.
The dynamics for the vehicle and carrier are those considered in Garone et al. [103]: the vehicle speeds are constant in value, while the vehicle trajectories can range from line segments to circular arcs in order to let the vehicles synchronize at landing points. It is assumed that the fast vehicle operational capability is instantaneously restored when it lands back to the carrier deck.
The remainder of the chapter is organized as follows. In Section 2.2, a MISOCP formulation for the CVTSP is proposed. In Section 2.3, we present an exact enumeration procedure inspired by Benders' decomposition algorithm (Benders [34], Geoffrion [108]). Computational results comparing these two exact solution approaches are presented in Section 2.4, whereas some conclusions are drawn in Section 2.5.

2.2 A MISOCP Model for Solving CVTSP

In this section we present an MISOCP model for solving the CVTSP, which is an extension of the continuous model presented in [103]. The novel feature of the model is that the target visiting sequence is a decision to be taken, which is expressed by assignment variables. In addition, decision variables representing target points coordinates are introduced. The input parameters of the problem are:

n	number of target points	
q_i	set of target points coordinates in $\mathbb{R}^2$	$i = 1, \dots, n$
q_{min}	vector of the minimum of the target point coordinates	
q_{max}	vector of the maximum of the target point coordinates	
V_v	vehicle speed	
V_c	carrier speed	
a	vehicle autonomy (in time units)	
p_o	coordinates in $\mathbb{R}^2$ of the starting point of the trajectory	
p_f	coordinates in $\mathbb{R}^2$ of the ending point of the trajectory.	

The decision variables are:

Q_i	coordinates in $\mathbb{R}^2$ of the i-th target point to be visited	$i = 1, \dots, n$
w_{ij}	binary variable taking value 1 if target point j is visited in position i (i.e., $Q_i = q_j$)	$i, j = 1, \dots, n$
$p_{to,i}$	coordinates in $\mathbb{R}^2$ of the take-off point for the visit of Q_i	$i = 1, \dots, n$
$p_{l,i}$	coordinates in $\mathbb{R}^2$ of the landing point after the visit of Q_i	$i = 1, \dots, n$
$t_{i,1}^{to,l}$	time taken by the vehicle to reach Q_i from $p_{to,i}$	$i = 1, \dots, n$
$t_{i,2}^{to,l}$	time taken by the vehicle to reach $p_{l,i}$ from Q_i	$i = 1, \dots, n$
$t_i^{to,l}$	time taken by the carrier to reach $p_{l,i}$ from $p_{to,i}$	$i = 1, \dots, n$
$t_1^{l,to}$	time taken by the carrier to reach $p_{to,1}$ from p_o	
$t_i^{l,to}$	time taken by the carrier to reach $p_{to,i}$ from $p_{l,i-1}$	$i = 2, \dots, n$
$t_{n+1}^{l,to}$	time taken by the carrier to reach p_f from $p_{l,n}$	

A formulation for the CVTSP is then given by the following model.

$$z_{CVTSP} = \min \sum_{i=1}^{n} t_i^{to,l} + \sum_{i=1}^{n+1} t_i^{l,to} \tag{2.1}$$

$$s.t.$$

$$\|Q_i - p_{to,i}\| \leq V_v\, t_{i,1}^{to,l} \qquad \forall\ i = 1, \dots, n \tag{2.2}$$

$$\|Q_i - p_{l,i}\| \leq V_v\, t_{i,2}^{to,l} \qquad \forall\ i = 1, \dots, n \tag{2.3}$$

$$\|p_{to,i} - p_{l,i}\| \leq V_c\, t_i^{to,l} \qquad \forall\ i = 1, \dots, n \tag{2.4}$$

$$\|p_o - p_{to,1}\| \leq V_c\, t_1^{l,to} \tag{2.5}$$

$$\|p_{l,i-1} - p_{to,i}\| \leq V_c\, t_i^{l,to} \qquad \forall\ i = 2, \dots, n \tag{2.6}$$

$$\|p_f - p_{l,n}\| \leq V_c\, t_{n+1}^{l,to} \tag{2.7}$$

$$t_{i,1}^{to,l} + t_{i,2}^{to,l} \leq t_i^{to,l} \qquad \forall\ i = 1, \dots, n \tag{2.8}$$

$$Q_i = \sum_{j=1}^{n} w_{i,j} q_j \qquad \forall\ i = 1, \dots, n \tag{2.9}$$

$$\sum_{j=1}^{n} w_{i,j} = 1 \qquad \forall\ i = 1, \dots, n \tag{2.10}$$

$$\sum_{i=1}^{n} w_{i,j} = 1 \qquad \forall\ j = 1, \dots, n \tag{2.11}$$

$$t_{i,1}^{to,l} \geq 0 \qquad \forall\ i = 1, \dots, n \tag{2.12}$$

$$t_{i,2}^{to,l} \geq 0 \qquad \forall\ i = 1, \dots, n \tag{2.13}$$

$$0 \leq t_i^{to,l} \leq a \qquad \forall\ i = 1, \dots, n \tag{2.14}$$

$$t_i^{l,to} \geq 0 \qquad \forall\ i = 1, \dots, n+1 \tag{2.15}$$

$$q_{min} \leq Q_i \leq q_{max} \qquad \forall\ i = 1, \dots, n \tag{2.16}$$

$$w_{ij} \in \{0, 1\} \qquad \forall\ i, j = 1, \dots, n. \tag{2.17}$$

The objective function (2.1) is the mission completion time, namely the time required by the carrier to travel between consecutive take-off and landing points until it reaches the final destination.
Constraints (2.2) and (2.3) define the time spent by vehicle to perform a take-off/landing process. More precisely, the time required to get to the target location from the take-off position is computed in (2.2), while (2.3) express the time taken to return to the carrier deck after leaving the target position.
Similarly, constraints (2.5)-(2.7) model the time the carrier requires to travel between consecutive take-off and landing positions.
Inequalities (2.8) express the synchronization between the carrier and the vehicle trajectories at landing points.
The sequencing (or assignment) variables are subject to constraints (2.9)-(2.11), which impose that target position variables assume all target point coordinates with no repetitions.
The time variables are required to be non-negative in bounds (2.12)-(2.15); in particular, bound (2.14) takes into account the limited stand-alone autonomy of the vehicle.
Target points variables can be safely limited as bound (2.16) prescribe: although redundant, these conditions may help optimization solvers to handle the model. Finally, bounds (2.7) express the requirement for assignment variables to be binary.

Model (2.1)-(2.17) is an MISOCP. Indeed, it is constituted by a linear objective function (2.1), second-order conic constraints (2.2)-(2.7) and linear constraints (2.8)-(2.11). Since both the number of variables and constraints is polynomial in n, as a first attempt we used state-of-the-art optimization solvers as an exact solution strategy for the model.
The results presented in Section 2.4 were obtained by using solvers with algorithms tailored for conic problems. We also performed a preliminary testing by using global optimization solvers, such as COUENNE ([30]), ANTIGONE (Misener and Floudas [192]) and BARON ([220]). Such solvers do not identify the conic structure in constraints (2.2)-(2.7); therefore they treat them as general non-convex constraints and they perform a term-by-term convexification. Hence, the global solvers prove to be extremely slow in finding provably optimal solutions, even for very small-sized instances with five target points.

2.3 A Benders-like Enumeration Procedure for Solving CVTSP

The CVTSP can be considered as a nonlinear extension of the well-known Traveling Salesman Problem (TSP). When restricting CVTSP to the case in which $p_o = p_f$, the resulting TSP is the problem of determining a minimum-cost Hamiltonian circuit for visiting the set $\{1, \ldots, n\} \cup \{p_o = p_f\}$.

In the CVTSP formulation (2.1)-(2.17) a mixed structure is present. On the one hand, the *combinatorial* problem is that of selecting the sequence in which the target points are visited by the vehicle. On the other hand, the *continuous* problem is a CVP. As already observed in [103], the CVP can be formulated as a convex continuous model. The problem actually turns out to be a Second-Order Conic Programming (SOCP) problem, which is efficiently solvable with dedicated solvers.
In this section, we propose a Benders-like Enumeration Algorithm (BEA) for finding an optimal solution for CVTSP.
The BEA is an iterative method in which the master problem is identified with the combinatorial problem and the slave problem is the CVP. At each iteration, selecting a target visiting order gives a lower bound on the CVTSP, while solving the CVP means solving the feasibility problem of CVTSP. The general structure of our algorithm is as follows.

Algorithm 1

1. Set $UB = +\infty$, $LB = 0$, $k = 1$.
2. Generate the k-th target visiting order ord according to the list of associated lower bound $LB(ord)$.
3. $LB = LB(ord)$.
4. Check if the termination criterion $LB \geq UB$ is satisfied; if $LB \geq UB$ then stop.
5. Solve the CVP with ord target visiting sequence. The solution yields an upper bound $UB(ord)$.
6. If $UB(ord) < UB$ then $UB = UB(ord)$.
7. Set $k = k + 1$.
8. Go to Step 2.

When the termination criterion is satisfied, then UB is the optimal solution value z_{CVTSP}. Section 2.3.1 describes how a TSP sequence determined in Step 2 of Algorithm 1 can provide a valid lower bound for CVTSP.
The algorithm effectiveness is strongly influenced by the actual implementation of its main components, namely:

- The combinatorial lower bound to be computed at each master problem iteration.
- How to rank the target visiting orders according to the associated lower bound value.

Such components are responsible both for the convergence speed of the algorithm, measurable in terms of the number of TSP sequences to enumerate, and of the required computing time. It is important to point out that the target visiting sequences to enumerate are those sequences ord for which $LB(ord) < z_{CVTSP}$. Therefore, the tightness of the bound is directly related to the number of iterations required to prove CVTSP optimality.

2.3.1 Combinatorial Lower Bound for the CVTSP

Our lower bound for CVTSP is based on that for the CVP presented in [103]. Let ord be a target visiting order and $TSP(ord)$ be the length, measured in spatial distance, of the Euclidean Hamiltonian path starting from p_o, ending at p_f and visiting all target points in the sequence ord. We will refer to $TSP(ord)$ as the TSP value of the target visiting sequence ord.

A lower bound for the CVP optimal value with ord target visiting sequence is given by

$$LB(ord) = \frac{TSP(ord)}{V_c} - n\frac{V_v a}{V_c} + na, \tag{2.18}$$

where

- $nV_v a$ is the maximum distance that the vehicle can cover in all take-off/landing processes;
- $n\frac{V_v a}{V_c}$ corresponds to the time spent by the carrier to cover the maximum distance covered by the vehicle;
- $\frac{TSP(ord)}{V_c} - n\frac{V_v a}{V_c}$ is the minimum time that the fast vehicle may spend on the carrier deck;
- na: maximum time for the fast vehicle outside the carrier deck.

The quantity $LB(ord)$ is a non-decreasing function of the TSP value of the sequence ord, therefore an equivalent ranking criterion for the target visiting orders is to refer at their TSP values.
We now show that generating target visiting sequences in order of ascending $LB(ord)$ values in the master problem of Algorithm 1 ensures that (2.18) is not greater than z_{CVTSP} within the iterative procedure. Consider the following proposition.

Proposition 1.
Let ord^* be the target point visiting sequence in an optimal CVTSP solution. Then, for each visiting order ord such that $LB(ord) \leq LB(ord^*)$, $LB(ord)$ is lower than z_{CVTSP}.

Proof.
Let ord be such that $LB(ord) \leq LB(ord^*)$. Then:

$$LB(ord) \leq LB(ord^*) \leq CVP(ord^*) = z_{CVTSP}.$$

□

The tightness of (2.18) is strongly dependent on the target points positions and the CV system parameters. Indeed, when the target points are quite close to each other, the value of $TSP(ord)$ has a little impact on $LB(ord)$. The same situation occurs when V_v is considerably large. Note that, given a CVTSP instance with fast vehicle speed V_v and carrier vehicle speed V_c, an equivalent instance is obtained by dividing speeds of a factor f and by dividing the spatial distances among points by the same f. Therefore, in Section 2.4 we tested the algorithm on instances with different geometries.

2.3.2 Ranking of TSP Solutions

To rank symmetric TSP solutions in order of non-decreasing TSP value, we adopted the basic version of the enumeration procedure described by Lawler [168], which ranks the feasible solutions of a discrete binary optimization problem according to their objective function values. Such iterative method consists of a branching strategy in the solution space. Every node is a subproblem of the original problem, in the sense that it is generated from binary impositions on a subset of the variables. At each iteration k, the branching rule has the effect of excluding the k-th best solution from further consideration.
The ranking of TSP solutions according to the Lawler's procedure requires to have an efficient method for solving TSP subproblems.

Since we wanted to solve instances with up to 20 targets, we used as a black-box the enumeration code with pruning in Chapter 1 of Applegate et al. [9], denoted as *ACDJ code* in the following. In ACDJ code, the TSP solution space is represented as a tree of partial permutations in which the level of a node is the number of vertices included in the permutation; while at the root node no vertex is selected, the leaf nodes correspond to complete tours. Given a cost matrix, the algorithm computes an upper bound on the TSP optimal value with a nearest-neighborhood algorithm (see, e.g., a recent paper by Hurkens and Woeginger [144]) and then explores the space of permutations on TSP nodes with the aim of determining a minimum-cost one. A speed-up of the computational time is obtained by pruning nodes according to bounding rules. The method implemented for pruning is to compute a lower bound on the completion of the partial tour at the current node: if the sum of the partial cost and of the lower bound exceeds the best known upper bound on the optimal value, then the current node is fathomed. The lower bound chosen by the authors is the well-known minimum spanning tree bound (Held and Karp [132]).

The Traveling Salesman Problem can be modeled as a discrete optimization problem where integer variables are the binary arc variables (Dantzig et al. [73]). We simulated the fixing of TSP binary variables by modifying the edge Euclidean costs appropriately: forbidding edge (i, j) in the solution is equivalent to change its original cost c_{ij} to a considerably high cost C (e.g., $2,000$ times the maximum cost edge); analogously, including (i, j) in the solution is obtained by setting $c_{ij} = -C$.
In addition, whenever a vertex i is connected with two vertices k_1, k_2 due to the edge impositions, we forbid edges (i, j) with $j \neq k_1, k_2$. Within these settings, an infeasible TSP subproblem is associated with tours with at least one edge having cost C. Since the *ACDJ* code is not naturally able to recognize the possible infeasibility of a TSP subproblem, the feasibility of the minimum-cost permutation obtained will be determined by the absence of edges with cost C. Our computational experience shows that the time required to run *ACDJ* algorithm on infeasible subproblems can be several orders of magnitude larger than that required to solve feasible ones. This motivated us to apply techniques for detecting the infeasibility of symmetric TSP subproblems before calling the *ACDJ* code.

2.3.2.1 Detecting the Infeasibility of a Symmetric TSP Subproblem

As stated in Section 2.3.2, a method for speeding up the time required for the master problems of our BEA is to efficiently detect if a symmetric TSP subproblem is infeasible or not. The question can be formulated on a graph as follows.

Problem 1. *Given a sparse graph $G = (V, E)$ and a subset $E' \subset E$, determine if a Hamiltonian circuit including all edges in E' exists in G.*

The missing edges in graph G represent the edges forbidden in the subproblem. The edges in set E' are instead the edges included in the solution. We implemented three different algorithms to address Problem 1. The methods, called V1, V2 and V3, differ in the increasing aggressiveness in the detection of infeasibilities. Only method V3 is able to prune all infeasible subproblems, so the other versions check for the presence of an edge with cost C in the minimum-cost permutation given by the *ACDJ* code.
The code versions V1 and V2 implement two algorithms for inspecting the graph G of the current subproblem, after its edge costs matrix has been generated according to Lawler's procedure.

The first check performed (version V1) is searching in G for vertices of degree greater than 2 with respect to edges in E'. Clearly, such situation violates the degree constraints on TSP nodes and therefore the subproblem is infeasible.

After this check, the second version V2 investigates the presence of subtours using a Depth-First Search (DFS) algorithm (Tarjan [235]). In graphs in which every node is connected to at most two vertices, the connected components of the graph are disjoint paths. The depth-first exploration will find a subtour in the graph if there exists a connected component which is a circuit.

In version V3, the remaining infeasible subproblems are detected by using Constraint Programming (CP) techniques.
After checking the degree condition of version V1, version V3 invokes the $MakeCircuit$ constraint of the software suite *or-tools* of [122]. Given a set of variables with finite domain, the $MakeCircuit$ function determines the presence of complete Hamiltonian paths on the variable set by using filtering algorithms for the *Circuit* constraint (Benchimol et al. [33], Kaya and Hooker [156]). Note that in the CP setting, the value of a variable is its successor node in the Hamiltonian path: since subproblem graph G is not oriented, the setting of the CP problem equivalent to solve Problem 1 in G is not straightforward. After determining the TSP subproblem according to Lawler's branching rules and checked the degree condition, the following CP model is solved in feasibility version. The set of variables is given by the set $\{1, \dots, n\} \cup \{p_o = p_f\}$ plus convenient auxiliary vertices. The domain $D(i)$ of variable i is determined according to its degree with respect to edge set E' in graph G. Namely,

- If degree of i is 0, then $D(i) = \{j \in V : (i, j) \in E\}$.
- If degree of i is 1, then two situations are considered. Let $k = \{j \in V : (i, j) \in E'\}$.
  - If edge (i, k) is a connected component of the subgraph (V, E') (i.e., the degree of k is 1), then

$$D(i) = \{j \in V : (i, j) \in E\} \cup \{ik\},$$

$$D(k) = \{j \in V : (k, j) \in E\} \cup \{ik\},$$

    with $ik \notin V$ auxiliary vertex, $D(ik) = \{i, k\}$.
  - If edge (i, k) is not a connected component of the subgraph (V, E'), then $D(i) = \{k\} \cup \{j \in V : (i, j) \in E\}$.
- If degree of i is 2, then $D(i) = \{j \in V : (i, j) \in E'\}$.

For issues regarding the compatibility of *ortools* functions, the code version V3 was run on a different machine with respect to versions V1 and V2, so the computational results are not fully comparable and we prefer not to report them in detail in Section 2.4. Nevertheless, based on a rough conversion between the two machines, for instances with 14 or 15 target points, V3 proves to be remarkably faster than version V1 and V2. However, in smaller-sized instances, invoking the filtering algorithm for the *Circuit* constraint generally slows down the overall procedure.

2.4 Computational Results

In this section, we present computational results on four sets of CVTSP instances inspired by those proposed by Garone et al. [104]. The instances are available on request from the authors. All instances have the following Carrier-Vehicle parameters: $V_c = 1$, $V_v = 5$ and $a = 1$. For all sets of instances, the number of target points varies from 10 to 15.
The four sets are divided in two groups, which differ by the size of the area in which the target points are distributed. More precisely, in the first group, including sets called SD and MD, the target points coordinates are generated from a uniform distribution in the $[-25, 25] \times [-25, 25]$ box. In addition, for set MD, a minimum distance of $V_v a$ between target points is also imposed to evaluate the impact on the quality of lower bound (2.18). Similarly, in the second group, which includes sets called LD and VLD, the target points are generated in the $[-50, 50] \times [-50, 50]$ rectangle and, for VLD instances, the minimum distance condition is also imposed.
All runs were performed on a QEMU Virtual CPU version 0.14.1 @ 2.40 GHz (Cluster). One core in an isolated node was used. A time limit of 1 hour on BEA and on the solver executions was imposed.
In the following, we describe the results of the testing of the BEA on the four sets of instances. We first discuss the overall performance of the BEA by comparing the results that can be obtained with the two versions V1 and V2 of the infeasibility detection methods described in Section 2.3.2.1. We next compare the Benders-like approach with the direct solution of the model (2.1)-(2.17) with some optimization solvers.

2.4.1 Results of the Benders-like Enumeration Algorithm

The BEA has been implemented in *C*. The CVP slave problems have been solved using MOSEK C API 7.1.0.30 ([8]), which, according to preliminary testing, showed performance comparable to other SOCP solvers.

The results of the two versions of the BEA for CVTSP are presented in Tables 2.1 and 2.2. In particular, Table 2.1 summarizes the results for instances SD and MD, while Table 2.2 regards instances LD and VLD. In both tables, the columns have the following meaning:

- Instance: instance name in the format *type N_id*.

The instance *type* is SD, MD, LD or VLD. The number N is the number of target points plus the initial (coincident with the final) point of the CV trajectory. In our computational tests, N ranged from 11 to 16. The number *id* is an identifier of the instance; three instances for each value of N have been considered.

- BestUB: best upper bound computed by the BEA version V2; if the instance is solved to optimality within the time limit, then it is marked with an asterisk.
- #CVTSP: number of the TSP solution whose target visiting order is that of BestUB.
- #TSPSolV1: number of TSP solutions enumerated by BEA in version V1.
- #Subprob: number of TSP subproblems generated by the Lawler's procedure.
- #ACDJ: number of TSP subproblems solved with ACDJ code in version V1.
- #Feas: number of TSP subproblems certified as feasibile after the ACDJ code call.
- tV1: elapsed time in seconds for completing the resolution procedure in V1 (when time limit is reached, TL is inserted).
- #TSPSolV2: number of TSP solutions enumerated in version V2.
- #Subtour: number of TSP subproblems detected as infeasible by the DFS algorithm.
- tV2: elapsed time in seconds for completing the resolution procedure in V2.

After reporting the results on a type of instance, the line Average*type* shows the average gaps and average computational times for both versions V1 and V2.

Instance	BestUB	#CVTSP	#TSPSolV1	#Subprob	#ACDJ	#Feas	tV1	#TSPSolV2	#Subtour	tV2
SD11_1	108.757183*	1	318	9938	3682	2339	2.16	=#TSPSolV1	292	1.96
SD11_2	113.919302*	3	2437	68190	25031	13348	9.17	=#TSPSolV1	1575	8.62
SD11_3	125.185513*	1	418	14769	5924	4136	2.20	=#TSPSolV1	540	1.93
SD12_1	107.603481*	1	249	11329	4052	2964	3.09	=#TSPSolV1	336	2.28
SD12_2	153.936018*	2	192	7085	2608	1709	2.40	=#TSPSolV1	228	1.96
SD12_3	118.606354*	4	1077	44911	19161	12820	10.28	=#TSPSolV1	1187	8.76
SD13_1	116.117180*	1	341	14279	4594	2991	8.88	=#TSPSolV1	280	4.12
SD13_2	136.855216*	3	14445	531807	170582	99326	160.21	=#TSPSolV1	14656	137.52
SD13_3	121.472522*	1	1357	49334	13077	7605	19.32	=#TSPSolV1	1153	13.11
SD14_1	128.136957*	7	6168	283775	89961	62938	281.06	=#TSPSolV1	7072	201.05
SD14_2	124.662960*	3	4034	191564	63920	43886	254.30	=#TSPSolV1	3730	152.53
SD14_3	138.071040*	1	1500	78996	28438	20382	163.42	=#TSPSolV1	1834	103.62
SD15_1	123.668010*	2	7527	336761	93147	60564	882.99	=#TSPSolV1	5328	403.20
SD15_2	136.101252*	3	5128	302986	96939	71800	1966.39	=#TSPSolV1	5495	643.02
SD15_3	132.717432	401	37259	1977642	560497	425224	TL	59506	63472	TL
SD16_1	145.096484	1	189	12983	4550	3416	TL	2865	3320	TL
SD16_2	155.355044	4	35	3428	1378	1209	TL	1147	1716	TL
SD16_3	128.384932	4	659	45781	17414	13613	TL	3872	4025	TL
AverageSD							1009.22			893.54
MD11_1	146.850721*	1	36	1209	472	330	0.20	=#TSPSolV1	21	0.19
MD11_2	132.116278*	1	69	2382	1005	743	0.63	=#TSPSolV1	22	0.60
MD11_3	133.416279*	4	121	4061	1680	1139	0.84	=#TSPSolV1	80	0.76
MD12_1	157.736159*	4	178	7584	3382	2355	2.89	=#TSPSolV1	195	2.15
MD12_2	165.631428*	5	172	7431	3405	2376	2.46	=#TSPSolV1	227	1.98
MD12_3	121.236980*	1	200	7590	2829	2002	1.95	=#TSPSolV1	158	1.58
MD13_1	150.894158*	4	915	37016	11108	7632	22.36	=#TSPSolV1	1126	15.25
MD13_2	130.994466*	2	819	35636	13025	9076	25.03	=#TSPSolV1	593	16.92
MD13_3	150.368207*	2	431	22757	8831	6631	22.68	=#TSPSolV1	743	12.72
MD14_1	146.951135*	2	702	39121	14325	10642	94.43	=#TSPSolV1	782	24.23
MD14_2	163.589027*	1	1888	75598	17723	10250	108.24	=#TSPSolV1	1331	54.06
MD14_3	153.005893*	3	752	34235	10963	7796	85.86	=#TSPSolV1	697	38.34
MD15_1	168.238198*	1	15868	865021	297700	215916	2838.08	=#TSPSolV1	16985	1476.65
MD15_2	136.935306*	1	4922	263463	88032	64752	2388.48	=#TSPSolV1	5038	1277.98
MD15_3	157.863700*	1	7435	386263	125229	86265	1838.00	=#TSPSolV1	9340	624.48
MD16_1	166.214182	1	451	33377	13103	10537	TL	3841	3857	TL
MD16_2	177.234610*	2	652	48982	20464	16518	TL	3380	4933	3232.09
MD16_3	164.369465	1	140	13132	5911	5010	TL	1004	2220	TL
AverageMD							1012.90			776.67

Table 2.1: BEA - Instances SD and MD - Infeasibility detection versions V1, V2

Instance	BestUB	#CVTSP	#TSPSolV1	#Subprob	#ACDJ	#Feas	tV1	#TSPSolV2	#Subtour	tV2
LD11_1	311.548775*	1	3	153	84	66	0.07	=#TSPSolV1	7	0.06
LD11_2	345.194615*	1	21	976	396	307	0.29	=#TSPSolV1	42	0.23
LD11_3	299.534474*	1	19	787	397	281	0.23	=#TSPSolV1	15	0.21
LD12_1	296.067765*	1	9	416	160	126	0.30	=#TSPSolV1	7	0.29
LD12_2	308.263255*	1	81	3194	1230	783	0.97	=#TSPSolV1	74	0.69
LD12_3	270.309649*	1	12	597	294	236	0.17	=#TSPSolV1	12	0.16
LD13_1	261.636939*	1	68	3034	1140	787	2.16	=#TSPSolV1	107	1.43
LD13_2	294.854881*	1	67	3756	1607	1248	4.63	=#TSPSolV1	47	2.16
LD13_3	307.338625*	2	55	2931	1226	907	4.48	=#TSPSolV1	82	2.39
LD14_1	319.795018*	1	88	4865	1843	1369	22.38	=#TSPSolV1	111	5.84
LD14_2	282.914903*	1	436	23055	9002	6455	67.20	=#TSPSolV1	653	23.94
LD14_3	301.595062*	1	101	5175	1561	1066	18.26	=#TSPSolV1	148	5.35
LD15_1	299.036907*	2	216	14012	4986	4160	86.36	=#TSPSolV1	181	45.95
LD15_2	314.007695*	2	196	14030	5078	3886	473.73	=#TSPSolV1	433	160.90
LD15_3	324.791195*	1	1862	108119	36977	27884	1129.15	=#TSPSolV1	2622	440.25
LD16_1	322.052535*	1	61	5139	2172	1768	1111.27	=#TSPSolV1	117	350.77
LD16_2	338.703722*	1	122	10490	3406	2882	TL	727	986	2726.22
LD16_3	353.866958*	5	625	44918	17357	13930	TL	1042	1231	2403.94
AverageLD							562.31			342.82
VLD11_1	257.236771*	1	34	1297	622	470	0.33	=#TSPSolV1	28	0.31
VLD11_2	324.746866*	1	7	317	153	118	0.09	=#TSPSolV1	7	0.08
VLD11_3	226.129466*	1	18	787	337	263	0.17	=#TSPSolV1	30	0.14
VLD12_1	326.035651*	1	12	621	292	226	0.27	=#TSPSolV1	29	0.14
VLD12_2	274.192197*	1	17	787	318	250	0.22	=#TSPSolV1	11	0.21
VLD12_3	281.940457*	1	20	1089	458	364	0.72	=#TSPSolV1	34	0.50
VLD13_1	316.709241*	3	183	8621	3452	2480	7.12	=#TSPSolV1	151	4.26
VLD13_2	239.259399*	1	69	3949	1815	1413	2.93	=#TSPSolV1	99	1.56
VLD13_3	281.329381*	1	40	2153	962	738	1.68	=#TSPSolV1	78	0.98
VLD14_1	319.627905*	1	205	10726	3820	2977	32.05	=#TSPSolV1	254	11.69
VLD14_2	300.167644*	2	84	4435	1458	1128	8.30	=#TSPSolV1	92	2.38
VLD14_3	280.366264*	1	28	1826	705	562	6.71	=#TSPSolV1	63	3.79
VLD15_1	295.334042*	1	17	1245	525	450	104.58	=#TSPSolV1	8	2.18
VLD15_2	314.700103*	2	131	9566	3490	2846	152.51	=#TSPSolV1	333	50.91
VLD15_3	264.945505*	1	70	5096	2090	1612	83.24	=#TSPSolV1	115	5.16
VLD16_1	379.909039*	1	227	19081	6055	4917	TL	323	611	624.76
VLD16_2	355.421988*	1	241	19229	7282	6269	TL	511	384	861.46
VLD16_3	305.994558	4	132	12048	5401	4441	TL	1141	2161	TL
AverageVLD							622.27			287.25

Table 2.2: BEA - Instances LD and VLD - Infeasibility detection versions V1, V2

In groups SD and MD, the target points have been generated in a smaller box with respect to LD and VLD. This makes the lower bound (2.18) weaker in SD and MD; therefore, in general these are the hardest instances for BEA in the sense that a larger number of TSP solutions is needed to prove CVTSP optimality.
The code version V1 is able to solve to optimality 60 instances out of 72 within the time limit. The most difficult instances to solve are those with 15 target points (SD16_∗, MD16_∗) and also the SD15_3, in which the lower bound is particularly weak. The difficulties in the largest instances are due to the scarce aggressiveness of V1 in detecting infeasible subproblems and to the computational time required by the Lawler's ranking procedure. In spite of its size, instance LD16_1 requires only 61 target visiting sequences to rank and it is solved within time restrictions.

Using version V2 generally speeds up the required time for completing the BEA. The time savings are more evident in the largest instances, for which detecting even a relatively small number of subproblems with subtours yields a considerable decrease of computational time (see, e.g., instance VLD16_2). Indeed, the TSP subproblems with a subtour in graph (V, E') are represented by a matrix cost with a high number of entries with value C. This results in having large-cardinality sets of permutations with huge costs, so the pruning in V1 is rarely applicable.
Version V2 improves V1 results also in terms of 5 additional instances solved to optimality. When the time limit is reached in V2, the number of ranked TSP solutions is considerably higher than that of permutations enumerated by V1. The average computational times reported for version V2 highlight the relationship between geometry of the instance type and easiness of solution.

2.4.2 Comparison Between BEA Version V2 and Optimization Solvers

The MISOCP model (2.1)-(2.17) has been written in GAMS 24.4.2. The model has been solved with five different optimization solvers: Cplex ([70]), Gurobi ([128]), SCIP (Achterberg [4]), MOSEK (Andersen and Andersen [8]) and Xpress ([256]). Apart from MOSEK, all solvers require the second-order conic constraints to be written in the equivalent quadratic form obtained by squaring both sides of the constraint.
Table 2.3 indicates the GAMS solver versions used.

Solver	Version
Cplex	12.6.1.0
Gurobi	6.0.2
SCIP	3.1(67d713c)
MOSEK	7.1.0.24
Xpress	27.01.02

Table 2.3: GAMS solver link versions

We note that, to let the optimization solvers obtain optimal solutions with a precision comparable to BEA, it is necessary to set the GAMS relative optimality tolerance to a threshold of 1.00E-6: computational testing showed that the default value 1.00E-1 led the solvers to select not-optimal target visiting order in some cases. Regarding the BEA, as reported in column #CVTSP of Tables 2.1 and 2.2, it is very likely that an optimal CVTSP sequence is quickly found, while the main computational effort is needed for proving its optimality.
Except for the relative optimality tolerance, the solvers are invoked with their default algorithmic settings.
Table 2.4 shows indicators of precision and performance of BEA algorithm V2 and of different solvers in the GAMS implementation. For each algorithm, the average relative gap (GapAvg) with the best solution obtained among all methods and the average resolution time (tAvg, in seconds) are reported for instances types SD, MD, LD, VLD. We note that, even if solvers select the same optimal sequence of BEA, the objective function values may differ for quantities of up to the order of magnitude of 1.00E-4. The small difference in take-off and landing optimal coordinates determined by different solvers and BEA is motivated by the approximation in floating points operations applied by each method.
Regardless of the instance type, the optimization solvers are not generally able to solve to optimality the instances with 14-15 target points within the time limit of 1 hour. Cplex, SCIP and MOSEK resolutions also meet the time limit for some 13-target points instances. This explains the larger average gaps obtained by the GAMS solvers. The superiority of BEA in the considered instances is clear both in terms of precision and performance.

		SD	MD	LD	VLD
BEA V2	GapAvg	9.10E-07	1.61E-07	1.26E-07	6.06E-08
	tAvg	893.54	776.67	342.82	287.25
Cplex	GapAvg	1.86E-02	1.88E-02	1.63E-02	1.78E-02
	tAvg	1509.92	2258.14	1930.70	1771.82
Gurobi	GapAvg	4.42E-03	6.19E-03	6.06E-03	6.03E-03
	tAvg	1330.32	1455.37	1555.13	1411.52
SCIP	GapAvg	9.80E-03	1.98E-02	1.66E-02	2.05E-02
	tAvg	1408.44	1814.22	1726.96	1800.70
MOSEK	GapAvg	1.30E-03	3.89E-03	6.00E-03	2.50E-03
	tAvg	1557.67	1726.17	1769.89	1651.62
Xpress	GapAvg	3.37E-03	7.22E-03	5.77E-03	3.09E-03
	tAvg	1290.28	1491.36	1422.62	1407.42

Table 2.4: Solvers comparison for the groups SD, MD, LD, VLD

Table 2.5 summarizes the number of instances not solved to optimality in each algorithm for the considered groups of 18 instances each. Even for the hardest instances for BEA (i.e., SD, MD), our algorithm is impressively faster than the optimization solvers.

The difference between BEA and GAMS solvers becomes more evident for instances of type LD and VLD.

Solver	SD	MD	LD	VLD
BEA V2	4	2	0	1
Cplex	6	7	8	6
Gurobi	5	4	4	5
SCIP	4	5	7	6
MOSEK	6	5	6	5
Xpress	5	4	5	4

Table 2.5: Time limit instances for each algorithm

2.5 Conclusions

In this chapter, we considered the Carrier-Vehicle Traveling Salesman problem in fast rescue mission situations. We proposed a MISOCP formulation and a Benders-like enumerative algorithm for practically solving problem instances with up to 15 target points. These results improve the previous exact methods proposed in the literature, which solved instances with less than half the number of target points.
Future research directions consist in finding stronger CVTSP lower bounds, also for instances with a weak relationship with the associated TSP.

Acknowledgements

We thank Pietro Belotti for his pieces of advice on handling conic constraints in numerical optimization solvers, William W. Cook for sharing his TSP code and Emanuele Garone for the help in gaining insights in the previous studies on CVTSP.
The research has been partially supported by PRIN 2012.

Chapter 3

The Interceptor Vehicle Routing Problem: Formulation and Branch-and-Price Algorithm

3.1 Introduction

The family of Vehicle Routing Problems (VRPs) has received remarkable attention in the field of combinatorial optimization after its introduction in the paper of Dantzig and Ramser [76]. VRPs determine a set of vehicle routes in order to accomplish transportation requests at minimum cost.
Routing problems are constantly considered as hot topics for decision makers for several reasons. For instance, the range of practical situations that they can cover and the difficulty in developing efficient algorithms for finding optimal or even sub-optimal solutions, especially when modeling real-life applications. For a survey on state-of-the-art methods for solving routing problems and considering practical issues, the reader is referred to Toth and Vigo [240].

The focus of this chapter is to formulate and solve a dynamic variant of VRP, which we refer to as Interceptor Vehicle Routing Problem (IVRP). The problem determines a set of vehicle routes in order to intercept a set of moving target points in the Euclidean plane where the targets are moving over time according to a known motion. Although the problem has a dynamic nature, the complete knowledge of the targets' motion permits the adoption of *a priori* optimization techniques.

IVRPs have applications in several areas. For instance, an interesting case arises when the target points are people requiring a means of transport for reaching a common destination, such as employees of the same company (see Figure 3.1): in this context, optimization problems in carpooling and ride-sharing services have been formulated (see, e.g., Aïvodji et al. [5], Bruck et al. [48], Bit-Monnot et al. [39] and Varone and Aissat [245]).

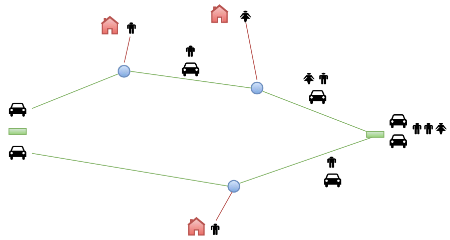


FIGURE 3.1: A ride-sharing system with two vehicles and three customers

Another area of practical applications is related to target tracking missions conducted by Unmanned Aerial Vehicles (UAVs). The UAVs can be successfully applied in military and civilian contexts for example for surveillance, defence, security, reconnaissance, weather monitoring, pollutant estimation (see, e.g., Sundar and Rathinam [234], Mallick et al. [186]) and aerial refuelling (Barnes et al. [23] and Thomas et al. [236]). The rest of the chapter is organised as follows. In Section 3.2 we review the literature that deals with problems similar to IVRP, mainly concerned with single-vehicle situations. In Section 3.3 we present a mathematical model for the general case of IVRP. Section 3.4 adjusts the general model to the relevant case in which the target points are moving along a predefined line. Valid cuts for strengthening the IVRP formulation are proposed in Section 3.4.1. Section 3.5 illustrates the Lagrangian Decomposition of the problem, which leads to a Branch-and-Price (B&P) approach. Section 3.6 discusses implementation details and presents computational results comparing Cplex and the B&P algorithm on a set of randomly generated instances. Finally, Section 3.7 provides a brief conclusion and future research directions.

3.2 Literature Review

The Traveling Salesman Problem (TSP) is one of the most important and studied combinatorial problems. The TSP requires the determination of a minimum-cost Hamiltonian cycle in a directed (asymmetric TSP) or undirected (symmetric TSP) graph. Despite the simplicity of the problem statement, the TSP has been extensively studied form the 18th century for its practical relevance and computational complexity. A description of the TSP, along with overviews of solution methods for the problem, can be found, for instance, in Gutin and Punnen [129], Laporte [167] and Lawler et al. [169].

In the remainder of the section, we review the contributions in literature for solving TSPs in which the nodes of the graph are moving during the planning horizon. Such class of TSP variants is referred to as Kinetic Variant of TSP, Moving-Target TSP or Non-Stationary TSP. Hammar and Nilsson [130] consider a Kinetic TSP (KTSP) in the Euclidean plane in which each node travels a line with a constant speed. The authors prove the existence of a Polynomial Time Approximation Scheme (PTAS) for the problem called translational TSP in which the targets are sharing same direction and speed intensity. Under restrictions on the number of different speeds and their maximum value, the PTAS can be adapted to the general KTSP. While the static version of Euclidean TSP admits PTAS (Arora [11]), the Kinetic TSP cannot be approximated better than by a factor of $2^{\Omega(\sqrt{n})}$ by a polynomial time algorithm, unless $P = NP$.

Helvig et al. [134] study a set of variants of Moving-Target TSP, assuming a fixed speed for the target movement. An exact polynomial algorithm based on dynamic programming is presented for the case in which the pursuer vehicle and the targets are restricted on the same line. An approximate algorithm for the case in which most of the targets are stationary is also given. In addition, the authors address the moving-targets TSP with resupply, in which the vehicle is required to head back to the depot after intercepting a single target: under the restriction of targets moving along lines passing from the depot, exact and approximate results are presented. The resupply requirement is also considered in the multi-vehicle version of the problem for two particular problem instances.

Asahiro et al. [14] analyze the situation in which the interceptor vehicles (robots, in their case) move on straight track-lines and targets travel at fixed speed. Polynomial time algorithms and proof of NP-hardness are given for problem variants in which either the number of intercepted targets or the number of interceptor vehicles are variables to be optimized. Choubey [62] considers the situation in which the moving targets are traveling at constant speed in given direction. A simple genetic algorithm is proposed and compared with a greedy algorithm based on the ratios between the proximity to the origin with the speed intensity of each target. Limited computational testing is made on a random dataset in which the maximum number of targets considered is 10. Jiang et al. [148] consider the problem in which the targets are moving with constant speed in straight lines in a two-dimensional space. A genetic algorithm is proposed with two possible crossover mechanisms: order crossover and cycle crossover. Limited tests are conducted on a test-bed of 30 randomly generated instances with 10 nodes.

Finally, Stieber et al. [231] address the TSP with multiple vehicles and moving targets in a general framework, i.e., no restrictions on the targets trajectory, target speed and dimension of the space are imposed. The authors propose a mathematical formulation for the problem in a time-extended graph. A time-discretization step is needed for considering the problem as a MILP, instead of a nonlinear model. For producing good-quality solutions in a reasonable computing time, a heuristic procedure is developed to tackle instances with up to 36 target points and 3 vehicles.

In this chapter we address the routing problem described in Section 3.3, which shares some features with the problem variants present in the aforementioned papers. The common consideration for this class of problem is that the targets are moving in the space according to a known motion. Our intention is to classify this set of problems as particular cases of IVRPs.

3.3 IVRP Formulation: General Case

Let n be the number of targets to be picked up by a homogeneous fleet of K interceptor vehicles with capacity Q and vehicle speed V. At time $t = 0$, each target j has a given location, $q_j \in \mathbb{R}^2$, while all interceptor vehicles are located at depot O. In order to reach the meeting point with an interceptor vehicle, each target is allowed to move at speed v_j. The aim of the Interceptor Vehicle Routing Problem is to determine at most K minimum-time vehicle routes starting from the depot O and ending in the drop-off location D such that all targets are picked up by a vehicle in a convenient location (*meeting point*).

In order to develop a mathematical model, we introduce the following decision variables: the assignment of targets to vehicles is expressed by binary variables x_{ij}^k, where $i = 1, \dots, Q,\ j = 1, \dots, n+1 : (i, j) \neq (1, n+1),\ k = 1, \dots, K$, where

$$x_{ij}^k = \begin{cases} 1 & \text{if target } j \text{ is the } i\text{-th target visited by vehicle } k \\ 0 & \text{otherwise.} \end{cases}$$

Target $n + 1$ is an artificial static target located in D. The variables xs are sequential and not arc-based variables, hence no sub-tour elimination constraints are needed. The usage of vehicle k is represented by binary variable y_k where $k = 1, \dots, K$. The continuous component of the problem amounts to the determination of meeting points between targets and vehicles, of possible waiting times and of times required for reaching meeting points. The meeting points variables are divided into two groups:

- the sequential variables M_i^k, where $i = 0, \dots, Q + 1,\ k = 1, \dots, K$ express the coordinates in $\mathbb{R}^2$ of the i-th meeting point of vehicle k with a target; this means that at location M_i^k, vehicle k intercepts the i-th target.
- the second set of meeting point variables is given by the target-referred variables m_j, where $j = 1, \dots, n + 1$, which indicate the coordinates in $\mathbb{R}^2$ of the pick-up point of target j. Such non-sequential variables are introduced for dealing with the specific speed and trajectory of each target.

The meeting point variables with $i = 0$ coincide with O, while the variables with $i = Q + 1$ can take value D or O, depending on the vehicle usage. The waiting time for vehicle k between M^k_{i-1} and M^k_i is represented by $W^k_i, i = 1, \dots, Q,\ k = 1, \dots, K$, while variables $w_j, j = 1, \dots, n$ express the waiting time of target j before reaching m_j. Variables $T^k_i, i = 1, \dots, Q+1,\ k = 1, \dots, K$ compute the time required by vehicle k to travel from M^k_{i-1} to M^k_i, while $t_j, j = 1, \dots, n$ represent the time required by target j to reach m_j.

A formulation for the problem is given in model (3.1)-(3.21).

$$
\begin{aligned}
\min \quad & \sum_{k=1}^{K}\sum_{i=1}^{Q+1} T^k_i && && (3.1)\\
\text{s.t} \quad & \frac{\|M^k_i - M^k_{i-1}\|}{V} + W^k_i \le T^k_i && \forall\, i = 1, \dots, Q+1, \forall\, k = 1, \dots, K && (3.2)\\
& \frac{\|m_j - q_j\|}{v_j} + w_j \le t_j && \forall\, j = 1, \dots, n && (3.3)\\
& M^k_i \ge m_j - C_M(1 - x^k_{ij}) && \forall\, i = 1, \dots, Q, \forall\, k = 1, \dots, K, && \\
& && \forall j = 1, \dots, n+1 && (3.4)\\
& M^k_i \le m_j + C_M(1 - x^k_{ij}) && \forall\, i = 1, \dots, Q, \forall\, k = 1, \dots, K, && \\
& && \forall j = 1, \dots, n+1 && (3.5)\\
& M^k_{Q+1} = y_k \cdot D + (1 - y_k) \cdot O && \forall\, k = 1, \dots, K && (3.6)\\
& \sum_{i'=1}^{i} T^k_{i'} \ge t_j - C^T_j(1 - x^k_{ij}) && \forall\, i = 1, \dots, Q, \forall\, k = 1, \dots, K, && \\
& && \forall j = 1, \dots, n && (3.7)\\
& \sum_{k=1}^{K}\sum_{i=1}^{Q} x^k_{i,j} = 1 && \forall\, j = 1, \dots, n && (3.8)\\
& \sum_{j'=1}^{n} x^k_{i,j'} \ge x^k_{i+1,j} && \forall\, j = 1, \dots, n, \forall\, i = 1, \dots, Q-1, && \\
& && \forall\, k = 1, \dots, K && (3.9)\\
& y_k = \sum_{j=1}^{n} x^k_{1,j} && \forall\, k = 1, \dots, K && (3.10)\\
& \sum_{j=1}^{n+1} x^k_{i,j} = y_k && \forall\, i = 2, \dots, Q, \forall\, k = 1, \dots, K && (3.11)\\
& t_j \ge 0 && \forall j = 1, \dots, n && (3.12)\\
& w_j \ge 0 && \forall j = 1, \dots, n && (3.13)\\
& T^k_i \ge 0 && \forall\, i = 1, \dots, Q+1, \forall\, k = 1, \dots, K && (3.14)\\
& W^k_i \ge 0 && \forall\, i = 1, \dots, Q+1, \forall\, k = 1, \dots, K && (3.15)\\
& M^k_i \in [X_{min}, X_{max}] \times [Y_{min}, Y_{max}] && \forall\, i = 1, \dots, Q+1, \forall\, k = 1, \dots, K && (3.16)\\
& m_j \in [X_{min}, X_{max}] \times [Y_{min}, Y_{max}] && \forall\, j = 1, \dots, n && (3.17)\\
& x^k_{i,j} \in \{0, 1\} && \forall\, i = 1, \dots, Q, \forall\, j = 1, \dots, n+1, &&
\end{aligned}
$$

$$\forall\ k = 1, \dots, K, (i, j) \neq (1, n+1) \quad (3.18)$$

$$y_k \in \{0, 1\} \qquad \forall\ k = 1, \dots, K \quad (3.19)$$

$$M_0^k = O \qquad \forall\ k = 1, \dots, K \quad (3.20)$$

$$m_{n+1} = D. \quad (3.21)$$

The objective function (3.1) consists of the travel time of the vehicles for reaching the drop-off location D. Constraints (3.2) ensure that the time interval in which the vehicle travels between two consecutive meeting points is the time required for following the segment between the two locations, plus an eventual waiting time within such segment. Distances are computed via the Euclidean norm $\|\cdot\|$ ($\|\cdot\| := \|\cdot\|_2$). The inequality sign is needed for maintaining the convexity of the feasible region. It is worth to note that the real waiting time is the difference between the right-hand side and the fraction in the left-hand side. Waiting time variables W_i^k have been explicitly introduced for the readability of the formulation and eventually to introduce further limitations on the waiting times. Constraints (3.3) define the time taken by the target to be picked up by a vehicle similar to constraints (3.2). Regarding the inequality sign and the waiting times, same considerations made for (3.3) hold. Constraints (3.4) and (3.5) are "big-M" constraints for expressing the logic conditions:

$$M_i^k = m_j \ \text{ if } x_{i,j}^k = 1 \quad \forall\ i = 1, \dots, Q, \forall\ k = 1, \dots, K, \forall\ j = 1, \dots, n+1.$$

Such equalities are needed for ensuring the compatibility between the sequential meeting points M_i^k and m_j. The constant $C_M = (C_{M,x}, C_{M,y})$ in (3.4) and (3.5) can be safely defined as $C_{M,x} = X_{max} - X_{min}, C_{M,y} = Y_{max} - Y_{min}$, where $X_{max}, X_{min}, Y_{max}, Y_{min}$ are the limitation for the meeting point variables. As constraints (3.6) state, the last point reached by a vehicle is either the drop-off location or the original depot: this depends on whether the vehicle has been used or not. Constraints (3.7) translate the logical inequalities:

$$\sum_{i'=1}^{i} T_{i'}^k \geq t_j \quad \text{if } x_{i,j}^k = 1 \quad \forall\ i = 1, \dots, Q, \forall\ k = 1, \dots, K, \forall\ j = 1, \dots, n, \quad (3.22)$$

which impose the synchronization between vehicle and target meeting point. The equality sign is not necessary since the vehicle-related time variables are minimized. Let δ be the length of the diagonal of the spatial limitation box, and $\bar{w}_j$ be the maximum allowable waiting time for target j, then the constants C_j^T can be set as $C_j^T = \frac{\delta}{v_j} + \bar{w}_j$. The requirement that each target is picked up by a vehicle is expressed in (3.8). Furthermore, such equalities explicitly forbid multiple pick-ups of a target. Constraints (3.9) impose the sequentiality in the visiting order in each vehicle. They are a stronger version of the valid constraints:

$$\sum_{j=1}^{n} x_{i,j}^{k} \geq \sum_{j=1}^{n} x_{i+1,j}^{k} \quad \forall\ i = 1, \ldots, Q-1, \forall\ k = 1, \ldots, K. \tag{3.23}$$

Constraints (3.10) define the y variables, which are merely introduced for the readability of the model. Constraints (3.11) are necessary to express that a vehicle that intercepted at least one target can either pick up an additional target or head towards the drop-off location D. As a consequence of constraints (3.9), D is a destination site for each used vehicle. Constraints (3.12)-(3.19) define the decision variables of the problem. Bounds (3.20) state that each vehicle is at the depot O at time $t = 0$: since they are vehicle dependent conditions, they can handle the case of several depots, along with constraints (3.6). Bounds (3.21) express the destination role of D in a vehicle route. As stated in constraints (3.18), variables $x_{1,n+1}^{k}$ are not needed in the model, since it is never profitable for a vehicle to leave the depot for going directly to the drop-off locations. Finally, it should be underlined that the sequential meaning of the x variables has the advantage of naturally containing the capacity restrictions on the fleet of vehicle. Indeed, for imposing that each vehicle cannot intercept more than Q targets, it is enough to avoid defining variables $x_{i,j}^{k}$ with $i > Q$.

The model (3.1)-(3.21) is a Mixed-Integer Second Order Conic problem (MISOCP). It consists of a linear objective function, linear constraints and second order conic constraints (3.2) and (3.3). In order to solve the model with an optimization solver, such as Cplex, it is generally required for second order conic constraint to be expressed in the quadratic version of the standard form

$$\|x\| \leq t \quad x \in \mathbb{R}^n, t \in \mathbb{R}.$$

Constraints (3.2) and (3.3) can be rewritten in standard form by adding artificial variables subject to linear equalities. For instance, in order to obtain the standard form of constraints (3.2), additional variables $\mathcal{M}_i^k$ and $\mathcal{T}_i^k$ are added subject to the following conditions:

$$\mathcal{M}_i^k = M_i^k - M_{i-1}^k \quad \forall\ i = 1, \ldots, Q+1, \forall\ k = 1, \ldots, K$$
$$\mathcal{T}_i^k = T_i^k - W_i^k \quad \forall\ i = 1, \ldots, Q+1, \forall\ k = 1, \ldots, K.$$

3.4 IVRP Formulation: Targets Moving Along a Fixed Line

The main focus of the work is the special case of Interceptor Vehicle Routing Problem in which each target point j is allowed to move from initial site q_j along a predefined

line. The lines are expressed by vectors $d_j,\ j = 1, \dots, n$, which indicate the direction of the targets movement.
A mathematical model for this special variant is obtained from the IVRP formulation (3.1)-(3.21) by adding constraints for expressing the limitations in the targets trajectories. Scalar variables $\lambda_j,\ j = 1, \dots, n$ are used to parametrize the line passing from q_j with direction vector d_j. The limitations in determining the meeting points are imposed in the following conditions

$$m_j - q_j = \lambda_j\ d_j \quad \forall\ j = 1, \dots, n. \tag{3.24}$$

An important observation is that when constraints (3.24) are imposed, then constraints (3.3) reduce to the following linear conditions

$$\frac{\lambda_j \|d_j\|}{v_j} + w_j = t_j \quad \forall\ j = 1, \dots, n. \tag{3.25}$$

Taking advantage of the linearity of (3.25), the equality sign can be imposed without losing the convexity of the feasibility region. Variables w_j represent the real waiting time of target j. To sum up, the model for the IVRP with target points moving along a fixed line is given by:

$$\begin{aligned}
\min \quad & (3.1) && (3.26)\\
\text{s.t} \quad & (3.2), (3.4) - (3.21), && (3.27)\\
& (3.24), (3.25), && (3.28)\\
& \lambda_j \geq 0 \qquad \forall\ j = 1, \dots, n. && (3.29)
\end{aligned}$$

The proposed formulation (3.26) - (3.29) is a MISOCP as in the general case. However, the model for this special case has n conic constraints less.

3.4.1 Valid Inequalities

In this section, we propose a set of valid inequalities for the Interceptor Vehicle Routing Problem. They are valid cuts for the general formulation (3.1)-(3.21) and are used to strengthen the continuous relaxation and consequently speed up the solution algorithm of CPLEX. The inequalities are defined in CPLEX as user cuts. The solver is free to check the possible violation of the cut at any stage of the optimization; therefore, if the inequalities are not inferred by the original matrix constraint, then there are no guarantees that the optimal solution given by CPLEX is feasible for such cuts.

The first family of cuts aims to tighten the requirement that the drop-off location D is the destination of each vehicle trajectory. The condition can be directly expressed on column $n+1$ of variable matrix x^k in one of the two ways:

$$x^k_{i,n+1} \le x^k_{i+1,n+1} \qquad \forall\ i = 2,\dots,Q-1, \forall\ k = 1,\dots K, \tag{3.30}$$

$$(Q-i)\ x^k_{i,n+1} \le \sum_{i'=i+1}^{Q} x^k_{i',n+1} \quad \forall\ i = 2,\dots,Q-1, \forall\ k = 1,\dots K, \tag{3.31}$$

or involving also the remaining columns of matrix x^k:

$$\sum_{i'=i+1}^{Q} \sum_{j=1}^{n} x^k_{i',j} + (Q-i)x^k_{i,n+1} \le (Q-i)y_k \quad \forall\ i = 2,\dots,Q-1, \forall\ k. \tag{3.32}$$

Since the fleet of vehicles is homogeneous, symmetry breaking cuts can be considered. We declare the following simple constraints in Cplex as standard constraints

$$y_1 = 1 \tag{3.33}$$

$$y_k \ge y_{k+1} \quad \forall\ k = 1,\dots,K-1, \tag{3.34}$$

so that for any feasible solution met by the solver conditions (3.33) and (3.34) hold. In addition to these constraints, we propose families of cuts inspired by the rules described in Fischetti et al. [93] and applied, for instance, in Coelho and Laporte [65]. Such symmetry breaking constraints arise from the following consideration. Given a feasible solution of IVRP, it is always possible to construct an equivalent solution in which each vehicle k is allowed to pick up target j only if vehicle $k-1$ picks up a target with index smaller than j. Constraints (3.35)

$$\sum_{i=1}^{Q-1} x^k_{i,j} \le \sum_{i'=1}^{Q-1} \sum_{j'<j} x^{k-1}_{i',j'} \quad \forall\ k = 2,\dots,K, \forall\ j = 2,\dots,n \tag{3.35}$$

express the symmetry breaking rule.

We also present cuts (3.36) and (3.37) that arise from the consideration of the number L_k of targets not served by any vehicle with index not greater than k.

$$n - \Big(Q - \sum_{i=2}^{Q} x^1_{i,n+1}\Big) - \sum_{k'=2}^{K}\Big(Qy_{k'} - \sum_{i=2}^{Q} x^{k'}_{i,n+1}\Big) \le (n-k)y_{k+1} \quad \forall\ k = 1,\dots,K-1, \tag{3.36}$$

$$n - \Big(Q - \sum_{i=2}^{Q} x^1_{i,n+1}\Big) - \sum_{k'=2}^{K}\Big(Qy_{k'} - \sum_{i=2}^{Q} x^{k'}_{i,n+1}\Big) \ge y_{k+1} \qquad \forall\ k = 1,\dots,K-1. \tag{3.37}$$

Proof of validity of (3.36) and (3.37).

The quantity L_k can be computed in the following way:

$$L_k = \underbrace{n - \left(Q - \sum_{i=2}^{Q} x^1_{i,n+1}\right)}_{\text{targets not served by vehicle 1}} - \sum_{k'=2}^{K} \underbrace{\left(Qy_{k'} - \sum_{i=2}^{Q} x^{k'}_{i,n+1}\right)}_{\text{targets served by vehicle } k'} \quad \forall\ k = 1, \ldots, K-1.$$

For feasible solutions of IVRP which respect the valid cuts (3.33) and (3.34), L_k can vary from $n-k$ to 0 (for vehicle $\bar{k} = \max_{k=1,\ldots,K}\{y_k = 1\}$). If L_k is strictly greater than 0, then k vehicles are not sufficient for serving all targets and therefore vehicle $k+1$ has to be used: this is attained by imposing $L_k \leq (n-k)\ y_{k+1}\ \forall\ k = 1, \ldots, K-1$ in (3.36). Otherwise $L_k = 0$ and all targets are picked up by the first k vehicles and hence vehicle $k+1$ is not needed: this is expressed by $L_k \geq y_{k+1}\ \forall\ k = 1, \ldots, K-1$ in (3.37).

The impact of the proposed valid inequalities is evaluated via the computational tests described in Section 3.6

3.5 Lagrangian Decomposition

Lagrangian Decomposition methods are well-known exact algorithms for solving complex optimization problems (see Section 1.3.4 and, for instance, Frangioni [97], Fisher [94], and the recent applications described in Ghaddar et al. [110], Hosni et al. [140]). Such methods are applied when the formulation exhibits a set of "complicating" constraints, while the remaining constraints form a substantially easier problem. In Vehicle Routing Problems, a common approach is to relax the assignment constraints of targets to vehicles in a Lagrangian fashion (see, e. g., Kohl and Madsen [164]). This section explains the Lagrangian Decomposition approach that we propose for tackling the IVRP with targets moving on fixed lines. We apply Lagrangian relaxation to constraints (3.8) with multipliers μ_j to obtain the following subproblem:

$$v_{SP(\mu)} = \min \sum_{k=1}^{K}\sum_{i=1}^{Q+1} T_i^k - \sum_{k=1}^{K}\sum_{i=1}^{Q}\sum_{j=1}^{n} \mu_j x^k_{i,j} + \left[\sum_{j=1}^{n} \mu_j\right] \tag{SP}$$
$$\text{s.t. } (3.2), (3.4) - (3.7), (3.9) - (3.21), (3.24), (3.25), (3.29).$$

In problem SP, variables in set $\mathcal{V} = \{m_j, \lambda_j, t_j \text{ and } w_j : j = 1, \ldots, n\}$ are the only vehicle-independent variables. In order to isolate single-vehicle subproblems, in principle, an artificial dependency on the vehicle should be added for each $v \in \mathcal{V}$ along with equality constraints

$$m_j^k = m_j^{k'}, \lambda_j^k = \lambda_j^{k'}, t_j^k = t_j^{k'}, w_j^k = w_j^{k'} \quad \forall\ j = 1, \ldots, n,\ \ k, k' = 1, \ldots, K. \tag{3.38}$$

In such a setting, the problem is not directly decomposable in single-vehicle problems. Calling $\overline{\mathcal{V}}$ the set obtained from $\mathcal{V}$ after adding the vehicle dependency, it should be observed that a variable $v_j^k \in \overline{\mathcal{V}}$ has an impact in the remainder of the model (that is, in constraints (3.4),(3.5) and (3.7)) only if an index i for which $x_{i,j}^k = 1$ exists. However, since no targets can be picked up by two different vehicles, constraints (3.38) are not necessary. These considerations prove that SP decomposes into K identical single-vehicle problems:

$$\min \quad \sum_{i=1}^{Q+1} T_i - \sum_{j=1}^{n}\sum_{i=1}^{Q} x_{i,j}\mu_j = v(SSP(\mu)) \tag{3.39}$$

$$\text{s.t} \quad \frac{\|M_i - M_{i-1}\|}{V} + W_i \le T_i \qquad \forall\, i = 1,\dots,Q+1 \tag{3.40}$$

$$\frac{\|d_j\|\lambda_j}{v_j} + w_j = t_j \qquad \forall\, j = 1,\dots,n \tag{3.41}$$

$$m_j - q_j = \lambda_j\, d_j \qquad \forall\, j = 1,\dots,n \tag{3.42}$$

$$M_i \ge m_j - C_M(1 - x_{ij}) \qquad \forall\, i = 1,\dots,Q, \forall\, j = 1,\dots,n+1 \tag{3.43}$$

$$M_i \le m_j + C_M(1 - x_{ij}) \qquad \forall\, i = 1,\dots,Q, \forall\, j = 1,\dots,n+1 \tag{3.44}$$

$$M_{Q+1} = y \cdot D + (1 - y) \cdot O \tag{3.45}$$

$$\sum_{i'=1}^{i} T_{i'} \ge t_j - C_T(1 - x_{ij}) \qquad \forall\, i = 1,\dots,Q, \forall\, j = 1,\dots,n \tag{3.46}$$

$$\sum_{j'=1}^{n} x_{i,j'} \ge x_{i+1,j} \qquad \forall\, j = 1,\dots,n, \forall\, i = 1,\dots,Q-1 \tag{3.47}$$

$$y = \sum_{j=1}^{n} x_{1,j} \tag{3.48}$$

$$\sum_{j=1}^{n+1} x_{i,j} = y \qquad \forall\, i = 2,\dots,Q \tag{3.49}$$

$$t_j \ge 0 \qquad \forall\, j = 1,\dots,n \tag{3.50}$$

$$w_j \ge 0 \qquad \forall\, j = 1,\dots,n \tag{3.51}$$

$$\lambda_j \ge 0 \qquad \forall\, j = 1,\dots,n \tag{3.52}$$

$$T_i \ge 0 \qquad \forall\, i = 1,\dots,Q+1 \tag{3.53}$$

$$W_i \ge 0 \qquad \forall\, i = 1,\dots,Q+1 \tag{3.54}$$

$$M_i \in [X_{min}, X_{max}] \times [Y_{min}, Y_{max}] \qquad \forall\, i = 1,\dots,Q+1 \tag{3.55}$$

$$m_j \in [X_{min}, X_{max}] \times [Y_{min}, Y_{max}] \qquad \forall\, j = 1,\dots,n \tag{3.56}$$

$$x_{i,j} \in \{0,1\} \qquad \forall\, i = 1,\dots,Q, \forall\, j = 1,\dots,n+1, \tag{3.57}$$

$$(i,j) \ne (1, n+1) \tag{3.58}$$

$$y \in \{0,1\} \tag{3.59}$$

$$M_0 = O \tag{3.60}$$

$$m_{n+1} = D. \tag{3.61}$$

The value $v(LR(\mu))$ of the Lagrangian relaxation of problem (3.26)-(3.29) is therefore computed as:

$$v(LR(\mu)) = K \cdot v(SSP(\mu)) + \sum_{j=1}^{n} \mu_j. \tag{3.62}$$

The Lagrangian Dual problem amounts to find $v(LR) = \max_{\mu \in \mathbb{R}^n} v(LR(\mu))$. In order to determine such bound, we develop the following iterative procedure. Let H be the set of feasible solutions of (3.39)-(3.61), the Lagrangian Dual problem can be written as:

$$\max_{\mu} \Big\{ K \min_{h=1,\dots,H} \Big\{ \sum_{i=1}^{Q+1} T_i^{(h)} - \sum_{j=1}^{n} \sum_{i=1}^{Q} \mu_j x_{i,j}^{(h)} \Big\} + \sum_{j=1}^{n} \mu_j \Big\},$$

which is equivalent to the Lagrangian Master Problem (LMP):

$$\max \sum_{j=1}^{n} \mu_j + K\theta \tag{3.63}$$

$s.t$

$$\theta \leq \sum_{i=1}^{Q+1} T_i^{(h)} - \sum_{j=1}^{n} \sum_{i=1}^{Q} x_{i,j}^{(h)} \mu_j \quad \forall\, h \in H \tag{3.64}$$

$$\theta \text{ free} \tag{3.65}$$

$$m_j \text{ free} \qquad \forall\, j = 1, \dots, n. \tag{3.66}$$

Denoting with α_h the dual variables associated with the constraint set (3.64), the dual of LMP is the Dantzig-Wolfe master problem (DLMP):

$$\min \sum_{h \in H} \Big(\sum_{i=1}^{Q+1} T_i^{(h)} \Big) \alpha_h \tag{3.67}$$

$s.t$

$$\sum_{h \in H} \alpha_h = K \tag{3.68}$$

$$\sum_{h \in H} \Big(\sum_{i=1}^{Q} x_{i,j}^{(h)} \Big) \alpha_h = 1 \quad \forall\, j = 1, \dots, n \tag{3.69}$$

$$\alpha_h \geq 0 \qquad \forall\, h \in H. \tag{3.70}$$

Each cut in LMP is associated to a column in DLMP. Since the set H is not known beforehand, in order to calculate $v(LR)$, an iterative procedure starting from a relaxation of LMP is developed. In each iteration, subproblem $SSP(\mu)$ is solved for a value of μ: being a feasible solution of the Lagrangian Dual, the optimal solution of $SSP(\mu)$ is a lower bound on $v(LR)$. Optimal solution values $\bar{T}_i$ and $\bar{x}_{i,j}$ correspond to a new cut in the Relaxed LMP (RLMP). Being a relaxation, the optimal value of each RLMP is an upper bound on $v(LR)$. The algorithm stops when a desired tolerance ϵ on the relative gap between lower and upper bound is reached. The relative gap is computed as $\frac{UB-LB}{UB}$ and the ϵ is set to 10^{-3} in our computation. An imposition of a large bound on the Lagrange multipliers μ is needed in the first iterations to overcome the fact that the initial RLMPs are unbounded.

3.5.1 Tightening the Lagrangian Bound

Since $SSP(\mu)$ is an MISOCP, the computational core of the iterative procedure lies in solving the subproblems. Early computations of such Lagrangian bound were proving it to be quite a weak bound: in some cases the bound was 50% below the optimal solution. The computational times where instead promising: the order of magnitude is 10^0 for instances with up to 12 target points and 3 vehicles. We decided to tighten the bound. A primary reason for the weakness of the initial bound is that the relaxation ignores not only the assignment of each targets to a single vehicle, but also the requirement of picking up a target in a unique visiting order. This also implies that in a non-null optimal solution of $SSP(\mu)$, it will always be convenient to pick up n targets in a vehicle, without the guarantee that such targets are distinct. We improved the Lagrangian bound by adding in subproblem $SSP(\mu)$ the following inequalities that forbid multiple visits of a targets:

$$\sum_{i=1}^{Q} x_{i,j} \leq 1 \quad \forall\, j = 1, \ldots, n. \tag{3.71}$$

The tightened Lagrangian bound is used in a branch-and-price algorithm. The algorithm starts with an initial incumbent obtained by assigning targets to vehicles according to the order given by their number label respecting the vehicle capacity. Each node is solved with the cutting plane procedure described in Section 3.5. If the solution of a node is feasible for the relaxed constraints, then a new upper bound is found. Otherwise, two child nodes are created.

The branching rule is established according to the dual variables α_h values. If a node yields a solution with integer α values, then it is a feasible solution for the original IVRP problem, otherwise branching constraints are imposed according to the fractional αs. The exploration of a node can be interrupted if the subproblem finds a solution of value higher than the incumbent: this can either mean that the node has a worse Lagrangian bound than the incumbent or that the node is infeasible since it violates some of the branching constraints.

3.5.2 Branching Strategy

In the solutions of the master problem with fractional values of α, some targets are partially assigned to more than one vehicle. We adopted the same branching rule applied in Elhedhli et al. [87] for a Bin Packing problem; the strategy was originally proposed in Ryan and Foster [218]. The imposition of two targets in a single vehicle routes versus in two different vehicles can be formulated without losing the structure of the subproblems. In other words, in each node, the branching constraints will be vehicle dependent and therefore distributed in the single-vehicle subproblems. The branching constraints are then:

$$\sum_{i=1}^{Q} x_{i,j_1}^k = \sum_{i=1}^{Q} x_{i,j_2}^k \quad \forall\, k = 1, \ldots, K, \tag{3.72}$$

$$\sum_{i=1}^{Q} x_{i,j_1}^k + \sum_{i=1}^{Q} x_{i,j_2}^k \leq 1 \quad \forall\, k = 1, \ldots, K. \tag{3.73}$$

Constraints (3.72) impose that if targets j_1 and j_2 are picked up by a same vehicle, then they share the same vehicle. Instead, constraints (3.73) forbid the two targets to be picked up by the same vehicle.
The determination of j_1 and j_2 is performed in the following way. At each fractional node, the matrix M of cuts (3.69) is explored until two rows j_1 and j_2 and two columns h_1 and h_2 with fractional α_1 and α_2 exhibit the scheme:

	h_1	$\cdots$	h_2	$\cdots$
j_1	1	$\cdots$	1	$\cdots$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
j_2	1	$\cdots$	0	$\cdots$

The introduction of constraints (3.71) guarantees the existence of the described branching pattern in each fractional node of the branch and bound tree. The α_hs with higher values and the targets with higher index are checked first. When creating a child node, the cuts in the father node master problem that satisfy the new branching constraints are used for warm-starting the relaxation of the child node master problem.

3.6 Implementation Details and Preliminary Computational Results

The B&P algorithm has been coded in C. The implementation comprehends several routines: generating a simple feasible route and computing an initial incumbent; solving a node with the cutting plane procedure; detecting branching patterns for fractional nodes and generating child nodes. Each optimization problem has been solved by Cplex

12.6 ([70]). The Cplex defaults setting were maintained. The tree was explored with a depth-first search strategy where the left nodes are associated to constraints (3.72). Regarding the formulation, constraints (3.4),(3.5),(3.7), (3.9) were declared as lazy constraints. They constitute a relatively large set of constraints and it is likely that the majority of them are not binding in the optimal solution. Early computational tests showed a decrease in the CPU time required by Cplex, after declaring the lazy constraints.

The section presents some computational results of both Cplex and B&P for the IVRP only in the special case in which targets are constrained to move along fixed lines. All runs were performed on a QEMU Virtual CPU version 0.14.1 @ 2.40 GHz (Cluster). One core in an isolated node was used. A time limit of 7200 seconds on the solution methods was imposed.

A set of 18 test instances was randomly generated. The number n of targets varied from 10 to 20, while the maximum number K of available vehicles ranged from 3 to 5. Vehicle capacity Q was set as $\lceil \frac{n}{K} \rceil + 2$. The target initial locations was randomly chosen in the portion of the Euclidean plane centred in $(0, 0)$, with width 50 and height 100. Vehicle depot is located in $(-20, 0)$, whereas drop-off location is the point $(20, 0)$. The vehicle speed intensity was randomly chosen between 2 to 3, while targets can move in the positive direction of the arbitrary line d_j with a speed sampled from the $[0, 1]$ interval.

Table 3.1 compares Cplex on the MISOCP formulation of Section 3.4 and the B & P algorithm of Section 3.5. For both methods, the following solution information are reported: the best solution value found (column *BestUB*), which is marked with an asterisk if it is proven to be optimal; the percentage gap between BestUB and the best known lower bound (column *Gap*); elapsed computational time in seconds (column *Time*). Regarding the percentage gap, the B & P lower bound is computed in the tree exploration as the smallest of the Lagrangian Relaxation bounds of the fathers of all open (i.e., not solved or fathomed) nodes. The CPLEX solver provides lower bounds by the best problem relaxation built.

The Cplex performance on the full formulation is deeply related to the size of the instance, regarding both the number of target points and vehicles available. The B & P algorithm is instead able to solve to optimality the 14 and 16-target points instances and some of the 18 and 20-target points instances. In the large majority of the cases, B & P beats CPLEX in terms of gap and time. The computational times required by B & P show a decreasing trend when the number of interceptor vehicles increases: this highlights the relevance of the Lagrangian Decomposition approach within the algorithm.

Table 3.2 displays additional statistics on the B & P runs. The number of solved nodes is reported in column *#Node*. The number of iterations required to solve the Lagrangian Relaxation, the average time spent in solving master problems and the average time spent in solving subproblems are reported at the root node (respectively, in columns *IterRoot*, *TimeMasterRoot* and *TimeSubRoot*) and at the children nodes with averaged values (respectively, in columns *IterChild*, *TimeMasterChild* and *TimeSubChild*). It can be observed that in some small-medium sized instances, the Lagrangian Bound at the root node is the IVRP optimal value. Note also that the root node can be very difficult to solve, especially when the number of targets increases and few vehicles are available (e.g., instances 18_3 and 20_3). The children nodes are less time-consuming thanks to the warm-starting techniques mentioned in Section 3.5.2. The computational bottleneck of the B & P procedure is in solving the MISOCP subproblems.

Table 3.3 reports gap and time measures of Cplex with the introduction of constraints (3.30) (called *usercuts1*), constraints (3.31) (*usercuts2*) and constraints (3.32) (*usercuts3*) separately upon the standard formulation (3.26) - (3.29) (*nousercuts*). The addition of such valid inequalities as user cuts for CPLEX generally speeds up the algorithm; the impact of the time saving seems less evident when the number of vehicle increases. Only *usercuts1* permits to solve an additional instance (i.e., 14_4) within the time limit with respect to *nousercuts*. The smallest average gap at the end of the CPLEX resolution is reported when using *usercuts3*, while the lowest average computational time is required by *usercuts1*.

Gaps and solution times are also displayed for evaluating the impact of the symmetry breaking constraints described in Section 3.4.1. In Table 3.4, the standard formulation *nosymm* (*nosymm=nousercuts*) is compared with the MISOCPs obtained by adding respectively: (3.33),(3.34) in *symm1*; (3.33),(3.34), (3.35) in *symm2*; (3.33),(3.34), (3.36), (3.37) in *symm3*. Introducing one of such symmetry breaking formulations allows to solve instance 14_5 to optimality within the time limit; *symm2* also solves instance 14_4. The average measures of precision and time indicate the superiority of the proposed symmetry breaking constraints over the valid inequalities *usercuts1, usercuts2, usercuts3*, even for the bigger-size instances.

		Cplex			B&P		
Instance n_K	Q	BestUB	Gap	Time	BestUB	Gap	Time
10_3	6	85.51*	0.01%	154.44	85.51*	0.00%	30.36
10_4	5	72.06*	0.01%	125.23	72.06*	0.00%	19.55
10_5	4	104.02*	0.01%	218.13	104.02*	0.00%	20.13
12_3	6	94.55*	0.01%	418.23	94.55*	0.00%	57.27
12_4	5	109.34*	0.01%	2925.22	109.34*	0.00%	138.75
12_5	5	136.26*	0.01%	3273.76	136.26*	0.00%	126.85
14_3	7	101.44*	0.01%	1633.92	101.43*	0.00%	294.57
14_4	6	107.88	9.59%	7200.00	105.19*	0.00%	615.52
14_5	5	89.31	9.10%	7200.00	89.31*	0.00%	106.85
16_3	8	100.63	11.05%	7200.00	100.63*	0.00%	3742.51
16_4	6	101.51	26.25%	7200.00	97.91*	0.00%	1511.33
16_5	6	105.37	13.44%	7200.00	105.37*	0.00%	483.68
18_3	8	162.01	24.47%	7200.00	167.57	15.60%	7555.78
18_4	7	86.57	16.53%	7200.00	85.96	3.86%	7472.99
18_5	6	99.33	24.66%	7200.00	97.94*	0.00%	1456.12
20_3	9	116.52	36.46%	7200.00	248.08	100.00%	56275.24
20_4	7	144.81	40.82%	7200.00	128.94*	0.00%	4984.58
20_5	6	139.86	29.27%	7200.00	137.36	7.23%	7248.04

TABLE 3.1: Cplex and Branch & Price comparison

Instance n_K	Q	#Node	IterRoot	TimeMasterRoot	TimeSubRoot	IterChild	TimeMasterChild	TimeSubChild
10_3	6	1	31	0.00	30.34	-	-	-
10_4	5	1	29	0.00	19.54	-	-	-
10_5	4	6	28	0.00	10.29	4	0.000478	0.831206
12_3	6	1	37	0.00	57.25	-	-	-
12_4	5	23	35	0.00	39.45	4	0.00049	2.766247
12_5	5	17	37	0.00	38.24	4	0.000596	2.458145
14_3	7	1	40	0.00	294.53	-	-	-
14_4	6	17	49	0.01	202.38	7	0.008095	18.559202
14_5	5	4	42	0.00	53.88	5	0.000727	6.293636
16_3	8	1	76	0.01	3742.34	-	-	-
16_4	6	22	57	0.01	284.80	8	0.001296	31.728302
16_5	6	3	72	0.01	393.78	10	0.001645	44.908916
18_3	8	18	96	0.27	4951.35	5	0.000914	142.603456
18_4	7	27	73	0.01	1455.41	11	0.001809	160.641088
18_5	6	22	58	0.01	418.90	6	0.000935	29.370864
20_3	9	1	105	0.08	56274.44	-	-	-
20_4	7	5	91	0.02	2536.52	20	0.003702	464.60414
20_5	6	230	77	0.01	660.13	4	0.001144	14.509537

TABLE 3.2: Branch & Price statistics

		nousercuts		usercuts1		usercuts2		usercuts3	
Instance n_K	Q	Gap	Time	Gap	Time	Gap	Time	Gap	Time
10_3	6	0.01%	154.44	0.01%	89.71	0.01%	107.85	0.01%	123.32
10_4	5	0.01%	125.23	0.01%	127.69	0.01%	127.80	0.01%	127.62
10_5	4	0.01%	218.13	0.01%	218.16	0.01%	218.36	0.01%	218.57
12_3	6	0.01%	418.23	0.01%	417.65	0.01%	417.79	0.01%	417.14
12_4	5	0.01%	2925.22	0.01%	2487.05	0.01%	4766.57	0.01%	2545.83
12_5	5	0.01%	3273.76	0.01%	3199.12	0.01%	3190.47	0.01%	3200.27
14_3	7	0.01%	1633.92	0.01%	1280.54	0.01%	3126.53	0.01%	1992.16
14_4	6	9.59%	7200.00	0.01%	6571.49	35.01%	7200.00	3.75%	7200.00
14_5	5	9.1%	7200.00	9.84%	7200.00	9.85%	7200.00	5.96%	7200.00
16_3	8	11.05%	7200.00	10.95%	7200.00	12.79%	7200.00	8.33%	7200.00
16_4	6	26.25%	7200.00	17.92%	7200.00	24.99%	7200.00	14.94%	7200.00
16_5	6	13.44%	7200.00	10.37%	7200.00	10.19%	7200.00	12.22%	7200.00
18_3	8	24.47%	7200.00	33.93%	7200.00	32.01%	7200.00	40.93%	7200.00
18_4	7	16.53%	7200.00	24.26%	7200.00	13.06%	7200.00	12.12%	7200.00
18_5	6	24.66%	7200.00	24.13%	7200.00	24.45%	7200.00	24.13%	7200.00
20_3	9	36.46%	7200.00	47.54%	7200.00	31.54%	7200.00	32.04%	7200.00
20_4	7	40.82%	7200.00	37.94%	7200.00	45.26%	7200.00	37.58%	7200.00
20_5	6	29.27%	7200.00	30.17%	7200.00	26.36%	7200.00	34.8%	7200.00
Average values		13.43%	4886.05	13.73%	4799.52	14.75%	5064.19	12.60%	4879.16

TABLE 3.3: Valid inequalities

		nosymm		symm1		symm2		symm3	
Instance n_K	Q	Gap	Time	Gap	Time	Gap	Time	Gap	Time
10_3	6	0.01%	154.44	0.01%	74.58	0.01%	106.87	0.01%	74.21
10_4	5	0.01%	125.23	0.01%	60.85	0.01%	42.71	0.01%	60.96
10_5	4	0.01%	218.13	0.01%	65.73	0.01%	42.36	0.01%	32.25
12_3	6	0.01%	418.23	0.01%	86.22	0.01%	68.78	0.01%	86.30
12_4	5	0.01%	2925.22	0.01%	1365.97	0.01%	1216.23	0.01%	1371.93
12_5	5	0.01%	3273.76	0.01%	915.24	0.01%	4836.35	0.01%	876.23
14_3	7	0.01%	1633.92	0.01%	735.28	0.01%	737.42	0.01%	737.17
14_4	6	9.59%	7200.00	2.27%	7200.00	0.01%	3143.31	2.2%	7200.00
14_5	5	9.1%	7200.00	0.01%	3335.44	0.01%	3287.48	0.01%	2490.84
16_3	8	11.05%	7200.00	4.07%	7200.00	8%	7200.00	3.86%	7200.00
16_4	6	26.25%	7200.00	10.07%	7200.00	10.95%	7200.00	11.14%	7200.00
16_5	6	13.44%	7200.00	1.44%	7200.00	1.92%	7200.00	14.49%	7200.00
18_3	8	24.47%	7200.00	37.99%	7200.00	34.22%	7200.00	32%	7200.00
18_4	7	16.53%	7200.00	12.98%	7200.00	13.58%	7200.00	15.42%	7200.00
18_5	6	24.66%	7200.00	19.1%	7200.00	31.5%	7200.00	19.12%	7200.00
20_3	9	36.46%	7200.00	38.06%	7200.00	42.62%	7200.00	36.49%	7200.00
20_4	7	40.82%	7200.00	33.92%	7200.00	27.25%	7200.00	31.58%	7200.00
20_5	6	29.27%	7200.00	23.87%	7200.00	28.96%	7200.00	24.41%	7200.00
Average values		13.43%	4886.05	10.21%	4368.85	11.06%	4348.97	10.6%	4318.33

TABLE 3.4: Symmetry breaking constraints

3.7 Conclusions

In this chapter, we presented novel MISOCP formulations for a class of VRP variants with moving targets, which we unify under the name of IVRPs. We proposed a branch-and-price algorithm based on the Lagrangian Relaxation of the vehicle-assignment constraints. The structure of the obtained MISOCP relaxation is exploited by a Lagrangian Decomposition strategy, which makes the solution method computationally viable for test instances with at most 20 targets. Under preliminary testing on a special problem variant, the branch-and-price dominates the standard Cplex resolution both in terms of required time for reaching termination criteria and in the number of instances solved. Valid inequalities have also proven to give a speed-up of the computational time: symmetry-breaking cuts seem to be the most beneficial.
Further comparison between Cplex and the proposed B&P method should be performed on instances with different values for the interceptors capacity. A computational validation of the general IVRP model of Section 3.3 is also required.

Chapter 4

Waste Flow Optimization: An Application in the Italian Context

During the last decades, the solid waste management increased its already substantial influence on a variety of factors impacting on the entire society, especially for what concerns the economical and environmental issues. Waste logistic networks became articulated and challenging as the straightforward source-to-landfill situation switched to multi-echelon networks in which waste flows generally go through more than one preliminary treatment before reaching the final destinations. Complex optimization problems arises in this context, with the objective of maximizing the overall profit of the service. In this chapter we propose mixed-integer linear formulations, and relative resolution methods, for problems arising in the context of waste logistic management, with an application on a real world case study. In response to the actual needs of an important Italian waste operator, we propose the modeling of some relevant features of these problems, such as digester facilities, transportation economies of scale and temporary storages of the waste.

4.1 Introduction

Waste management is a priority for urban and rural communities throughout the world. The large and generally increasing amount of waste generated each year in industrialized and developing countries, along with the public concern for environmental preservation, is making such a problem one of the most relevant issues in modern societies. In this context, an integrated waste management process represents a real request and a difficult challenge at the same time, because it involves institutional, social, financial, economic, technical and environmental factors.

An important source of complexity in waste logistic network is given by the typical need to treat waste flows in various kinds of processing facilities before reaching a disposal plant or an external market. Such multi-echelon networks have been used to model waste management networks and solve waste flow allocation problems from an optimization point of view (see [113] for a comprehensive overview). Operations research may help the waste manager to decide how to ship the waste inside the network in order to minimize logistic costs and maximize possible revenue coming from energy produced or recyclables sold.

The aim of the chapter is to present mathematical models for solving the waste flow allocation problem at a strategic or tactical level. The construction of the model is motivated by the modeling of a case study for Herambiente, the largest Italian waste operator based in Emilia Romagna, Italy, and it has been incorporated into a Decision Support System (DSS) tool by Optit Srl, an accredited spinoff company of the Alma Mater University of Bologna, Italy. The results obtained with the DSS helped the waste operator in obtaining remarkable cost savings in the network management.

The remainder of the chapter is organised as follows. In Section 4.2, a description of the waste commodity classification and waste management network is given. Particular attention is given to the Italian situation by providing statistical data, however the multi-echelon structure of the waste network presented is common also in other European cases. In Section 4.3, the reasons for which Operations Research is used in waste flow management are exposed. A brief literature review on the topic is also given. Section 4.4 contains a valid formulation for solving a waste flow allocation problem at a strategic or tactical level. The model is inspired by a regional case study in Italy; however, the proposed constraints can be easily adapted to similar waste management networks. A set of model extensions addressing more specific features of the facility and waste management is also described. The case study constituted by the collaboration between Optit Srl and HeraAmbiente Spa is explained in Section 4.5. The results obtained with the DSS are discussed in Section 4.6 and some conclusions are drawn in Section 4.7.

4.2 Waste Management in Italy

The Italian legislation (D.lgs 152/06 art. 184 ([2])) defines two alternative criteria for waste classification: by source and by level of danger of the waste. The source-based classification makes a distinction between “Industrial” Waste (“Rifiuti Speciali”, in Italian) (IW) and “Municipal” Waste (“Rifiuti Urbani”, in Italian) (MW). Roughly speaking, the former includes the waste produced by industrial and commercial entities while the latter includes the waste produced by citizens and urban environment in general. The level of danger classification makes instead a distinction between “dangerous” and “non-dangerous” waste. The Italian legislation identifies different classes of danger, and lists all the specific types of possible waste that have to be considered dangerous for some reason (toxicity, flammability, etc.).

Summarizing data reported by the governmental agency ISPRA in [146, 147], for what concerns the MW production, a decreasing trend characterized the last years, in line with the European Union (EU) situation. Moving from year 2011 to 2012, the amount of MW produced by Italian municipalities decreased by 1.3% (2.4% in the EU), totalling about 29.5 millions of Mg (246.8 in UE) and yielding roughly the same amount measured in 2001 (note that 1 Mg = 10^6g). The MW represents around 20% of the total amount of waste produced every year while the remaining part is made up by IW production. The trend of IW production is not as clear as the one of MW. In 2010 the amount of produced IW increased by 2.4%, reaching 137.9 millions of Mg.

The Italian waste logistic network is articulated and challenging (see Figure 4.1) and reflects the complexity of the associated supply chain. In fact, Industrial and Municipal Wastes flow go through one or more preliminary treatments in specialized facilities before reaching the final destination. As a consequence, waste flows follow inter-city or inter-regional paths among a multi-echelon network, with logistics and transformation costs impacting the overall national economy. This generic overview highlights an heterogeneous situation in which rather critical situations (see, e.g., [66]) coexist with excellencies, resulting in a national context far from the straightforward "producer-to-landfill" system, but still struggling to compete with more virtuous strategies implemented in EU.

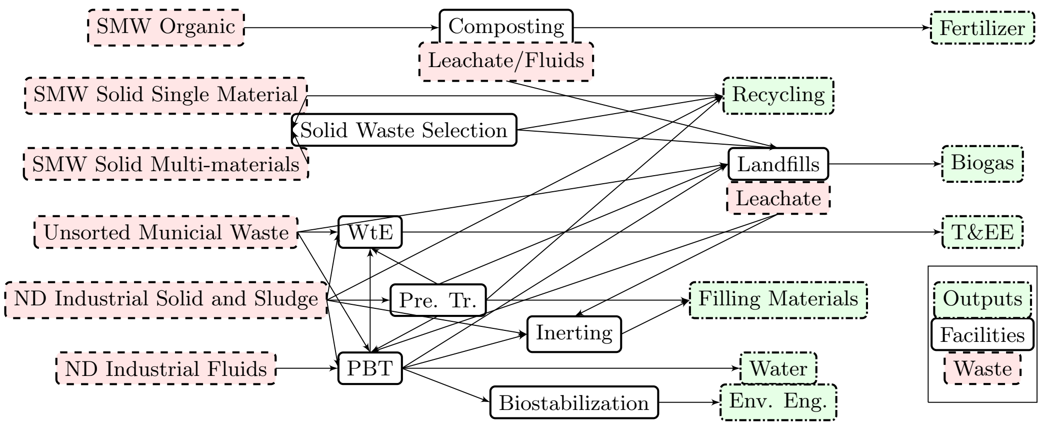


FIGURE 4.1: A diagram representing the typical waste facilities network. SMW stands for Sorted Municipal Waste, ND is Non-Dangerous, PBT is Phisiochemical Biological Treatment, WtE is Waste to Energy, T&EE is Termal and Electrical Energy, Env. Eng. is Environmental Engineering, Pre.Tr. is Preliminary Treatments (see [135], in Italian).

In the following we analyze in detail the various components of waste management in Italy with special attention to the territory managed by Herambiente, the largest Italian operator in the waste management marked that will be the focus of our case study.

4.2.1 Municipal Waste Management

4.2.1.1 Municipal Waste Production

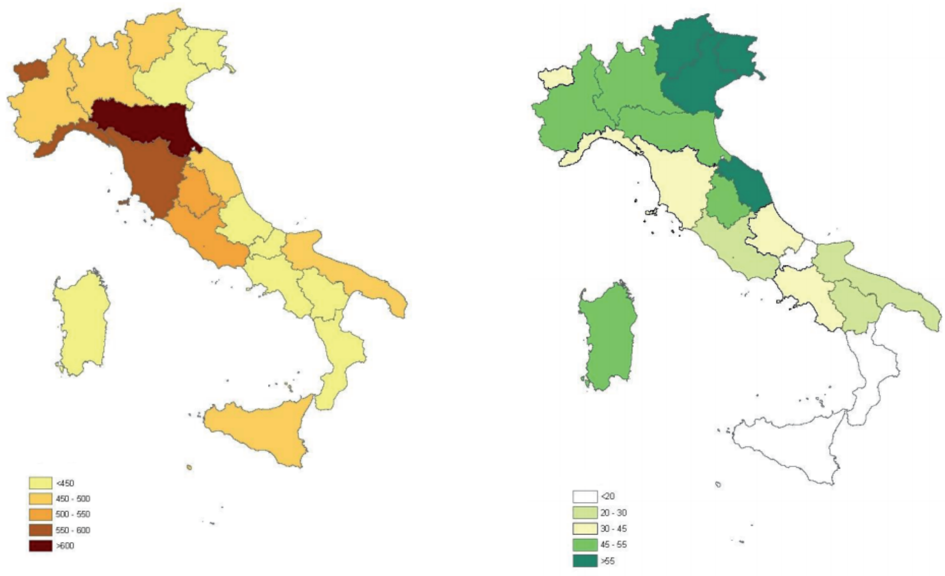
Figure 4.2a summarizes the waste production in Kg per capita in 2012. The region with highest amount of waste produced is Emilia-Romagna, with 625 Kg/capita produced, while the lowest production rate belongs to Basilicata, with 359 Kg/capita. Such differences in waste production may be motivated by the different economical and social situation in the Italian territory, in accordance with well-know relation between social-economical indicators (see, e.g., [78, 224]), such as the Gross Domestic Product (GDP). When considering the data of individual provinces, the Emilia-Romagna still represents a particularly interesting area, since 4 out of the 7 provinces have more than 650 Kg of waste per citizen produced in 2012.

4.2.1.2 Municipal Sorted Waste Collection

Two main sorted collection systems are active in the Italian territory: a “selective”, single material, sorted collection, and a “combined”, multi-material, sorted collection. Examples of projects implementing such models are reported in [115, 239], for instance. In a selective collection, the citizen sorts the single material and disposes it separately. In a combined collection system, the citizen sorts a group of materials and disposes all of them in the same waste bin. Combined collection systems are not uniform among the territory, but different strategies are adopted by different players (also, occasionally, in different subareas controlled by the same player). ISPRA estimated that almost 1.2 million of Mg has been collected via combined sorted collection during year 2012. Given the total amount of waste collected via combined systems, about 36% of them is composed by plastic materials, 29% glass, 11% paper, 7% is metallic materials, 1% wood and the remaining part can be considered as residual unsorted MW.

Overall, at a national level, out of the total amount of sorted waste collected 38% is estimated to be biodegradable, 28% paper, 15 % glass, 8% plastic materials, 6% wood, 2% metal, 2% electronic, and 1% textile. Such percentages consider sorted waste collected with both selective and combined systems.

For what concerns the Municipal Solid Waste (MSW), the highest rate of sorted collection in 2013 was registered in the Veneto region, where the 64.6 % of municipal waste was collected as sorted waste. The Emilia Romagna region, subject of our study in Section 4.5, went from a 45.6 % to a 53.0% during the same year. Figure 4.2b displays the percentage of sorted waste collected in each region in 2013.



(A) Yearly waste production in Kg for citizen (B) Percentage of Sorted Collection

FIGURE 4.2: Regional production of municipal waste per capita and regional percentage ratio of sorted municipal waste collection (source ISPRA [147])

4.2.1.3 Municipal Waste Treatment and Disposal

In Figure 4.1 a typical path for waste flow treatment and disposal is represented. Among the total amount of MW produced in Italy, 41% of it finds landfills as their final destination, while 18.2% is treated in incineration plants, 26% is recycled and 14.6% goes through biological treatment to become fertilizing materials (see [147]). In almost all the cases, MW is processed in one or more facilities before reaching the final destination and, generally, it changes its composition and classification several times during the process. A common intermediate process regards the MW collected via combined systems. In this case, typically, waste is directed to "Multi-Material Treatment Facilities" (MMTF). Such facilities may vary from manual separation to automatic separation plants and share the ability of sorting the single material that are combined in the collection phase. Generally speaking, around 15% of waste can not be recycled and is directed to landfills. The remaining percentage can be considered together with MW collected via selective systems.

Another typical intermediate step consists in Physico-chemical and Biological Treatment (PBT). About 58% of waste directed to landfills and the 53% of waste directed to incinerators is subject to a mechanical-biological treatment before reaching the respective destination. In Italy around 9 million of Mg of MSW receive a mechanical-biological treatment before being sent to other facilities, landfills, or incinerators. Remarkable examples of such processes take place in composting and digesters systems for organic waste.

Waste production is, in some cases, associated with landfills and plants. An example of such production is represented by the waste stocked in landfill, which produces leachate with different compositions for decades (see e.g. [90, 162, 165, 213]). Leachates from landfills require physiochemical or stabilization treatments (see [213, 241]) and must be routed to the relative facilities. Typically, the outgoing waste from such processes in facilities has an unknown relation with the incoming waste in the same facility. Therefore, the waste operator may prefer to consider the waste as generated from a plant or landfill and uncorrelated with the facility input flow.

Similar considerations can be done for composting systems. They generate leachates and fluids over time with quantities that are not easily predictable from the incoming waste, since their production is also dependent on weather conditions and climate in general. Leachate from composting facilities has a substantial different composition than the landfill leachates. The main difference is that they can be disposed directly in landfills without being treated in other processing facilities.

4.2.2 Industrial Waste Management

4.2.2.1 Industrial Waste Production

In Figure 4.3 are reported the percentage data on IW Italian production during 2010, which is the last year with available information from ISPRA.

The IW is subdivided into Dangerous and Non-dangerous waste. This sharp distinction is due to different composition and characteristics that lead to specific treatment processes and disposal systems. The Dangerous Industrial Waste (DIW) formed in 2010 the 8.2% of the total amount of IW. Concerning the Non-dangerous IW, construction and demolition wastes correspond to 46.2 % of the total amount of IW. Waste produced by manufacturing correspond to 26.4% of the total amount, followed by wastes originated by MW treatments corresponding to 20.2%. According to the EU Regulation No. 2150/2002 ([1]), ISPRA recorded around 35 million Mg of mineral waste deriving from construction and demolition followed by soil for 15 million Mg (see Figure 4.4a for the complete description of Non-dangerous IW quantities). For what regards the DIW, 47.8% of them derives from manufacturing processes, while 24.4% comes from commercial and logistic activities and 18.4% is generated during MW treatments (see Figure 4.4b). Only the 4.8% of DIWis originated from construction and demolitions. ISPRA recorded 2.5 million of Mg of industrial slug as DIW(see Figure 4.4b), while the other two categories with more that 1 million Mg are dismissed vehicles (1.6 million Mg) and chemical wastes (1.3 million Mg).
The nine regions composing the north of Italy produced around 77 million Mg of IW, which are 56% of national production. Lombardia, Veneto and Emilia-Romagna are the three regions with the largest value of IWproduction during 2010, with 23.8, 16.8,

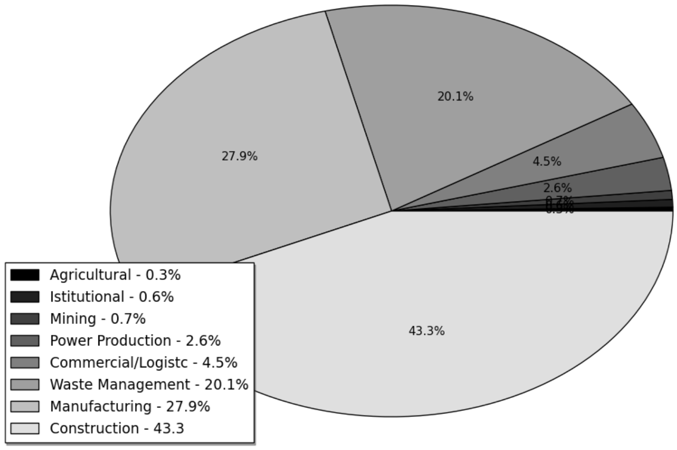


FIGURE 4.3: Percentage subdivision of IW total production in 2010.

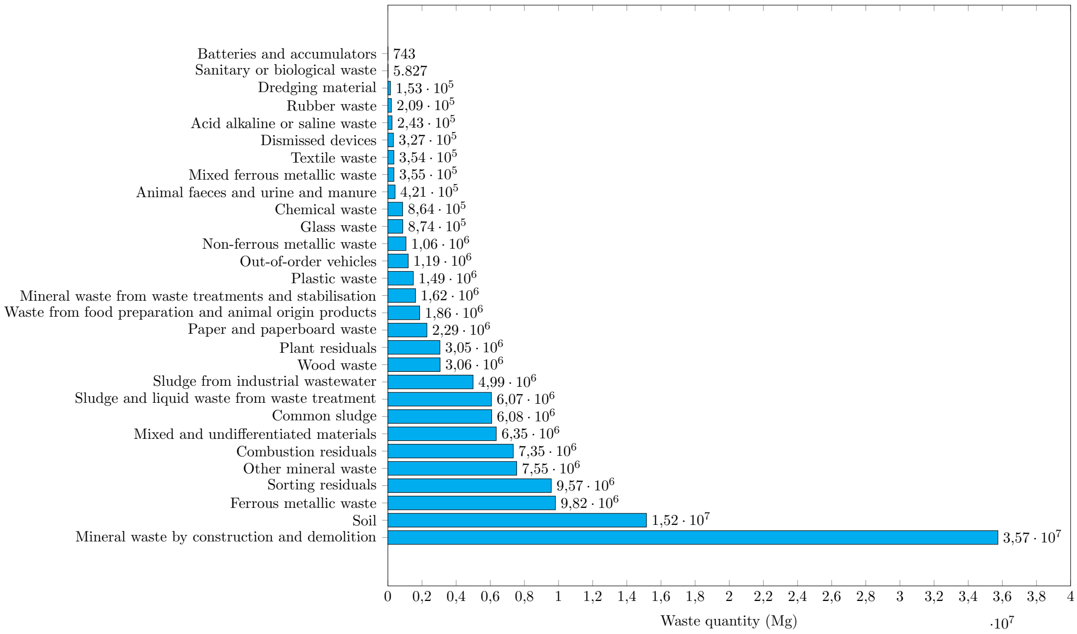
and 14.2 million Mg, respectively. Emilia-Romagna recorded the largest growth in IW production going from 2009 to 2010, with a net increase of 1.4 million of Mg.

4.2.2.2 Industrial Waste Collection

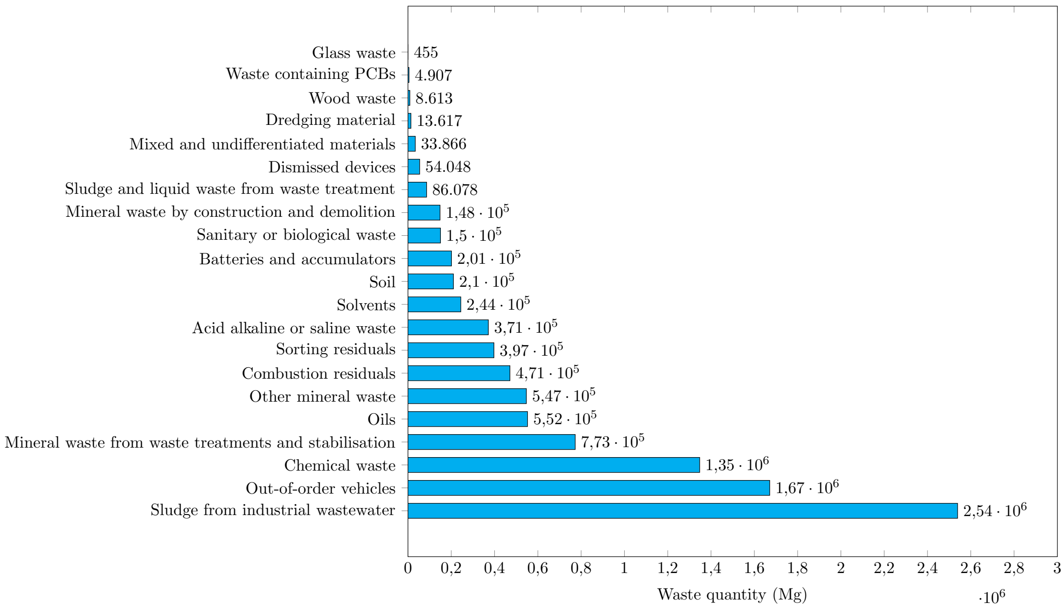
IW collection systems are not uniform in the Italian territory. Generally, IW producers take charge of waste hauling it to the appropriate facilities, after a preliminary agreement with the waste operator. Three players cover specific roles in IW collection: the *waste producer*, which is typically an industry operating in the private sector; the *carrier*, which is a logistic company with specific legal authorization for waste transportation; and the *waste operator*, which is a company owning or managing waste facilities. In most cases, producer, carrier and waste operator are three different subjects. However, when the producer takes over the waste transportation to the appropriate facility, the producer and the carrier are considered the same entity. In other cases, the waste operator may offer transport services, therefore the carrier is also the waste operator. Finally, the three operators coincide when IW is produced in a waste treatment facility.

4.2.2.3 Industrial Waste Treatment and Disposal

During 2010, only 12.1% of IW was disposed in landfills and 2.3% was converted into energy, while 84.6% was recycled. Such differences in destination with respect to the MW are mainly due to a different composition of the waste. Clearly, construction and demolition waste are not suitable to be converted to energy, whereas they are easily recyclable as filling materials. Such kinds of waste compose the large part of IW.



(A) Italian Non-dangerous IW production according to EU Regulation 2150/2002 coding. Data for year 2010.



(B) Italian DIW production according to EU Regulation 2150/2002 coding. Data for year 2010.

The IW often goes through one or more intermediate treatments, as happens for MW. A main role in preliminary treatment is played by physiochemical and biological systems, to which 16.5 million Mg have been directed during 2010, observing an increase of more than 4 million Mg with respect to 2009. Before any final destination or treatment process, included the physiochemical, some facilities performing preliminary operations or simply temporary stocking may come into play. In 2010, around 2.5 million Mg followed this first steps.

4.3 Waste Flow Optimization

The reasons why Operations Research (OR) techniques can be profitably used in waste management decision making are various.
For countries in the EU_{25} group, the municipal solid waste generated per year has reached the value of approximately 100 millions of Mg at the end of XX century. Such waste production rate is expected to face an increasing trend in the next 15 years. Similar amounts are disposed in landfills (see Mazzanti and Zoboli [189]). It is clear that such a huge amount of waste have to be collected, transferred, transformed and disposed while taking into account a variety of factors, such as social, political, legal, economic, environmental and technical implications (Wilson et al. [253]).

Also, for what concerns the IW, regardless of the production rate, it is important to handle these flows with special care. As explained in Section 4.2.1.3, the waste treatment processes consist in complex operations performed by several plants, with large differences in input and output products (see also Singh et al. [229]). The production of liquid waste such as leachates or industrial sludges has to be specifically considered, since their treatment is affected by environmental and technical implications.
The waste managers are therefore facing complex and relevant issues for modern societies. In this context, a mathematical model can describe the specific features of the network of waste treatment facilities and of the waste generation. OR methods will then help to determine the best planning strategy according to given optimization criteria. An extended and recent survey on the application of OR methodologies to Solid Waste Management is given by Ghiani et al. [113].

In problems in which the waste flow is a decision variable, one of the most important and used optimization criterion is that of minimizing the total transportation and processing cost, minus all revenue for reclaimed material and generated energy ([113]). Generally, the models proposed in literature can be considered as a multiperiod multicommodity flow with multiple sources and sinks. When the selection of the operating facility in each period is taken into account, a facility location component can be also identified in the model. Because of the large number of waste facilities features an OR model for the waste management should be tailored to the characteristics of the case study. General purpose models would be too hard to formulate or solve.

A major aspect to be taken into account in the model formulation is the time horizon in which the planning has to be made. Two planning levels are usually considered.
In the strategic level, long-term decisions have to be made at a regional level. Generally, the problem is to select which facilities to use and how to ship the waste in each period of the time horizon in order to minimize waste processing and transportation costs. Furthermore, if the time horizon involves more than four or five years, the expansions of the existing plants as well as the building of new facilities may be considered (see, e.g., Baetz et al. [19], Li and Huang [173], Vigo et al. [247]).

At the tactical level, short and medium term decisions have to be performed. Although the literature is still relatively scarce in this area, OR models can be profitably applied to incorporate operational issues, such as: waste flow allocation according to short term forecasts and aggregation of waste sources and commodities (see section 4.5.2 for further details), the districting phase, the collection sites location (Ghiani et al. [114]), the selection of the collection days and the determination of fleet and crew composition that performs the waste collection (Ghiani et al. [112]). The present chapter addresses waste flow allocation problems.

Another factor that influences the mathematical formulations for waste management is the uncertainty that affects the data related to waste generation rates, processing and transportation costs and revenues at the time of the decision making. The reader can refer to Sun et al. [233] for a recent survey on inexact programming methods for solving waste management problems with uncertain data. Stochastic parameters can be expressed with interval data, random variables with given probability distributions, or fuzzy sets. In such stochastic context, the selection of the solution method to be applied is strongly dependent on the capability of the waste manager to adopt robust decisions or rather use flexible planning strategies and the modality in which uncertain parameters are available and how uncertainty is revealed in the planning horizon. For instance, a Two-Stage Stochastic Programming formulation (Birge and Louveaux [38]) is commonly adopted when the waste manager is able to take a recourse action when the flow waste turns out to exceed the forecasted amount (see, e.g., Li and Huang [173], Maqsood and Huang [187]).

In the present chapter, all problem parameters are deterministic data obtained by using forecasting methods for the waste generation in the future planning period. The amount of historical data available in Optit is not sufficient for estimating stochastic tools such as probability distributions of uncertain parameters. A wide and general dissertation on demand forecasting techniques in logistic systems can be found in Ghiani et al. [111]. An accurate prediction of municipal solid waste generation is both an important and challenging task in a waste management problem (Dyson and Chang [84]). While traditional forecasting methods have taken into account demographic and economic factors on a per-capita basis, researches have shown that population growth and migration are not the only factors influencing the forecast. In addition to them, climate changes, employment status, education, social and public attitudes affect the waste generation interactively (Bandara et al. [21]). In developing countries, the waste forecast can be made with respect to the economic activity of the city by using regression modeling and time series analysis (Rimaityte et al. [215]). A vast survey on formulations for the municipal solid waste generation using economical, social, demographic and management-orientated data can be found in Beigl et al. [28].
A common approach in literature is to describe the waste management system as a multi-echelon supply chain (see, e.g., Ghiani et al. [113], Zhang et al. [261]). According to this assumption, the waste network can be considered having a *sources* - *facilities* -

destinations hierarchy. Waste generation sources are network nodes in which municipal and industrial waste is generated in each period and has to be shipped inside the network. Waste treatment, separation and composting facilities are plants in which both ingoing and outgoing flow are allowed. Destination sites are landfills and disposal markets in which the waste is required to be disposed.

4.4 Model Formulation for Waste Allocation Problems

In this section we introduce the model for solving the Strategic Waste Flow Allocation (SWFA) problem. This formulation is devoted to the solution of a wide range of waste allocation problems. The model is inspired by the case study in section 4.5.

The SWFA network is made up by the set of nodes V and the set of arcs A. In principle, each municipal collection area is considered as a waste production *source* node, although several homogeneous areas are often aggregated into a single node to reduce the size of the network. Similarly, industrial sites or their aggregations are included in the set of source nodes of the network. Note that source nodes have only outgoing flows and no ingoing ones. Furthermore, no limit on the outgoing flows from the sources is generally present.

Each *intermediate facility* is represented by two different nodes in V: one such node represents the plant itself that receives waste and, after the processing, sends waste, possibly of different types, to other nodes in the network. The transformation between different type of waste due to the processing done at a plant is modeled through a set of *transformation coefficients* $b_{vww'}$ of a unit of waste w' into w at plant j. The second node is a, possibly fictitious, waste production site which allows for modeling complex outputs of the plant that are not proportional to the input waste quantities, such as the leachate production explained in Section 4.2.1.3. Limits on ingoing and outgoing flows at intermediate facilities may be imposed, both for specific waste types and for the total.

The waste flow can be disposed in *destination nodes*, which correspond to landfills or markets for recycled products and energy (e.g., produced in waste-to-energy facilities). The destination nodes are grouped in node set V_L. A destination plant is characterized by the absence of outgoing waste flows.

The model takes into account real-world restrictions on the outgoing and ingoing waste flow in processing facilities, transfer stations and landfills. Such limitations arise from logistic, technical and environmental issues. Constraints on both absolute and relative flows of different waste commodities (i.e., types) are considered, along with compulsory deactivation periods for subsets of facilities.

To define the model we introduce the following notation:

Sets

V	set of waste network nodes
V_O	subset of V including the source nodes
V_F	subset of V including the intermediate facility nodes
V_L	subset of V including the destination nodes (e.g., landfills and markets)
A	set of network arcs corresponding to feasible waste shipments between nodes
W	set of waste commodities (types)
δ_v^+	set of arcs outgoing from node v
δ_v^-	set of arcs entering in node v
Θ_v^+	subset of commodities that can leave node v globally
Θ_v^-	subset of commodities that can enter node v globally
Ω_v^{t+}	subset of commodities that can leave node v in period t
Ω_v^{t-}	subset of commodities that can enter node v in period t
$\mathcal{W}_v^{t+}$	set of commodity pairs that can leave node v in period t
$\mathcal{W}_v^{t-}$	set of commodity pairs that can enter node v in period t
$\mathcal{D}$	set of facilities for which a deactivation is compulsory during the planning horizon

Parameters

T	number of time periods of the planning horizon
c_{awt}	unit transshipment cost for waste commodity w on arc a in period t
p^t_{vw}	unit profit or cost (if < 0) for commodity w leaving node v in period t
r^t_{vw}	unit profit or cost (if < 0) for commodity w entering node v in period t
G^t_{vw}	quantity of waste commodity w generated in node v in period t
$b_{vww'}$	transformation coefficient for a unit of waste commodity w' into the waste commodity w in node v
$\underline{C}^{t+}_{vS}$, $\overline{C}^{t+}_{vS}$	minimum and maximum quantity of commodities in set S leaving node v in period t
$\underline{C}^{t-}_{vS}$, $\overline{C}^{t-}_{vS}$	minimum and maximum quantity of commodities in set S entering node v in period t
$\overline{\alpha}^{(S,S')+}_v$, $\underline{\alpha}^{(S,S')+}_v$	superior and inferior limit for the outgoing flow from node v of commodities in set S as a percentage of outgoing flow of node v of commodities in set S'
$\overline{\alpha}^{(S,S')-}_v$, $\underline{\alpha}^{(S,S')-}_v$	superior and inferior limit for the ingoing flow in node v of commodities in set S as percentage of ingoing flow of node v of commodities in set S'
$\overline{\Gamma}^+_{vS}$, $\overline{\Gamma}^-_{vS}$	maximum overall outgoing and ingoing flow of commodities in set S for node v
D^t_v	duration of the deactivation for facility v starting in period t

Decision variables

x^t_{aw}	amount of waste flow of commodity w to ship in arc a in period t
z^t_v	binary variable assuming value 1 if facility v is active in period t or 0 otherwise
ρ^t_v	binary variable assuming value 1 if facility v is starting its deactivation term in period t or 0 otherwise

A valid model for the SWFA problem is formulated as follows:

$$\min \quad \sum_{w\in W}\sum_{a\in A}\sum_{t=1}^{T} c_{awt}x^t_{aw} - \sum_{w\in W}\sum_{v\in V\setminus V_O}\sum_{t=1}^{T} p^t_{vw}\sum_{a\in\delta^+_v} x^t_{aw} - \sum_{w\in W}\sum_{v\in V\setminus(V_O\cup V_L)}\sum_{t=1}^{T} r^t_{vw}\sum_{a\in\delta^-_v} x^t_{aw} \tag{4.1}$$

s.t.

$$\sum_{a \in \delta_v^+} x_{aw}^t = G_{vw}^t \qquad \forall\ w \in W,\ v \in V_O,\ t = 1, \ldots, T, \tag{4.2}$$

$$\sum_{a \in \delta_v^+} x_{aw}^t = \sum_{w' \in W} b_{vww'} \sum_{a \in \delta_v^-} x_{aw'}^t \qquad \forall\ w \in W,\ v \in V_F,\ t = 1, \ldots, T, \tag{4.3}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^+} x_{aw}^t \leq \overline{C}_{vS}^{t+}\ z_v^t \qquad \forall\ S \in \Omega_v^{t+},\ v \in V_F,\ t = 1, \ldots, T, \tag{4.4}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^+} x_{aw}^t \geq \underline{C}_{vS}^{t+}\ z_v^t \qquad \forall\ S \in \Omega_v^{t+},\ v \in V_F,\ t = 1, \ldots, T, \tag{4.5}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^-} x_{aw}^t \leq \overline{C}_{vS}^{t-}\ z_v^t \qquad \forall\ S \in \Omega_v^{t-},\ v \in V \setminus V_O,\ t = 1, \ldots, T, \tag{4.6}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^-} x_{aw}^t \geq \underline{C}_{vS}^{t-}\ z_v^t \qquad \forall\ S \in \Omega_v^{t-},\ v \in V \setminus V_O,\ t = 1, \ldots, T, \tag{4.7}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^+} x_{aw}^t \leq \overline{\alpha}_v^{(S,S')+} \sum_{w' \in S'} \sum_{a \in \delta_v^+} x_{aw'}^t \qquad \forall\ (S, S') \in \mathcal{W}_v^{t+},\ v \in V_F,\ t = 1, \ldots, T, \tag{4.8}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^+} x_{aw}^t \geq \underline{\alpha}_v^{(S,S')+} \sum_{w' \in S'} \sum_{a \in \delta_v^+} x_{aw'}^t \qquad \forall\ (S, S') \in \mathcal{W}_v^{t+},\ v \in V_F,\ t = 1, \ldots, T, \tag{4.9}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^-} x_{aw}^t \leq \overline{\alpha}_v^{(S,S')-} \sum_{w' \in S'} \sum_{a \in \delta_v^-} x_{aw'}^t \qquad \forall\ (S, S') \in \mathcal{W}_v^{t-},\ v \in V \setminus V_O,\ t = 1, \ldots, T, \tag{4.10}$$

$$\sum_{w \in S} \sum_{a \in \delta_v^-} x_{aw}^t \geq \underline{\alpha}_v^{(S,S')-} \sum_{w' \in S'} \sum_{a \in \delta_v^-} x_{aw'}^t \qquad \forall\ (S, S') \in \mathcal{W}_v^{t-},\ v \in V \setminus V_O,\ t = 1, \ldots, T, \tag{4.11}$$

$$\sum_{t=1}^{T} \sum_{w \in S} \sum_{a \in \delta_v^+} x_{aw}^t \leq \overline{\Gamma}_{vS}^+ \qquad \forall\ S \in \Theta_v^+,\ v \in V_F, \tag{4.12}$$

$$\sum_{t=1}^{T} \sum_{w \in S} \sum_{a \in \delta_v^-} x_{aw}^t \leq \overline{\Gamma}_{vS}^- \qquad \forall\ S \in \Theta_v^-,\ v \in V \setminus V_O, \tag{4.13}$$

$$\sum_{i=0}^{\min\{D_v^t-1,T-t\}} z_v^{t+i} \leq \min\{D_v^t, T-t+1\}(1-\rho_v^t) \qquad \forall\ v \in \mathcal{D},\ t = 1, \ldots, T, \tag{4.14}$$

$$\sum_{t=1}^{T} \rho_v^t \geq 1 \qquad \forall\ v \in \mathcal{D}, \tag{4.15}$$

$$x_{aw}^t \geq 0 \qquad \forall\ w \in W,\ t = 1, \ldots, T,\ a \in A, \tag{4.16}$$

$$z_v^t \in \{0, 1\} \qquad \forall\ v \in V \setminus V_O,\ t = 1, \ldots, T, \tag{4.17}$$

$$\rho_v^t \in \{0, 1\} \qquad \forall\ v \in \mathcal{D},\ t = 1, \ldots, T. \tag{4.18}$$

The model objective function and constraints are explained in detail in the following subsections. An overview on additional features is also presented.

4.4.1 Objective Function

The total flow transportation costs over all network arcs has to be minimized. More precise considerations on the expression of such costs are given in Section 4.4.5.
The objective function also takes into account two additional terms associated with flow processing net profits (or costs) that must be maximized. The first term is associated with the outgoing flows from the facilities while the second is associated with the ingoing flows to facilities and landfills. Profits and costs are considered the net unit value of all different profits and costs associated with the processing of a unit of flow, being negative when the costs prevail on the revenues for that specific waste and plant. Furthermore, net profits and costs can be dependent on the specific period, for example when considering the production of heat energy. In case negative parameters are present in the objective function, in some feasible solutions the waste flow of the same commodity may be transported in closed cycles: this situation can be avoided by appropriately modifying model and parameters setting, such as forbidding wrong arcs or introducing different names for outgoing waste flow commodities.

Note that the chapter focuses on medium and short term planning horizons. In such a context, the possibility of closing or opening facilities in the network is not realistic. Therefore, plant activation costs are not considered in the objective function.

4.4.2 Flow Balance

Constraints (4.2) and (4.3) ensure that all waste generated in network nodes is collected and shipped inside the network. The ingoing flow in the facilities of $V \setminus V_O$ is transformed according to a transformation coefficient b, which expresses the output quantity for a unit of incoming waste. Note that the transformation coefficient is not necessarily a reduction coefficient, since additional material may be needed for producing output waste (e.g., inerting treatments requiring whitewash supplement).

4.4.3 Flow Limitation

Constraints (4.4)-(4.11) impose the restrictions on outgoing and ingoing flow waste for subsets of waste commodities both in a absolute and a relative manner. Absolute limitations in a plant are valid only in its operating periods. Constraints (4.12) and (4.13) ensure that outgoing and ingoing flow of specific subsets of commodities are bounded over the entire planning horizon. Such overall limitations can be particularly appropriate for landfills that typically have a yearly capacity.
The flow limitation constraints are explicitly introduced only for specified critical subsets of commodities established by the waste manager. Indeed, the cardinality of the subsets of commodities is not polynomial in the problem size, hence a massive imposition of such groups of constraints would render the model computationally intractable.

4.4.4 Facility Deactivation

In constraints (4.14) and (4.15), facility deactivation periods are managed. A plant can be subject to maintenance operations for technical and issues.
The constraints ensure that, after starting the deactivation in period t, the facility j is not operational for D_j^t consecutive periods, or until the end of the planning horizon.

4.4.5 Economies of Scale

Cost structure analyses on waste transportation indicate that the relation between flow waste and transporting cost is not correctly expressed by a linear function (Callan and Thomas [51]). The actual situation is that unit transportation cost, i.e., the slope of the linear relation, decreases with increasing levels of waste flow because: (i) fixed cost are distributed over more units of output, and (ii) the improvement of operational efficiency when considering large scale of input waste flow (Abrate et al. [3]). This behavior is known as *economy of scale*. As a result, the arc transportation cost should be modeled as a piecewise linear function of the waste flow in the arc. A set of thresholds ξ_i, $i \in \{0, \ldots, N\}$ for the waste flow is given; in each interval between two thresholds, an affine function for computing the transportation cost is known. Hence, the transportation cost function is represented as:

$$c_{aw}(x^t_{aw}) = a_i + c_{aw,i}\; x^t_{aw} \quad \text{for} \quad x^t_{aw} \in [\xi_{i-1}, \xi_i],\; i = 1, \ldots, N,$$

where $\xi_0 = 0$ and $\xi_N = M$, with M sufficiently large upper bound.
The transportation cost is thus a concave continuous function of the waste flow, mathematically expressed by the monotonic decrease of the set of slopes $\{c_{aw,i}\}_{i=1,\ldots,N}$. In Figure 4.5 an example of economy of scale for transportation costs is depicted.

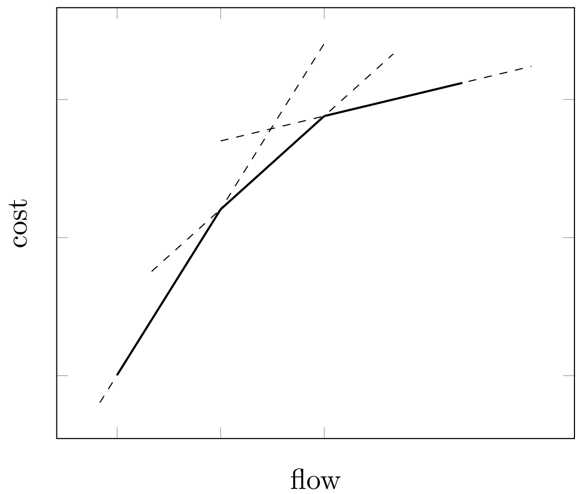


Figure 4.5: An example of concave piecewise linear cost function of the waste flow

In order to model the concave piecewise linear function c_{aw}, convex combination models may be adopted (see Croxton et al. [72], Vielma et al. [246]). They require the introduction of binary variables for the selection of threshold levels. The binary variables turn out to be Special Ordered Set of type II (SOS2) variables (Beale and Tomlin [27], Tomlin [238]): the SOS2 declaration has the advantage to convey a tailored branching for the MILP solution ([31]).

4.4.6 Additional Features

The proposed model (4.1)-(4.18) is a valid formulation for the tactical waste flow allocation, which takes into account a set of realistic characteristics for the waste management, such as transformation coefficients, limitation of flow in plants and facility deactivation periods.

Several other specific characteristics of the problem can be embedded in the model. In the remainder of the section we introduce three examples of such features.

4.4.6.1 Digester Facilities

Organic waste may be treated in *digesters*, in which anaerobic digestion is performed, in contrast to composting facilities characterized by aerobic processes.
A detailed formulation of the digesters operating principles is out ot the purpose of the strategic level management. Digesters are subject to a classification according to fuelling frequency operations, with a major distinction in Batch systems and continuous digesters (i.e., continuous flow stirred-tank reactors and plug flows). For an overview of anaerobic digestion processes and issues, see, e.g. Mata-Alvarez et al. [188].
In a digester, the chemical processes that ingoing waste face may require several months. Therefore, in planning problems in which the time is discretized in weekly or monthly units, single-period flow balance constraints (4.3) would not be applicable to digesters.

To model such facilities some additional notation is required. Let V_D be the set of digester facility nodes, and S_{vw} be the set of waste commodities w' required to produce output waste w in digester $v \in V_D$. The flow of output commodity w will be ready after τ^t_{wv} periods. We assume such processing time, τ^t_{wv}, to be dependent on the starting period t, on the used digester $v \in V_D$ and the output waste w.

The flow balancing constraints for digester facilities are then:

$$\sum_{a \in \delta_v^+} x_{aw}^{t+\tau^t_{wv}} = \sum_{w' \in S_{vw}} b_{vww'} \sum_{a \in \delta_v^-} x^t_{aw'} \quad \forall\, w \in W,\ v \in V_D,\ t \in \mathcal{T}_{wv}. \tag{4.19}$$

Note that constraints (4.19) are only valid in the set $\mathcal{T}_{wv}$ of periods for which the ingoing flow of waste in S_{vw} in digester v is entirely processed before the end of the planning horizon, i.e., $\mathcal{T}_{wv} = \{t \in \{1, \dots, T\} : t + \tau^t_{wv} \leq T\}$.

The output waste from digester v over the period T will be considered as an internal production of v in the next planning term horizon and managed with production constraints.

Typically, the waste amount entering a digester must be greater than a certain threshold to activate the digestion process. In addition, the incoming flow is subject to the maximum flow constraints. The limitations of the digester capacity γ_{wv} is expressed by constraints (4.20) below.

$$\sum_{w' \in W} b_{vww'} \sum_{\{t' \leq t : t' + \tau^{t'}_{wv} > t\}} \sum_{a \in \delta_v^-} x^{t'}_{aw'} \leq \gamma_{wv} \quad \forall\, v \in V_D,\ w \in W,\ t = 1, \dots, T. \tag{4.20}$$

4.4.6.2 Temporary Storage

The possibility of temporary waste storage at the facilities and at special waste generation sources such ad Single-Node Super-Sources (SNSS) (see Section 4.5.2.2) is considered in this section. Here the assumption is that the waste operator has the possibility of storing, for a limited amount of time, some type of waste nearby the facility location. The necessity of temporary waste storage may arise for several reasons. Recently, with the Council Directive 1999/31/EC and 2008/98/EC, the EU strongly discouraged the direct disposal of waste at landfills. In order to satisfy facilities capacity requirements, the number of several temporary waste storage sites has increased (see, e.g., [248]). The decision of temporarily storing waste is in some cases delicate and costly ([145]). In a nearly opposite but still realistic situation (see section 4.5), waste operators decide for temporary waste storage for economical reasons.

We hereby model this possibility as a particular case of the well known lot-sizing problem (see, e.g., [152]). To the best of our knowledge, temporary storages have not been generally included in waste flow optimization models from the literature. The waste temporary storage is allowed at the nodes in set $V_{\mathcal{S}} \subset (V_O \cup V_F)$. The set $V_{\mathcal{S}}$ is made up by plants in which municipal and industrial waste are produced and structures for the temporary storage are present. This type of storage has a different nature from the one present in digesters: the storage in digesters is due to the nature of the waste treatment processes and the waste manager cannot decide for a shorter storage times with respect to that required by the processing.

Continuous non-negative variables I^t_{wv}, bounded by γ_{wv} are introduced to measure the quantity of waste flow of commodity w stored in node v in period t. In the first period, the I^t_{wv} variables are initialized from the previous planning horizon. In the presence of temporary storage, flow balance is expressed by constraints (4.21).

$$G^t_{vw} + \sum_{a \in \delta^-_v} x^t_{aw} + I^{t-1}_{wv} - I^t_{wv} = \sum_{a \in \delta^+_v} x^t_{aw} \quad \forall\, v \in V_{\mathcal{S}},\ t = 2, \dots, T,\ w \in W. \tag{4.21}$$

By defining h^t_{wv} as the unit holding cost for waste type w in plant v during period t, the term

$$\sum_{w \in W} \sum_{t=1}^{T} \sum_{v \in V_{\mathcal{S}}} h^t_{wv} I^t_{wv}$$

has to be added in the objective function.

4.4.6.3 Logic Constraints on Incoming Waste Flow

The incoming waste flow at a facility can be subject to specific regulations regarding the mix of several waste commodities, for example to grant a sufficient calorific power at an incinerator. Restrictions on the mix of entering waste are expressed by logic constraints, which are typically formulated by using binary auxiliary variables. A massive introduction of binary variables affects the tractability of the model; hence, these constraints are imposed only for a limited subset $\tilde{V}$ of facilities.

In the following we provide an example of constraints that impose that the incoming waste of commodity w_2 has to be greater than the fraction $\sigma^t_{w_2v}$ of the incoming flow whenever the incoming waste of commodity w_1 is larger than a fraction $\sigma^t_{w_1v}$ of the incoming flow The subset of commodity pairs for which such constraints are defined is restricted to $\tilde{W}_1 \times \tilde{W}_2 \subset W \times W$.

$$\sum_{a\in\delta_v^-} x^t_{aw_1} \geq \sigma^t_{w_1v} \sum_{w\in W}\sum_{a\in\delta_v^-} x^t_{aw} - M_1(v,t)(1-y^t_{w_1v}) \quad \forall\ v\in\tilde{V},\ w\in\tilde{W}_1,\ t=1,\dots,T \tag{4.22}$$

$$\sum_{a\in\delta_v^-} x^t_{aw_2} \geq \sigma^t_{w_2v} \sum_{w\in W}\sum_{a\in\delta_v^-} x^t_{aw} - M_2(v,t)(1-y^t_{w_2v}) \quad \forall\ v\in\tilde{V},\ w\in\tilde{W}_2,\ t=1,\dots,T \tag{4.23}$$

$$y^t_{w_1v} \leq y^t_{w_2v} \quad \forall\ (w_1,w_2)\in\tilde{W}_1\times\tilde{W}_2,\ v\in\tilde{V},\ t=1,\dots,T \tag{4.24}$$

$$y^t_{wv} \in \{0,1\} \quad \forall\ \ w\in W,\ v\in\tilde{V}, t=1,\dots,T, \tag{4.25}$$

where $M_1(v,t)$ and $M_2(v,t)$ are suitably large constants. Assuming the capacity parameter $\overline{C}^{t-}_{vW}$ is known, then a possible value for such a constants are:

$$M_1(v,t) = \sigma^t_{w_1v}\overline{C}^{t-}_{vW}$$
$$M_2(v,t) = \sigma^t_{w_2v}\overline{C}^{t-}_{vW}.$$

4.5 Case Study

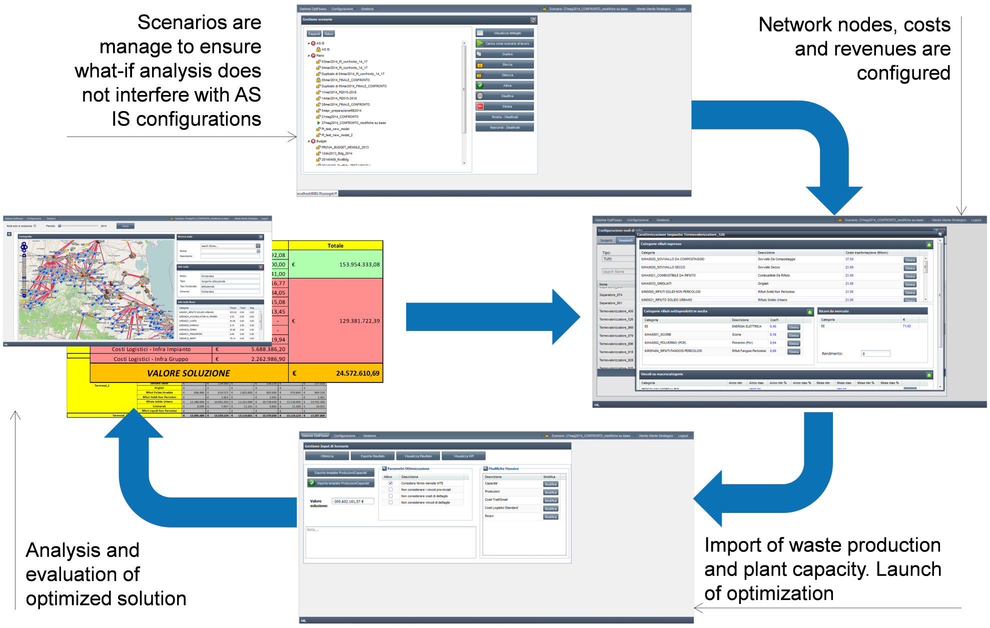
In this section, we present the results of the use of the model of Section 4.4 in a Decision Support System (DSS), called OptiWasteFlow, developed by Optit Srl, an accredited spinoff company of the University of Bologna, for Herambiente SpA. Herambiente is the largest waste operator in Italy and one of the largest in Europe, serving about 190 municipalities and 2.7 millions of citizens. In addition to municipal waste, Herambiente manages the flows of industrial waste coming from more than 60,000 private customers.

In more than 3,000 cases, Herambiente operates also as waste carrier, collecting directly the waste from the producer. More that 5 millions of Mg of waste every year are moved, under the managment of Herambente, from sources through facilities to final destinations. In line with the Italian and European situation, around 75% of waste treated by Herambiente is Industrial Waste, while the remaining 25% is Municipal Waste. Over 100 of different types of waste are routed over a network of almost 80 facilities directly controlled by Herambiente and several hundreds of facilities owned by third-party companies and located in centre-north of Italy.

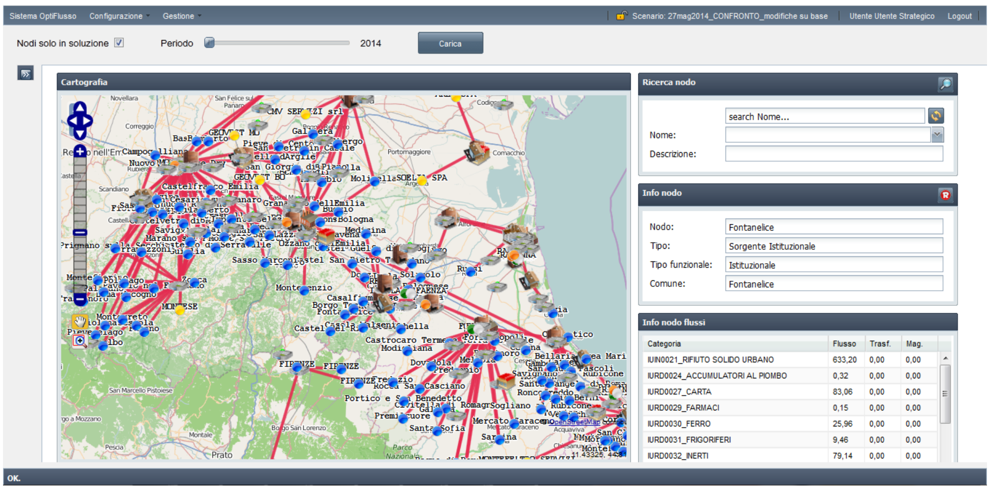
The waste flow management is operated at two levels: the strategic and the operational level. The strategic level relies on a DSS designed by Optit for Herambiente's central manager to plan the optimal waste flow allocation to the network facilities. The tool provides support to mid-term flow management decisions, however it yields solutions aggregated at weekly or monthly basis that are not directly implementable. Consequently, an operational level tool disaggregates the strategic level results until the level of individual shipments. The focus of this chapter is on the strategic level and concerns the solution of a specific implementation of the SWFA model presented in Section 4.4, populated with forecast waste production and actual data plant for a four-years planning horizon.

4.5.1 The Decision Support System Solution

OptitWasteFlow is a DSS developed by Optit that assists the waste flow manager in the formulation of the SWFA program (see Figure 4.6a), that models the system to optimize as accurately as possible. OptitWasteFlow is a web-based application with user-friendly Graphic User Interface (GUI) (see an example in Figure 4.6b). The user may generate and manage several alternate scenarios starting from a so-called "as-is" scenario which is used as a starting point and as comparison scenario. The as-is configuration is maintained up-to-date and preserved from temporary manipulation while branched scenarios permit the exploration of alternatives in a "what-if" fashion. For each scenario, the decision maker inputs data via the GUI. Data are translated by OptiWasteFlow in costs and constraints of the corresponding MILP formulation. Once the model is populated, the user launches the optimization on a remote server. For the practical case at hand, the results are available within a short computing time, fully compatible with the user needs. Results proposed in Section 4.6 have been obtained by using OptitWasteFlow to formulate and refine the scenario, as well as to analyze the outcomes.



(A)



(B)

FIGURE 4.6: OptiWasteFlow DSS: the Solution process (a) and an example of Graphical User Interface (b).

4.5.2 Solution Approach

4.5.2.1 Time Horizon and Time Granularity

For what concerns the strategic level, the SWFA model is solved with three different time horizons and relative time granularity. Time horizon and granularity have been defined with the aim to offer the decision maker the necessary level of detail.

The "Industrial Planning Level" operates with a time horizon of four years and a time granularity of one year. The Hera Group strategically defines every year a four-years planning in a rolling fashion. After filing an annual budget with a level of detail dependent on the necessities of the client, high-level strategic decisions about financial, marketing and managerial aspects are taken.

The "Budgeting Level" operates on a time horizon of one year and a time granularity of one month in which mid-term decision are involved. Generally, the main purpose of the budget level is related to the funds allocation among different areas and department of the group.

The "Operational Level" operates on a time horizon of one year and granularity of one week. This detailed level is not conditioned by group policies and it is run to take operational decisions.

In the following, we indicate both Industrial Planning and Budgeting levels as "strategic" levels. As previously mentioned, our case study is relative to an Industrial Planning scenario, nevertheless OptiWasteFlow is used by Herambiente to support decision at all three levels above.

4.5.2.2 Waste Commodities and Network Topology Definition

A one-to-one translation of each actual producer/facility into nodes in the graph and of waste types into waste commodities would lead to an extremely large optimization model. As a result, from a theoretical computational point of view, the overall resolution time would fall beyond practical solution possibility. In addition, from a more practical point of view, a greater level of detail would require a bigger effort for the population, validation and maintenance of the model. The strategic level operates the following simplifications to find an acceptable trade-off between model complexity and quality of the results.

Waste types aggregation The operational level aims at differentiating waste flows and acts at a level of detail similar to than the European Waste Catalogue (EWC) codification, originating more than 1000 different waste types. Instead, the strategic levels aggregate waste types in roughly 100 different typologies corresponding to commodities in the SWFA model.

Waste sources aggregation Herambiente manages more than 60,000 waste sources. The strategic levels aggregate waste generation sites in several hundreds of so-called Super-Sources (SS). Four different kinds of SS are considered:

- Single-Node Super-Sources (SNSS). This category includes nodes having a one-to-one correspondence with individual sources in the network. Generally, SNSS produce considerably large amount of waste. Hence, SNSS may be associated with a third-party waste operator that collects waste produced within one or more municipalities and then refer to Herambiente for waste treatment or disposal. SNSS also correspond to Herambiente facilities, to which private industries haul waste periodically. In this case, temporary stock is authorized and often economically convenient.

- Municipal Super-Source (MSS). The MW collected in a single municipality directly by Herambiente is aggregated into a single MSS, located in the geographical center of the municipality.
- Industrial Super-Sources (ISS). Small private industries are generally grouped in ISS. While public waste operators are allowed by the Italian legislation to collect the MW only over their controlled territory, private sector operators have access to the free market and to any operator located in the Italian territory. Herambiente attracts private industries from the whole country. Since most of the facilities are located in the centre-north area, a reasonable approximation is to aggregate ISS at provincial level. This is especially appropriate for industries located in the center-south or southern Italy.
- Extra-Territorial Super-Sources (ETSS). Sources considered out of the collection territory managed by Herambiente are grouped in ETSS.

Plants The plants made up the set of nodes $v \in V \setminus V_o$ with incoming and outgoing waste flows. The Herambiente network includes a variety of different plants, which may change year by year or depending on the considered scenario. Typically, an instance contains roughly fifty plants under direct supervision of Herambiente with the composition specified in Table 4.1.

Transshipment/Temporary Storage	18
WtE	7
Solid Waste Selection	12
Composting Facility	7
Biostabilization	4
Inerting	1

TABLE 4.1: Breakdown of Herambiente plants network

A larger number of facilities is not directly under Hera supervision. For those plants v it is hard for the decision maker to estimate the conversion factor $b_{vww'}$ for their realistic description in the model. To overcome such an issue, such facilities are associated with two types of nodes in the network: either a destination or a source. The total number of network nodes, including facilities managed by third-party, disposal plants with no waste outputs, and facilities that in general do not produce an output as a function of the input, is generally around the 400 units.

Finally, a fictitious destination is created to collect all the waste flows that cannot be treated or disposed in the network. From a mathematical point of view, this auxiliary destination can be viewed as a set of high-cost slack variables. In a practically feasible solution, in fact, the amount of waste flow sent to the fictitious destination should be equal to zero.

Distance Matrix A static distance matrix defines the distances between every pair of nodes present in the network considered at the strategic level. The distance matrix maps the distances between more than 800 nodes, with nodes corresponding to facilities located in the center-north part of Italy (see 4.7a) and sources distributed all over the Italian territory (see 4.7b).

(A) Facilities locations (B) Sources locations

Figure 4.7: Location of facilities (left) and sources (right) in the Herambiente case study

4.5.2.3 Costs and Revenues

The costs associated with the waste flow management are transformation costs (including disposal at landfills) and logistic costs. Regarding the logistic costs, the case study considers only costs paid by the waste manager. When a third-party producer hauls waste directly to a facility, logistic costs are generally taken by the producer and not considered in the model. Logistic costs from producer to facility are instead considered when Herambiente operates also as a carrier. This can happen both for Industrial and Municipal waste.

There are two main types of income: incomes deriving from waste disposal and revenues from the sale of products derived from waste, including energy. The energy production revenue is modeled as a particular case of the disposal revenue. Usually the incomes depend on the type of waste commodity but they can also be related with the facility in which they are disposed or produced.

In the considered case study, costs do not depend on time and volume, and economies of scale presented in Section 4.4.5 are not applied.

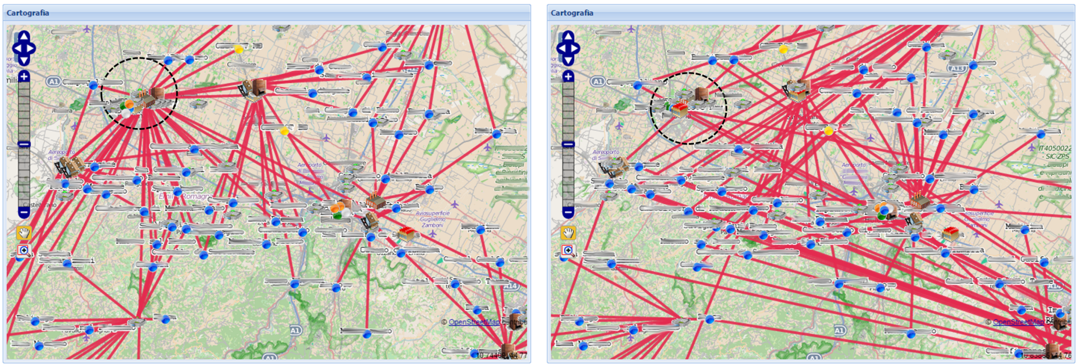
4.5.2.4 Operations Modeling and Constraints

The business cases addressed in this section corresponds to a particular instance of the model (4.1)-(4.18), together with a simplified form of the additional features introduced in Section 4.4.6. The DSS contains a number of heuristics for considering the additional features described: for confidentiality issues, the algorithms implemented in the software are not fully reported in the chapter. As expressed by the constraints (4.4)-(4.11), plants are often characterized by several flow limitations, both in absolute and relative terms. Facility operativeness constraints are also defined in order to take into account maintenance operations.

4.6 Results

In the considered case study, the resolution of SWFA via the commercial software Cplex required limited computational effort. In particular, the solver is typically able to close the gap already at the root node, and the MIP resolution required less than a minute. The overall time-to-solution, including the pre-processing and post-processing done by the OptiWasteFlow application, is generally smaller than ten minutes. Considering the strategic nature of the process, a precise measurement of the resolution times is not reported in this chapter.

The pre-existing yearly budget process, created "manually" with support of office automation tools, typically required two Full Time Equivalent (FTE) resources for about two weeks. Such process time can not be directly compared with the computational time expressed above, as the direct application of a solution proposed by the OptiWasteFlow is not a practical option. In fact, not all the economical and environmental aspects of the waste management system can be easily modeled as constraints or costs in the SWFA . The decision maker generally sets up the model by realistically replicating the system to optimize. The solution obtained from this initial model is analyzed to evaluate the satisfaction of additional qualitative requirements. The model can then be adjusted, thus leading to a set of alternative scenarios, each with its optimal solutions. In this phase a what-if analysis is performed where the decision maker adds some (fictitious) costs or some additional constraints to produce solutions that can be compared with the original one. For example, in Figure 4.8, the consequences of forbidding a WtE to receive MSW are displayed. Figure 4.8b shows that additional facilities are required for disposing the waste, leading to waste allocation on longer transportation links with respect to the original solution showed in Figure 4.8a. The possibility of quickly evaluating alternative scenarios in what-if analysis is clearly a relevant feature of the model both at strategic level and at an operational one; this allows to readily react to unforeseen event and restore practical and feasible solutions.



(A) Without limitations on flow (B) With limitations on flow

FIGURE 4.8: Changes on flow allocation as a consequence of introduced limitation of waste acceptance on a WtE facility

With the integration of Operations Research methods in OptiWasteFlow, only the set-up operation remains FTE-intensive, requiring one FTE, while the main computational effort is required by the MIP solver. In fact, it is possible to produce several alternative solutions with the higher level of detail within few hours. Furthermore, the network is unlikely to change radically one year from the following, so the set-up phase is only needed in the four-year planning and most data are inherited by the other levels.

In Section 4.6.1 we present the results obtained by using OptiWasteFlow for the design a four-years planning. The results are compared with the as-is scenario currently implemented, representing the on-going flows operated by Herambiente. The as-is scenario is the result of progressive adjustments and fine tuning of pre-existing situations built through several years of service. For confidentiality reasons the actual data of the scenarios have been altered but still capture the nature of the system operated by Herambiente.

4.6.1 Comparative Results for the Case Study

We call "*optimized*" the plan designed by the decision maker by using OptiWasteFlow with the support of Optit. The plan that we call "*as-is*" corresponds to a solution of the SWFA that replicates the on-going solution. The optimized solutions have similar restrictions with respect to the as-is and use the same costs for logistic operations and treatment at the facilities. The amount of flow incoming from the sources is the same for the two compared solutions.

The main purpose of the comparison is to show which decisions taken by the mathematical model modify the as-is solution and which are the effects on the main key performance indicators (KPI). Any economical, social, or ethical evaluation of the policies applied by the decision maker goes beyond the purpose of this chapter. For this reason (and also to avoid the disclosure of potentially confidential information) we tend to avoid the representation of results in absolute terms. In most of the cases, differences in percentage from the values measured in the as-is solution and the ones obtained in the optimized solutions are presented and discussed. In particular, if not differently specified, $\Delta\%$ in the tables expresses the difference between optimized and as-is solution expressed as a percentage of the corresponding value in the as-is solution. A negative sign stands for a decrease in an indicator in the optimized solution.

4.6.1.1 Economic Key Performance Indicators

Tables 4.2, 4.3, and 4.4 summarize the variations for the main KPI considered for the case study. Disposal costs arise for $r^t_{vw}, p^t_{vw} < 0$ in the model and can be payed when waste is sent to the first treatment facility (if any) or when the waste reaches a final "destination" outside the system (e.g., recycle market). The main relative reduction for the disposal cost has been obtained for transshipment facilities with a reduction of 29.02%. When costs increase for some of the facility or destination nodes, as for recycle, this often means a more intense usage. The revenues deriving from waste disposal arise are associated to $r^t_{vw}, p^t_{vw} > 0$ in the model. They are mainly obtained from the municipalities as a result of the management of: municipal solid waste (covering roughly 95% of the total revenues in both the as-is and optimized solutions); waste deriving from the activity of street cleaning (3.9%) and cemetery waste (0.1%). The revenues are earned when waste leaves the source: if waste departs from source v to enter in facility w, then the corresponding revenue is represented by $r^t_{vw} > 0$. Note that the revenue $r^t_{vw} > 0$ does not depend only the source, but potentially depends also on the destination w. Tables 4.4 shows that the optimized solution makes less use of the transshipment nodes, sending more often the waste directly to the successive facility and preferring the WtE among the other destinations.

Performance indicator	$\Delta\%$
Disposal costs	−2.03%
Treatment revenues	−0.02%
Sub-products revenues	−1.83%
EBIT	+6.01%
Estimated Logistic costs	−43.97%

TABLE 4.2: Comparison of main logistic and economic KPI: negative values means a reduction in the optimized solution with respect to the as-is solution.

Revenues for sub-products derives from the selling of electric energy (for the 97.9% in both solutions) and composts (for the remaining 2.1%). The optimized solution faces

Facilities	Δ%
Selection	+15.05%
WtE	−3.44%
Landfill	−3.28%
PBT	+18.29%
Composting	−5.32%
Transshipment	−29.02%
Biostabilization	−3.01%
Inerting	≈ 0%

Destinations	Δ%
Filling material	−20.80%
Fertilizer	−6.70%
Recycle	+23.64%

Table 4.3: Percentage variation of total disposal cost in detail for various facilities and final destinations.

Municial solid waste

Destination	Δ%
Transshipment	−15.62%
WtE	+15.62%
Waste selection	−4.96%
Landfill	+23.45%

Street cleaning waste

Destination	Δ%
Transshipment	−66.05%
WtE	+295.37%
Waste selection	≈ 0%
Landfill	−79.96%

Cemeterial waste

Destination	Δ%
Transshipment	≈ 0%
WtE	−2.12%
Waste selection	≈ 0%
Landfill	+8.16%

Table 4.4: Percentage variation of treatment revenues in detail for some aggregated waste types and destinations.

an increase of the 12.7% on the revenues from composting, which is counterbalanced by a reduction on the revenue from electric energy of the −2.14%: this leads to lower overall revenues. The acronym EBIT, in Table 4.2 stands for Earnings Before Interest and Taxes, which are the quantities to be maximized in the SWFA model.

The objective function also includes fictitious costs associated with slack variables for constraints satisfaction. Slack variables are necessary in order to build a feasible solution for the MILP model in the setup phase, in which the introduction of a large set of constraints often causes infeasibilities. The introduction of slacks helps the practitioners to understand "how far" they are from a feasible solution and which are the conflicting constraints. Moreover, the decision maker tolerates the presence of a marginal percentage of waste not routed in the solution proposed by the solver. To this end, a penalty cost is paid in the optimal solution, as an estimate of the (unknown) disposal cost for the corresponding waste. Typically, the disposal of this waste is assigned with public tenders. In the optimized solution, 0.75% of the flow remains not allocated, while in the as-is solution the entire waste flow is treated.

The estimated logistic costs are not included in the evaluation of the waste disposal net profit (EBIT) in the table. This is because such costs estimation does not directly measure an actual expenditure for the decision maker. In fact, the decision maker stipulates a variety of periodic contracts with several carriers. Economies of scale are often considered in the contract definition, therefore the nonlinear cost definition presented in Section 4.4.5 can be useful to give a more accurate definition of transshipment costs. Even if a reduction of the estimated logistic costs does not translate in an immediate cost reduction for the decision maker, this indicator is still interesting. A reduced cost for the carrier may lead to more convenient contractual terms also for the waste operator when such contracts are periodically renewed with the carriers. Composing roughly the 12% of the overall costs of waste treatment in the as-is solution, the reduction of logistic costs have a significant impact on the cost of the service.

4.6.1.2 Waste Flow Allocation to Facilities

Table 4.5 summarizes the differences between optimized solution and as-is solution in flow allocation. The percentage variation in the amount of flow (in tons) sent to the main of facilities is measured. The amount of flow produced by the sources is the same in the two solutions. The variation in the total amount of flow traveling over the network is due to a different usage of plants, which leads to different conversion terms $b_{vww'}$. As mentioned before, a small part of the flow, equal to 0.75%, is not processed in the optimized solution. In the post-optimization phase, such flows are allocated similarly as prescribed by the as-is solution.

The optimized solution decreases significantly the amount of waste routed to transshipment points, confirming the results of Table 4.4. In general, such facilities are meant for the wastes temporary stock and to consolidate trucks loads. A manual planning tends to use transshipments because they simplify the system with the introducing of some buffers in the transportation. The common sense decision-making often includes the route of flow from many small sources into a transshipment node. Aggregated flow is then routed from the transshipment node to the facilities. In some cases, this intuitive good practice may hide inefficiencies, which are instead avoided in the optimization-based approach.

Facilities	Δ%
PBT	−3.24%
WtE	−1.95%
Solid Waste Selection	−4.66%
Transshipment	−18.56%
Composting	−0.58%
Biostabilization	−0.08%
Inerting	−56.82%

Destinations	Δ%
Fertilizer	−4.17%
Recycle	+13.38%
Landfill	+4.61%
Filling Materials	+6.46%

Table 4.5: Percentage variation in flow allocation among facilities and final destinations.

4.6.2 Landfill Disposal Limitation Scenario

Generally, the disposal of waste in landfill is discouraged. This can be motivated by the difficulty of estimating operational costs because, for instance, of the peculiarity of leachate production that can necessitate decades. Furthermore, a variety of environmental reasons and regulations makes landfill disposal a non attractive choice. We here present the effects that limitations on the amount of waste disposable in landfill may have on operational costs. The proposed solution has been obtained by adding to the SWFA a set of constraints that forbids an increase of flow routed to the landfills with respect to the as-is solution.

4.6.2.1 Economic Key Performance Indicators with Landfill Disposal Limitations

As reported in Table 4.6, the disposal cost increases when flow limitations are introduced for landfills. Such cost is partially covered by an increased revenue from selling sub-products (typically electric energy produced in WtEs). The EBIT is lower than the one in the optimized solution, but still higher than the as-is situation. The estimated logistic costs increase with respect to the optimized solution without landfill disposal limitation constraints. This is due to the presence of more complex routes for some waste that involve more treatments before reaching their final destination. Anyway, the logistic cost remains considerably smaller than the one in the as-is situation, because the limitations for the disposal in landfills affect a limited amount of waste. This happens because such kind of waste disposal was a “back-up” option already in the optimized solution.

Performance indicator	$\Delta\%$
Disposal costs	-0.72%
Treatment revenues	-0.04%
Sub-products revenues	-0.86%
EBIT	$+1.41\%$
Estimated Logistic costs	-41.56%

Table 4.6: Percentage variation of total disposal cost in detail for various facilities and final destinations. Scenario with landfill disposal limitations.

Facilities	$\Delta\%$
Selection	$+15.34\%$
WtE	-2.85%
Landfill	-9.78%
PBT	$+13.68\%$
Composting	-2.01%
Transshipment	-28.49%
Biostabilization	-1.74%
Inerting	$\approx 0\%$

Destinations	$\Delta\%$
Filling material	-12.85%
Fertilizer	-5.62%
Recycle	23.64%

Table 4.7: Percentage variation of total disposal cost in detail for various facilities and final destinations. Scenario with landfill disposal limitations.

4.6.2.2 Waste Flow Allocation to Facilities with Landfill Disposal Limitations

In Table 4.9 the allocation of the waste is summarized for the scenario with landfill disposal limitations. The amount of flow sent to PBT and waste selection facilities increases as preliminary phase of treatment. Being a disposal alternative to landfills, the WtE is the destination site mainly affected by the introduced limitations. The amount of recycled waste does not vary significantly because the recycle option for "noble materials", such as glass or wood, was generally already chosen in the original optimized solution. A slight increase of fertilizer and filling material is observed, indicating that a minor part of the waste used to produce them were disposed in landfills in the original optimized solution.

4.7 Conclusions and Future Works

In the chapter, we proposed mathematical models for addressing the waste flow allocation problem in a medium-long term horizon of planning. We showed that the MILP formulation is used in a Decision Support System developed by the consulting

Municial solid waste

Destination	Δ%
Transshipment	-14.63%
WtE	$+15.95\%$
Waste selection	-4.84%
Landfill	$+19.10\%$

Street cleaning waste

Destination	Δ%
Transshipment	-56.10%
WtE	$+261.06\%$
Waste selection	$\approx 0\%$
Landfill	-78.24%

Cemeterial waste

Destination	Δ%
Transshipment	$\approx 0\%$
WtE	-2.98%
Waste selection	$\approx 0\%$
Landfill	$+11.47\%$

Table 4.8: Percentage variation of treatment revenues in detail for some aggregated waste types and destinations. Scenario with landfill disposal limitations.

Facilities	Δ%
PBT	-3.09%
WtE	-1.24%
Solid Waste Selection	-4.21%
Transshipment	-19.15%
Composting	-0.57%
Biostabilization	$+0.94\%$
Inerting	-56.82%

Destinations	Δ%
Fertilizer	-3.59%
Recycle	$+13.38\%$
Landfill	–
Filling Materials	$+19.62\%$

Table 4.9: Percentage variation in flow allocation among facilities and final destinations. Scenario with landfill disposal limitations.

company Optit for the waste manager Herambiente SPA. The proposed tool is able to give solutions that lower the operators cost and enables a fast evaluation of solution alternatives in response to modifications of network features (e.g., limiting waste disposal in landfills).

A relevant future research direction lies in the explicit consideration of the uncertainty of the waste generation rates. Such variability is expected to have a significant impact especially in a strategic level of planning. The MILP model should be then reformulated in a two-stage multiperiod stochastic or multistage stochastic framework.

Chapter 5

A Solid Waste Management Problem with Stochastic Parameters at a Tactical Planning Level

5.1 Introduction

In this chapter we present a planning problem in Solid Waste Management (SWM) with stochastic parameters at tactical level with yearly time horizon. Waste management problems were introduced in Chapter 4 in their deterministic version, namely with the assumption that all problem data are known at the moment of planning. In the real setting, such planning problems are naturally affected by uncertainty. The primary stochastic component is the amount of waste generated in towns; in addition, uncertainty can affect transportation costs, as well as processing costs and waste transformation coefficients in waste treatment plants, whenever the precise composition of the incoming flow is crucial for determining the output of the treatment.

Two different two-stage multiperiod stochastic mixed-integer formulations for the SWM problem are presented in this chapter. They differ for the possibility of incurring in higher transport and processing costs when the amount of waste generated turns out to be greater than what expected.

The remainder of the chapter is organized as follows. Section 5.2 provides a non-exhaustive overview on the contributions in literature on SWM problems with uncertain parameters; both strategic and tactical planning problems are mentioned. In Section 5.3, the specific SWM planning problem is described, while Section 5.4 presents the stochastic models we formulated. In Section 5.5, the scenarios generated from the available data are described. In Section 5.6, some observations on the preliminary numerical testing of the stochastic models on a realistic instance are expressed. Finally, some conclusions are drawn and future research directions are indicated in Section 5.7.

5.2 Literature Review

For a thorough discussion on mathematical formulations and solution approaches for waste management problems under uncertainty, the reader is referred to the recent survey of Sun et al. [233]. A common claim in papers on this topic is that the quality and quantity of available data and information for many relevant uncertain parameters of the planning problem are not sufficient for estimating their probability distributions. Especially in large-scale problems, obtaining an accurate description of the real case-study is particularly laborious (Maqsood and Huang [187]).

5.2.1 Strategic Planning

One of the first studies on the SWM at a strategic level of planning is in Maqsood and Huang [187]. The authors consider a hypothetical network composed by three cities, one incinerator and a landfill in which environmental policies in terms of allowable waste-loading levels must be set. In the considered planning horizon of 15 years, there is no possibility of opening new facilities or closing the existing ones. Every five years, the decisions on the amount of waste to be sent to the incinerator or landfill must be taken. The uncertainty affects the waste-generation amounts, the waste transportation costs, the operation cost of facilities and the revenues obtained from the incinerator activities; cost and revenues are subject to modifications when associated with excess waste flow and residues from incinerators. A probability is associated with each level of intervals of waste generated in each city and the aim of the optimization is to minimize the expected value of net system cost in the region while respecting the facilities capacity and waste-disposal demand constraints. A two-stage interval-stochastic programming model in which uncertain data are expressed as interval parameters is developed. The interactive solution algorithm (Huang et al. [142]) consists in considering two deterministic submodels, producing lower and upper bounds for the objective-function value. The computational testing shows that the stable intervals solutions contain the trade-offs between the waste-management cost and the system-failure risk related to allowable waste-loading levels. The interval solutions can be used for generating decision alternatives by evaluating the impact of environmental, economic, and system-reliability factors.

Considering the same planning horizon in a network of similar size, Li and Huang [175] propose an interval minimax regret programming formulation. Uncertain data are expressed as interval random variables, for which no probability distributions is available. The objective is to minimize the maximum regret levels among all considered scenarios, where the regret is defined as the difference between the expected cost of the chosen waste-flow allocation plan and the actual cost paid after the realization of the uncertain parameters.

In Li and Huang [173], an inexact two-stage mixed-integer linear programming method is proposed for a SWM problem in the city of Regina, Canada, with a planning horizon of 25 years and 5 periods. The main difference in the modeling part with respect to [187] and [175] is the introduction of discrete variables for representing the expansion options for waste management facilities in different periods. This possibility is particularly relevant in a long-term horizon since the amount of waste generated is expected to increase in the future.

Uncertainty in both left and right hand side parameters of probabilistic constraints is tackled by means of intervals and probability distributions by Guo et al. [127], who combine stochastic programming, integer programming, and interval semi-infinite programming for expressing a strategic problem. In such a paradigm, waste-generation rates and capacity expansion options are expressed as functional intervals dependent by a time variable.

In Li et al. [174], the concept of waste-fluctuation rate is considered in a two-stage fuzzy robust integer programming formulation. The authors observe that two phenomena cause that not all the waste generated in sources is delivered to treatment and disposal facilities: mass loss may occur during the collection and transportation of the generated waste; in a situation of traffic congestion, delays in transportation times can prevent a complete waste treatment within the current period and hence produce raised flow in the following period. Such variability is considered in fuzzy constraints regarding the waste capacity requirements in waste treatment facilities.

5.2.2 Tactical Planning

Despite the practical importance of the problem, the contributions on the tactical level of planning are quite limited.

A weekly waste flow allocation problem is considered in Huang et al. [143]. Since uncertain parameters vary in fluctuation intervals without indications of probability distribution, the unknown data are expressed as gray numbers (see, e.g., Liu and Lin [177]). The authors develop a gray linear programming model for addressing the regional municipality of Hamilton-Wentworth in Ontario. The network is made up by 6 cities divided into 17 waste-generation districts and 8 waste-treatment and disposal facilities. The optimal waste flow in the network arcs is determined by respecting specific requirements such as facility capacity constraints and bounds on the operating level in treatment plants. Three scenarios dependent on different levels of operativeness of the waste-to-energy facility are considered. The stable interval solutions show that minor changes to the existing waste-flow allocation plan can lead to interesting cost savings.

Yeomans et al. [259] improve the results of [143] on the same case study by proposing an Evolutionary Simulation-Optimization (ESO) procedure. The uncertain data are provided with uniform distribution in the fluctuation intervals. In each iteration of the algorithm, every solution candidate is evaluated with a simulation based on performance measures. The results of the simulation phase are then compared and the genetic step let the population evolve. The ESO algorithm considers the gray linear programming solution of [143] as initial solution, hence it yields a final improved solutions.

An additional ESO is proposed by Yeomans [260] for speeding up the solution approach of [259]. Infeasible solutions are considered in the evolutionary phase by means of penalty terms in the objective function.

To the best of our knowledge, the work of the chapter is the first to propose two-stage multi-period stochastic mixed-integer programming models for waste flow allocation problems at a tactical level of planning. We also explicitly model the modifications that the incoming waste experiences in waste treatment facilities by introducing transformation coefficients for some waste commodities.

5.3 The Planning Problem

Given a set of waste generation sources and a set of potential facilities for waste treatment, separation and disposal, the SWM planning decisions amount to determine which facilities should be used, and how waste should be routed, processed and disposed in each period in order to minimize the total cost, net of any revenue for reclaimed material and generated energy.

We assume that the network structure is fixed, that is we cannot build new facilities and landfills or closing them permanently. This assumption is appropriate in the medium-short level of planning. The waste manager is aware of compulsory deactivation terms for facilities due to maintenance operations; the decision regards the selection of the beginning period of deactivation.

The total planning cost is made up by a transportation cost per unit of waste associated with each network arc, a fixed cost for each operating facility in each period, a processing cost per unit of waste at each facility in each period and a unit revenue for the produced energy.

As already mentioned, an SWM problem involves parameters that are not known at the moment in which the planning has to be made. In this study, we focus only on the stochasticity in waste generation quantities, since they can be considered the major source of uncertainty in waste management problems.

5.4 Stochastic Models

We consider two two-stage multiperiod formulations that reflect the way in which the uncertainty in waste generation is expressed and revealed during the planning horizon. In the two-stage multiperiod paradigm, the actual realization of the waste generation values in each period (say, a month) of the planning horizon becomes known as soon as the facility activation decisions for the first period are taken. The set of waste manager decisions is dependent on his/her possibility to take operational recourse action or not. The absence of operational recourse actions is typical in a budget planning problem in which the main concern of waste manager is the decision on the facilities activation in order to minimize expected future costs. A different problem formulation can arise if operational corrective actions can be taken, such as shipping the unforeseen waste outside the network with additional vehicles: such decisions will be subject to "high" recourse costs.

The notation used in the models formulation is now presented.

Sets

$G = (V, A)$	directed graph representing the waste management network;
V	set of network nodes, $V = (V_O \cup V_S \cup V_P \cup V_L)$ (see Figure 5.1);
A	set of feasible waste shipments between network sites;
V_O	set of waste generation sources;
V_S	set of separation and transfer stations (i.e., plants in which waste is temporarily disposed and then loaded into larger vehicles);
V_P	set of processing facilities (e.g., incinerators, waste-to-energy plants);
V_L	set of landfills, disposal facilities and markets for recycled products and energy;
W	set of waste commodities.

Parameters

T	number of periods of the planning horizon;
q^t_{jw}	capacity of facility $j \in V_S \cup V_P \cup V_L$ for commodity $w \in W$ in period $t = 1, \ldots, T$;
$b_{jww'}$	transformation coefficient per unit weight (or volume) of the waste commodity $w \in W$ into the waste commodity $w' \in W$ at facility $j \in V_S \cup V_P$;
m^t_{jw}	minimum threshold of incoming waste commodity w in period $t = 1, \ldots, T$ in operating plant $j \in V_S \cup V_P$;
a_j	overall capacity of facility $j \in V_L$;
D^t_j	duration of deactivation term of facility j starting in period t;
τ^t_j	number of periods of temporary deactivation of facility j within the time horizon of planning (i.e., $\tau^t_j = \min\{D^t_j - 1, T - t\}$).

Cost parameters

c^w_{ij}	unit waste transportation cost associated with arc $(i, j) \in A$ and $w \in W$;
f^t_j	fixed cost for the operativeness of facility $j \in V_S \cup V_P$ in period $t = 1, \ldots, T$;
p^t_{jw}	unit processing cost of waste commodity $w \in W$ in facility $j \in V_S \cup V_P \cup V_L$ in period $t = 1, \ldots, T$;
r^t_{jw}	revenue obtained from a unit of waste commodity $w \in W$ entered in plant $j \in V_P$ in period $t = 1, \ldots, T$.

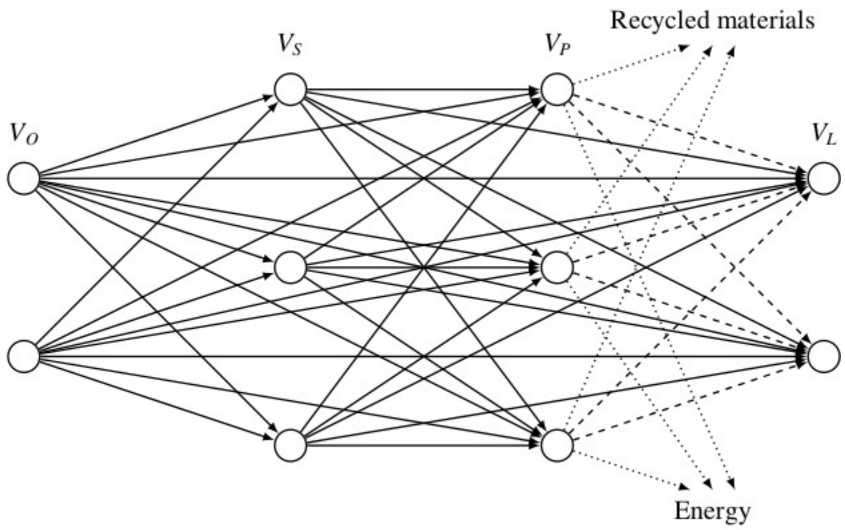


FIGURE 5.1: The SWM network

The SWM problem is modeled as a multicommodity flow problem (Shevchik [226]) with multiple sources and sinks.

5.4.1 Two-Stage Multiperiod Formulation without Operational Actions

We consider a first two-stage multiperiod formulation where the *nonanticipative* decision variables represent the facility activation, while the *recourse action* is the waste flow allocation in each period of the planning horizon. The uncertain amount of waste commodity $w \in W$ generated in source $i \in V_O$ in period t according to scenario s is denoted by $g_{iw}^{t,s}$. The set of scenarios is represented by $\mathcal{S}$ of cardinality $|\mathcal{S}|$; each scenario has a probability π_s, $s \in \mathcal{S}$.
The decision variables are:

- y_j^t binary variables assuming the value 1 if facility $j \in V_S \cup V_P$ is operating in period $t = 1, \dots, T$ or 0 otherwise;
- ρ_j^t binary variables assuming value 1 if facility $j \in V_S \cup V_P$ is starting its deactivation term in period $t = 1, \dots, T$ or 0 otherwise;
- $x_{ijw}^{t,s}$ actual waste flow of commodity $w \in W$ shipped in arc $(i,j) \in A$ in period $t-1$ ($t = 2, \dots, T+1$) in scenario $s \in \mathcal{S}$.

Note that binary variables y_j^t and ρ_j^t do not depend on the particular scenario s. Equivalently, we can impose the *nonanticipativity* constraints.:

$$\begin{aligned} y_j^{t,s_1} &= y_j^{t,s_2} \quad \forall\, j \in V_S \cup V_P, t = 1, \dots, T, s_1, s_2 \in \mathcal{S}, \\ \rho_j^{t,s_1} &= \rho_j^{t,s_2} \quad \forall\, j \in V_S \cup V_P, t = 1, \dots, T, s_1, s_2 \in \mathcal{S}. \end{aligned}$$

A mathematical formulation of such a problem is the following two-stage multiperiod mixed-integer stochastic programming model. In the remainder of the chapter, the model is referred to as Model (M1).

$$\begin{aligned} \text{(M1)}: \min \quad & \sum_{t=1}^{T} \sum_{j \in V_S \cup V_P} f_j^t y_j^t + \sum_{s=1}^{|\mathcal{S}|} \pi_s \Big(\sum_{t=2}^{T+1} \sum_{w \in W} \sum_{(i,j) \in A} c_{ij} x_{ijw}^{t,s} + \\ & + \sum_{t=2}^{T+1} \sum_{w \in W} \sum_{j \in V_S} p_{jw}^{t-1} \sum_{i \in V_O} x_{ijw}^{t,s} + \sum_{t=2}^{T+1} \sum_{w \in W} \sum_{j \in V_P} p_{jw}^{t-1} \sum_{i \in V_O \cup V_S \cup V_P} x_{ijw}^{t,s} + \\ & + \sum_{t=2}^{T+1} \sum_{w \in W} \sum_{j \in V_L} p_{jw}^{t-1} \sum_{i \in V_O \cup V_S \cup V_P} x_{ijw}^{t,s} + \sum_{t=2}^{T+1} \sum_{w \in W} \sum_{j \in \cup V_P} r_{jw}^{t-1} \sum_{i \in V_O \cup V_S \cup V_P} x_{ijw}^{t,s} \Big) \end{aligned} \tag{5.1}$$

$$\text{s.t.} \sum_{j \in V_S \cup V_P \cup V_L} x_{ijw}^{t+1,s} = g_{iw}^{t,s} \qquad \forall\, i \in V_O,\ w \in W, \quad t = 1, \dots, T,\ s \in \mathcal{S}, \tag{5.2}$$

$$\sum_{w \in W} b_{jww'} \sum_{i \in V:(i,j) \in A} x_{ijw}^{t,s} = \sum_{i \in V:(j,i) \in A} x_{jiw'}^{t,s} \qquad \forall\, j \in V_S \cup V_P, \forall\, w' \in W, \quad t = 2, \ldots, T+1,\ s \in \mathcal{S}, \tag{5.3}$$

$$\sum_{i \in V:(i,j) \in A} x_{ijw}^{t+1,s} \leq q_{jw}^t y_j^t \qquad \forall\, j \in V_S \cup V_P,\ w \in W, \quad t = 1, \ldots, T,\ s \in \mathcal{S}, \tag{5.4}$$

$$\sum_{i \in V:(i,j) \in A} x_{ijw}^{t+1,s} \geq mf_{jw}^t y_j^t \qquad \forall\, j \in V_S \cup V_P,\ w \in W, \quad t = 1, \ldots, T,\ s \in \mathcal{S}, \tag{5.5}$$

$$\sum_{t=2}^{T+1} \sum_{w \in W} \sum_{i \in V:(i,j) \in A} x_{ijw}^{t,s} \leq a_j \qquad \forall\, j \in V_L,\ s \in \mathcal{S}, \tag{5.6}$$

$$\sum_{i=0}^{\tau_j^t} y_j^{t+i} \leq (\tau_j^t + 1)(1 - \rho_j^t) \qquad \forall j \in V_P \cup V_S, \quad t = 1, \ldots, T, \tag{5.7}$$

$$\sum_{t=1}^{T} \rho_j^t \geq 1 \qquad \forall\, j \in V_P \cup V_S, \tag{5.8}$$

$$y_j^t \in \{0,1\} \qquad \forall\, j \in V_S \cup V_P, \quad t = 1, \ldots T, \tag{5.9}$$

$$\rho_j^t \in \{0,1\} \qquad \forall\, j \in V_P \cup V_S, \quad t = 1, \ldots, T, \tag{5.10}$$

$$x_{ijw}^{t,s} \geq 0 \qquad \forall\, w \in W, (i,j) \in A, \quad t = 2, \ldots T+1,\ s \in \mathcal{S}. \tag{5.11}$$

The objective function (5.1) is composed by the operational costs for active facilities (first term of (5.1)) and by recourse terms. The recourse costs are given by the waste transportation costs, the processing costs in transfer stations, in processing plants and in landfills and a revenue in processing facilities.

Constraints (5.2) ensure that the stochastic waste generated in each source is collected. Equations (5.3) impose the reduced flow balance in each transfer or processing facility. Constraints (5.4) represent capacity limitations for active plants, while inequalities (5.5) model the requirement for operating facilities to receive a minimum amount of incoming waste flow. Constraints (5.6) are capacity restrictions within the entire planning horizon for disposal sites. Constraints (5.7) and (5.8) manages facility deactivation terms. In particular, constraints (5.7) assure that, after starting the deactivation term in period t, the facility j is not operational for D_j^t consecutive periods. If the non-operativeness term exceeds the end of the planning horizon, such situation will be considered in the following planning period. Constraints (5.8) impose to begin a facility deactivation term within the planning horizon.

Finally, constraints (5.9)-(5.10)-(5.11) define the decision variables of the problem. In Figure 5.2, the scenario tree which describes the situation represented by the model (5.1)-(5.11) over 12 months with $|\mathcal{S}| = 3$ is presented.

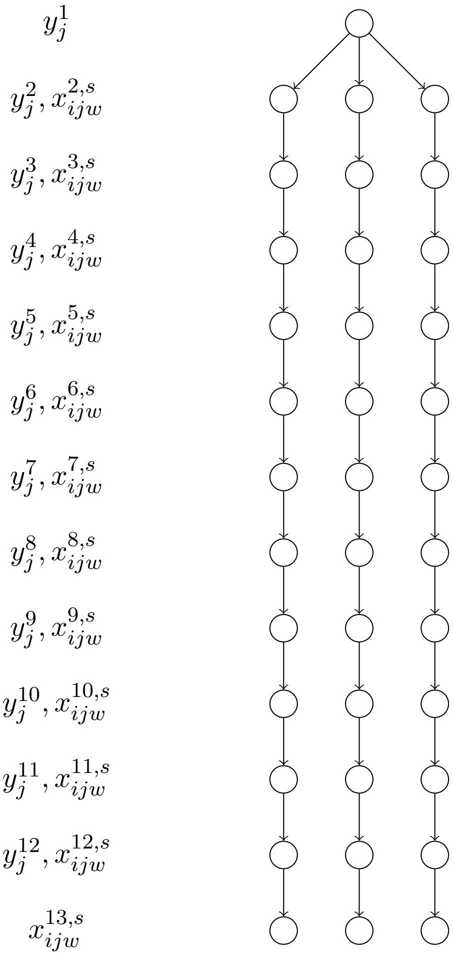


FIGURE 5.2: Two-stage multiperiod scenario tree related to formulation (5.1)-(5.11) with $|\mathcal{S}| = 3$ and $T = 12$

5.4.2 Two-Stage Multiperiod Formulation with Operational Actions

In the following two-stage multiperiod formulation, operational corrective decisions are taken into account. The resulting set of decision variables is given by:

y_j^t binary variables assuming the value 1 if facility $j \in V_S \cup V_P$ is operating in period $t = 1, \dots, T$ or 0 otherwise;

ρ_j^t binary variables assuming value 1 if facility $j \in V_S \cup V_P$ is starting its deactivation term in period $t = 1, \dots, T$ or 0 otherwise;

x_{ijw}^t planned waste flow of commodity $w \in W$ shipped in arc $(i, j) \in A$ in period $t = 1, \dots, T$;

$\xi_{iw}^{t,s}$ excess waste of commodity $w \in W$ present in source $i \in V_O$ in period $t = 2, \dots, T+1$ in scenario $s \in \mathcal{S}$.

Observe that, in addition to binary variables y_j^t, ρ_j^t , also the decision x_{ijw}^t in the planned waste flow is nonanticipative in this formulation. The corrective waste flow $\xi_{iw}^{t,s}$ is non-negative if in period $t-1$ the waste generation has turned out to be lower than expected. Such waste can be treated in several ways in practical applications. We assume that the unexpected waste is collected in waste generation sources incurring in $C_{i,w}^t$ costs, higher than network transportation costs, and shipped outside the network. In this situation, the excess flow affects only the waste collection constraints in sources sites. Another possibility of treatment of the unforeseen waste could be that of routing it inside the network, at the price of additional transportation costs, because of possible vehicle overloading or usage of extra vehicles. All the other deterministic and stochastic parameters are the same as those presented in Section 5.4.

A mathematical formulation of the two-stage multiperiod mixed-integer problem with operational actions is given in the following model, called Model (M2):

$$
\begin{aligned}
(\text{M2}): \min \quad & \sum_{t=1}^{T} \sum_{j \in V_S \cup V_P} f_j^t y_j^t + \sum_{t=1}^{T} \sum_{w \in W} \sum_{(i,j) \in A} c_{ij} x_{ijw}^t + \\
& + \sum_{t=1}^{T} \sum_{w \in W} \sum_{j \in V_S} p_{jw}^t \sum_{i \in V_O} x_{ijw}^t + \sum_{t=1}^{T} \sum_{w \in W} \sum_{j \in V_P} p_{jw}^t \sum_{i \in V_O \cup V_S \cup V_P} x_{ijw}^t + \\
& + \sum_{t=1}^{T} \sum_{w \in W} \sum_{j \in V_L} p_{jw}^t \sum_{i \in V_O \cup V_S \cup V_P} x_{ijw}^t - \\
& - \sum_{t=1}^{T} \sum_{w \in W} \sum_{j \in \cup V_P} r_{jw}^t \sum_{i \in V_O \cup V_S \cup V_P} x_{ijw}^t + \\
& + \sum_{s=1}^{\mathcal{S}} \pi_s \Big(\sum_{t=2}^{T+1} \sum_{i \in V_O} \sum_{w \in W} C_{i,w}^t \xi_{iw}^{t,s} \Big)
\end{aligned}
\tag{5.12}
$$

$$
\text{s.t.} \sum_{j \in V_S \cup V_P \cup V_L} x_{ijw}^{t+1,s} + \xi_{iw}^{t+1,s} = g_{iw}^{t,s} \qquad \forall\, i \in V_O,\ w \in W, \quad t = 1, \dots, T,\ s \in \mathcal{S}, \tag{5.13}
$$

$$\sum_{w \in W} b_{jww'} \sum_{i \in V:(i,j) \in A} x^t_{ijw} = \sum_{i \in V:(j,i) \in A} x^t_{jiw'} \qquad \forall\ j \in V_S \cup V_P,\ w' \in W,$$

$$t = 1, \dots, T, \tag{5.14}$$

$$\sum_{i \in V:(i,j) \in A} x^t_{ijw} \leq q^t_{jw} y^t_j \qquad \forall\ j \in V_S \cup V_P,\ w \in W,$$

$$t = 1, \dots, T, \tag{5.15}$$

$$\sum_{i \in V:(i,j) \in A} x^t_{ijw} \geq m f^t_{jw} y^t_j \qquad \forall\ j \in V_S \cup V_P,\ w \in W,$$

$$t = 1, \dots, T, \tag{5.16}$$

$$\sum_{t=1}^{T} \sum_{w \in W} \sum_{i \in V:(i,j) \in A} x^t_{ijw} \leq a_j \qquad \forall\ j \in V_L, \tag{5.17}$$

$$\sum_{i=0}^{\tau^t_j} y^{t+i}_j \leq (\tau^t_j + 1)(1 - \rho^t_j) \qquad \forall j \in V_P \cup V_S,$$

$$t = 1, \dots, T, \tag{5.18}$$

$$\sum_{t=1}^{T} \rho^t_j \geq 1 \qquad \forall\ j \in V_P \cup V_S, \tag{5.19}$$

$$y^t_j \in \{0, 1\} \qquad \forall\ j \in V_S \cup V_P,$$

$$t = 1, \dots T, \tag{5.20}$$

$$\rho^t_j \in \{0, 1\} \qquad \forall\ j \in V_P \cup V_S,$$

$$t = 1, \dots, T, \tag{5.21}$$

$$x^t_{ijw} \geq 0 \qquad \forall\ w \in W, (i, j) \in A,$$

$$t = 1, \dots T, \tag{5.22}$$

$$\xi^{t,s}_{iw} \geq 0 \qquad \forall\ w \in W, i \in V_O,$$

$$t = 2, \dots T + 1,\ \ s \in \mathcal{S}. \tag{5.23}$$

In the objective function (5.12), the recourse costs are represented by the last term, given by the penalties for treating the excess flow waste outside the network. The other terms are the deterministic version of the corresponding terms of objective function (5.1).

Waste collection constraints (5.13) impose that the generated waste is collected and shipped either inside the network or outside the network. Constraints (5.14)-(5.22) are adapted from those already considered in ($M1$). Finally, constraints (5.23) define the decision variables $\xi^{t,s}_{iw}$.

The scenario tree that describes the situation of model (5.12)-(5.23) is shown in Figure 5.3 in an example with $|\mathcal{S}| = 3$ and $T = 12$.

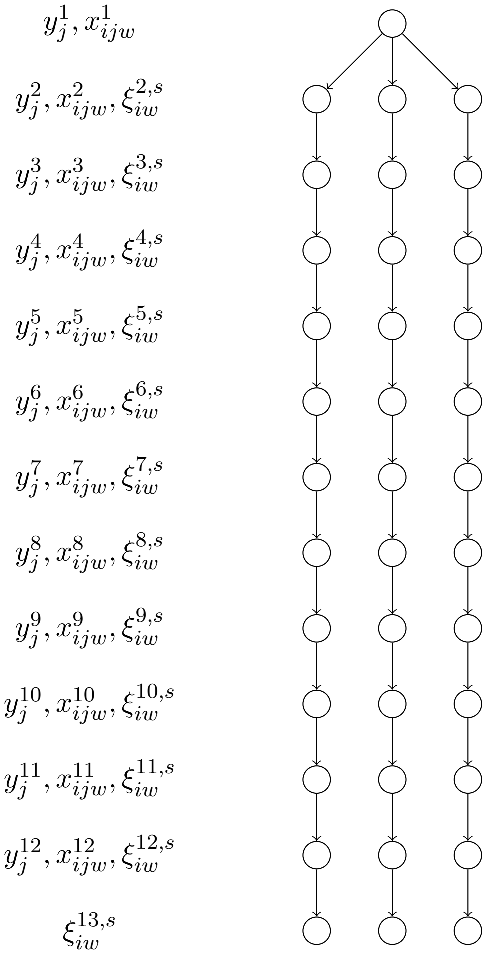


FIGURE 5.3: Scenario tree associated with formulation (5.12)-(5.23) with $|\mathcal{S}| = 3$ and $T = 12$

5.5 Scenario Generation

From our industrial partner Herambiente SpA, we received monthly historical data of unsorted waste (UW) generated in years 2011, 2012 and 2013 in several towns: this amount of data is not sufficient to obtain a good estimation for probability distributions of the uncertain waste generation values. For this reason, a set of 15 equiprobable scenarios was generated directly from the available historical data. The considered towns are in the Italian region of Emilia-Romagna, which is located on the Adriatic Coast. The town population is hence generally subject to seasonal trends: during the summer, in internal cities such as Bologna people migrate, while, during the same period, coastal cities, for example Rimini, receive tourists. The UW produced in towns is clearly dependent on the number of inhabitants, therefore the historical data are characterized by seasonality as well. Scenarios were obtained by aggregating months with a similar waste generation profile: in particular, all the values of the 5 months of November, December, January, February and March for 3 years were considered as future scenarios for the waste generated in such months. In this way, we get 15 scenarios from the historical data, while preserving the effect of seasonality.

5.6 Preliminary Computational Results

In this section, we report the numerical testing based on an instance extracted from an existing waste management network provided by the consulting company Optit Srl. Privacy issues of the considered data do not allow us to explicitly give their values, even if small modifications were made to the original data.

Our industrial partners gave us: the actual values of UW generated in 124 towns of Emilia-Romagna and some relevant information of the waste management network associated with such waste generation sources. In the real-world situation, the network is meant for treating every commodity of municipal waste. Hence, when we consider that only the UW is produced in urban sites, then the set of facilities is oversized and parameters like capacity limitations may be overestimated. This motivated the extraction of a smaller instance. We investigated the trade-off between considering a waste management network of realistic size and dealing with the memory limitations of the modeling language used in the implementation, namely AMPL (Fourer et al. [95]). We also observed that using too few plants would result in trivial stochastic solutions of planning (i.e., every facility is operative throughout the whole year in every scenario).

The instance we considered has the following features. The original set of 124 waste generation sources is treated by 28 plants, divided in: 7 separation facilities, 6 Waste-to-Energy (WtE) facilities, 6 other processing plants, 8 landfills and 1 market for the Electric Energy (EE) produced by WtEs. Beside UW, the network nodes can accept other 13 waste commodities, obtained as results of the various operations taking place in facilities. As a consequence of the "shrinking" of the real network, some of its original parameters were changed accordingly. Some of the processing costs for landfills were increased, so as to discourage a direct shipment of UW from sources to disposal sites. In order to limit an excessive impact of the revenue for EE in the objective function, a capacity restriction for incoming UW in WtEs was set. The transportation costs c_{ij} were determined as the distance between nodes i and j; for the sake of simplicity, for the moment we neglected economies of scale in the transportation costs (see Section 4.4.5). In our tactical horizon of planning, there is no possibility of closing permanently a facility or building a new one; hence, in objective functions (5.1) and (5.12), we only required to make the facilities operate as less as possible by setting all activation costs f_j^t to 1.

Since overall capacities of landfills and length of deactivation terms for facilities were not known from the real data, we preferred not to impose the constraints (5.6), (5.7) and (5.8) in (M1) and (5.17), (5.18) and (5.19) in (M2).

For both models, we report the following stochastic measures introduced in Section 1.2.4: the optimal value of the stochastic problem, also called *Recourse Problem*

(RP), the *Expected Value of Perfect Information* ($EVPI$) and the *Value of the Stochastic Solution at stage t* (VSS^t). The $EVPI$ is computed as the difference between the RP and the *Wait-and-See* (WS) value (Birge and Louveaux [38], Kall and Wallace [151]). Since the stochastic models are multistage, the VSS is dependent on every stage t of the planning horizon: each VSS^t measures the difference between the expected result of using the expected value solution EEV^t and the RP (Maggioni et al. [184]).

The stochastic models (M1) and (M2) were implemented in AMPL and solved with Cplex 12.6.3.0 ([70]) on a Intel Core $i5 - 4440$ machine with 3.10 Ghz CPU. Due to the limited size of the scenario tree, the resolution of both models is relatively easy. Cplex on (M1) requires 84.88 seconds of computation, while (M2) is solved in 8.70 seconds: the difference in the computational times is motivated by the higher number of decision variables of formulation (M1).

5.6.1 Model (M1)

Model (M1) reports an $EVPI$ that is 0.039% of the RP. This small percentage indicates that the price the waste manager should be willing to pay for obtaining a perfect knowledge of the waste generated in sources is negligible. Hence, the stochastic formulation is not a profitable decision support system in this case. This is also confirmed by $VSS^1 = 2357.36$, which is relatively small in comparison with an RP of the order of magnitude of 10^7: hence, the stochastic model gives a solution quite close to deterministic model solution where uncertain parameters are approximated by their mean values. For periods t following the first one, whenever variables x are fixed at their EV solution values, the waste collection constraints (5.2) can not be satisfied. Hence problem EEV^t is infeasible, and $VSS^t = +\infty$. In such situations, the analysis of pairs subproblems with the *Multistage Sum of Pairs Expected Values* ($MSPEV$) (Maggioni et al. [184]) could give additional insights on the significance of the stochastic solution. Despite the relatively small value of VSS^1, the values of VSS^t for stages t ensuing the first one show that the deterministic solutions in a multistage stochastic setting are largely inappropriate.

5.6.2 Model (M2)

In the deterministic counterpart of Model (M2), the waste manger will never require the treatment of excess waste flow, because the perfect knowledge of the future allows a perfect determination of the decision variables x. This explains why the $EVPI$ of (M2) is a high percentage (i.e., 30.62%) of the stochastic solution RP. In this formulation, the stochasticity of the waste generation plays and important role in the decision making. Regarding the VSS measures, the EEV problems are infeasible already at the first stage. Indeed, being non-negative, the corrective flows ξ cannot satisfy the constraint

(5.2) whenever the waste flow x in the EV problem collects more than the actual waste generated in a source. The infeasibility of such deterministic models motivates the adoption of the stochastic model M2.

5.7 Conclusions and Future Works

The work described in this chapter is a starting point for considering two-stage multiperiod stochastic formulations for addressing a SWM tactical problem of realistic size. The preliminary computational results of the model introduced in Section 5.4.2 highlight the impact of the random parameters on the planning decisions. In (M2), the $EVPI$ shows that good estimation of the waste generation could yield important cost savings for the waste management company and, in addition, the values of VSS^t ($t = 1, \ldots, T$) indicate a bad behavior of the deterministic solution in the stochastic framework.

In order to complete the models validation from a stochastic point of view, we will test them on bigger sets of scenarios generated from predictive models (Maggioni et al. [183]): in-sample and out-of-sample stability will be analyzed (Kaut et al. [155]). A proper multistage stochastic formulation could also be developed, in the case the uncertain parameters are revealed at the end of every period.

Chapter 6

Overview of Optimization Problems in Electric Car-Sharing System Design and Management

Car-sharing systems are increasingly employing environmentally-friendly electric vehicles (EV). The design and management of Ecar-sharing systems poses several additional challenges with respect to those based on traditional combustion vehicles, mainly related with the limited autonomy allowed by current battery technology. In this chapter, we review the main optimization problems arising in Ecar-sharing systems at strategic, tactical and operational levels, and discuss the existing approaches often developed for similar problems, for example in car-sharing systems with traditional vehicles. We also outline open problems and fruitful research directions.

6.1 Introduction

Car-sharing is a general public mobility mode that is based on the shared use of vehicles by a set of users, who are generally subscribers of the service and pay flat and per-use fees. These systems were introduced around 1970-80 in some limited pilot implementations (see Shaheen et al. [225]), but only recently have seen a considerable development in urban areas. In huge cities congestion and parking costs make the ownership of private cars much less attractive for citizens who rely on public transportation for their regular commuting, and need cars only for special purposes. For a general overview of car-sharing systems we refer to Shaheen et al. [225] and Millard-Ball et al. [190], whereas a recent survey on optimization problems arising in such context is given by Jorge and Correia [149]. Finally, the important aspect of demand estimation for car-sharing systems is discussed in Stillwater et al. [232] and Schmöller and Bogenberger [221].

Car-sharing systems are increasingly employing environmentally friendly vehicles that may reduce the overall negative impact of the mobility on the environment, and may have easier access to congested urban areas. For car-sharing systems the most commonly used environmentally friendly vehicles are indeed electric ones. In this chapter, for short we indicate car-sharing systems employing electric vehicles as Ecar-sharing systems.

As described in Pelletier et al. [204, 205], several types of electric vehicles actually exist and their characteristics may influence heavily their use possibilities in general and in relation to shared transportation systems. In particular, we consider plug-in electric vehicles (PEVs) that may be charged by plugging-in them into the electric grid. In turn, these vehicles can be classified into plug-in battery electric vehicles (PBEVs), which use the power provided by the battery only, and plug-in hybrid electric vehicles (PHEVs) which also have an internal combustion engine. Both vehicle types are able to recover energy generated during travel (from braking and driving downhill) to recharge the battery. Whenever no specific distinction is required, we call all these vehicles electric vehicles.

For what concerns the organizational issues, an important distinction has to be made between *two-way* (or *roundtrip*) systems, in which the vehicle must be returned to the station where it has been picked up, and *one-way* systems in which vehicles may be also returned to a different station. The second model is clearly more flexible for the users but, as we will extensively discuss in the following, it requires a rebalancing of the vehicles at different stations during the service. We finally mention that recently some car-sharing systems in which vehicles are no longer based at specific stations were introduced. Such systems are generally called *free-floating* (see e.g., car2go and BMW DriveNow).

Designing and operating car-sharing systems that use electric vehicles poses additional technological and practical challenges with respect to the systems employing traditional combustion vehicles. For example, the relatively limited autonomy of currently available electric cars requires recharging the vehicles during the day, which has to be performed at specific charging stations. In addition, due to the high costs involved, not too many charging stations have been built, and charging times can be quite long unless expensive fast-charging stations are present. Finally, the electricity consumption is considerably affected by the driving and environmental conditions (e.g., the speed profile or the outside temperature) that need to be accurately modeled to better estimate the actual charge status of the vehicles during the day.

In the following sections we examine the main problems that are relevant for the optimal design and management of electric car-sharing systems. We note that the existing literature on Ecar-sharing is very limited. Therefore, on the one side we highlight the optimization problems that arise in this context. On the other side, we examine the relevant literature on related problems, such as works focusing on electric vehicles (privately owned, taxis, etc.) or on car-sharing systems with conventional vehicles. For each such problem we both describe the characteristics that have been faced so far in the literature and discuss the components of real-world systems that have not been examined so far, so as to provide interesting and practically motivated research directions.

More precisely, we organized the exposition into two separate sections. The first part (Section 6.2) is devoted to strategic and tactical problems, which are appropriate in the design of the systems. Within such category falls mainly the problem of locating the charging stations for the electric vehicles and for privately owned cars (Section 6.2.1). Section 6.2.2 discusses the tactical problem of defining the allocation strategies for the assignment of vehicles to the stations.

In the second part (Section 6.3) we present operational problems that arise in the short-term management of Ecar-sharing systems. Section 6.3.1 introduces the relocation of vehicles between the available stations, which is required to balance the supply and demand patterns. Section 6.3.2 examines the possibilities offered by battery-swap technologies and Section 6.3.3 considers the computation of shortest paths specifically designed to incorporate the main characteristics of electric vehicles. Section 6.3.4 deals with the definition of multi-stop travels for electric vehicles that typically occur in freight distribution. Finally, Section 6.4 draws some conclusions.

6.2 Strategic and Tactical Problems

As their name suggests, the problems of this class deal with making good high-level, big-picture decisions. These determine the overall structure of the underlying car-sharing system and can therefore have a great impact on how well the system performs. Decisions made at this level are usually long-term, i.e., once they are made, they cannot easily be reversed. As they often imply high costs, they also have a significant impact on the car-sharing operator. Thus, high solution quality is of great importance for these problems. Combined with the fact that strategic decisions need not be made very frequently, this suggests the use of exact or combined methods for solving them.

Although some pilot systems are already in use, not much scientific literature dedicated to the study of the design and operational challenges of Ecar-sharing systems (from a general perspective) exists. Notably, Barth and Todd [24] were among the first to consider the use of electric cars in the context of car-sharing systems. Based on a case study from a resort in Southern California, they concluded that (already) 3-6 vehicles are sufficient per 100 trips of each day to satisfy customer waiting times, but approximately 18-24 vehicles would be necessary to also minimize the necessary number of relocations. Besides the number of vehicles per trip, they conclude that the relocation algorithm and the used charging scheme are the main factors for successfully using such a system. Note that particular characteristics of the considered case include the fact that trips are shorter than 5 miles on average, thus, the charging state of cars never drops below approximately 70%.

Considering a real-world use case from Genoa, Cepolina and Farina [55] are concerned with the design of a flexible, multi-station Ecar-sharing system for pedestrian areas. Their aim is to optimize the dimension and distribution of the fleet among a set of stations at the beginning of operation, so that the sum of total transportation and waiting costs is minimized. Particular characteristics of the system include the possibility for instant access, open ended reservation and one-way trips. A simulated annealing approach that uses a microscopic simulation of user behavior and waiting times is developed, in which a subset of users is assumed to be flexible in the sense that they have an associated set of acceptable stations. Recharging is not explicitly treated but simply assumed to occur in idle times and no explicit relocation actions are considered (i.e., relocation by users). The authors analyze the cost changes with respect to the total number of vehicles and, as in Barth and Todd [24], the influence of the vehicle-to-trip ratio on the total average waiting time.

Other pilot implementations are that of the Kyoto public car system project described in Kitamura [161], and the system with different types of electric vehicles discussed in Luè et al. [181].

Strategic decisions arising in Ecar-sharing systems mainly involve planning locations and sizes (i.e., numbers of charging slots) of charging stations throughout the operational region. The operator's main goal is to minimize their cost arising from building the stations while at the same time ensuring that the profit obtained from satisfied user requests during operation is maximized. Since users will only consider using a car-sharing system if their requests are accepted with a relatively high probability, an operator is facing a difficult trade-off between the initial costs to set up the car-sharing system (long term investment) and the profits obtained later on (operational phase), especially since the latter are highly uncertain.

Tactical decisions are instead related to mid-term planning horizons. Within this time horizon the main optimization problem that is relevant in Ecar-sharing systems is that of allocating the vehicles to the charging stations. Such a problem is mainly relevant for two-way models in which the initial position of the vehicles is critical and may need to be adjusted whenever substantial changes in the demand distribution patterns occur.

6.2.1 Location of Stations

As mentioned above, a key factor determining the performance of a car-sharing system is the location of each currently unused car within the system, as it determines which customers can actually use it. Since many car-sharing systems are station-based (i.e., cars are always picked up from and returned to a fixed set of parking spots owned by the car-sharing company), the location of these stations becomes equally important. This is especially true for those systems that use electric cars, since they must usually be recharged at the aforementioned stations during the day in addition to (fully) recharging them overnight.

In the following, existing studies on strategic decisions are classified into four categories: (i) location of charging stations in Ecar-sharing systems; (ii) location of charging stations to serve privately owned cars; (iii) location of charging stations for electric taxi cabs; and (iv) location of stations for car-sharing systems with non-electric cars. Note that we include literature related to the latter three categories, as the literature on Ecar-sharing systems is still sparse and as the arising optimization problems share many characteristics. A first brief overview that acts as a guideline to this section's content is given in Table 6.1.

TABLE 6.1: Classification of the literature related with location of charging stations.

Category		Methodology	
type	vehicle type	exact	heuristic / simulation
car-sharing	electric	[43]	
private fleet	electric	[22, 54, 60, 96, 121, 251, 254, 257]	[60, 107, 136, 250]
taxi cabs	electric	[15]	[223]
car-sharing	traditional	[68, 69]	[91]

6.2.1.1 Location of Charging Stations for Ecar-sharing systems

Boyacı et al. [43] describe a bi-objective mixed-integer programming (MIP) model for a station-based one-way system. Potential sites for the charging stations are first found by solving a set covering problem. Then the authors seek to optimize the location and size of the stations, together with the number of vehicles, their initial allocation and relocation during the system's operation with respect to both the operator's revenue and the users' benefit. To reduce the size of their model, they use an aggregated model where all relocations happen from or to imaginary hubs, each representing a set of stations, instead of between individual stations. The charge state of each vehicle's battery is not explicitly considered in the model – instead, the necessary pauses for recharging must be provided as an input. The authors evaluate their model for the Nice region by using data from an existing two-way car-sharing system and analyze the effects of various parameters like increased demand on the optimal solution. A preliminary study on the design of a comprehensive vehicle-sharing involving various types of electric vehicles and different types of ownership is described in Luè et al. [181].

6.2.1.2 Location of Charging Stations for Privately Owned Cars

The most studied case is that of the location of charging stations for privately owned cars. Frade et al. [96] provide an MIP formulation to decide on the location and capacity of electric vehicle charging stations with the objective of maximizing the demand covered under a certain service level and budget constraints. They conduct a case study based on real-world data from Lisbon (Portugal). A similar model is later developed by Cavadas et al. [54] and improved in order to provide a better coverage when some portion of the demand can be transferred between the successive stops of a trip. In addition to transfer of demand, the model is further adapted to a more realistic case where the variation of demand during the day is modeled by splitting the day into time intervals. The comparison of the models using data from Coimbra (Portugal) under different parameter settings reveals two important findings: (i) if there is a possibility of transferring demand, its inclusion in the model might provide significant improvements of the solution; and (ii) independently from its transferability, the consideration of the demand based on different time intervals prevents solutions with overcapacity, which might be the case if demand is aggregated.

Wang and Lin [251] consider a similar objective under budget constraints to decide on the location of multiple types of charging stations that differ in charging speed, and provide an MIP formulation for this problem. They also consider a variant in which the total cost to satisfy all demands is minimized. Both formulations are tested on a network from Penghu Island (Taiwan) and the test results show that the consideration of mixed stations yields benefits in terms of objective values compared to using a single station type only.

Minimization of the total cost is adopted also by Baouche et al. [22] when deciding on the optimal locations of the charging stations. Based on a survey on the metropolitan area of Lyon (France), they split the surveyed region into several demand clusters and calculate the energy demand at each of them. The MIP formulation they propose then finds the minimum cost set of potential charging stations that covers all energy demands. The cost takes into account both the construction of the stations and the energy demand for traveling to them. In addition, each station has a fixed type that determines how much charging they can provide. The individual state of vehicles, namely their location or charge state, and the temporal component of demand is only considered in an aggregated way.

A similar approach is used by Chen et al. [60] for the Seattle (Wa, USA) area. Their MIP model determines which charging stations should be opened to minimize the total walking distance required for satisfying all demand. The authors note that a simple greedy heuristic finds solutions of similar quality, but with a significantly higher maximum walking distance.

González et al. [121] seek to find an optimal charging schedule for private electric vehicles in the Flanders region of Belgium with respect to the cost of electricity used. To estimate the recharging demand, traffic data for conventional vehicles is used. While the locations of charging stations that are opened are not considered in their problem variant (they assume that charging can happen at any time and place), the authors note that in their optimal solution, some zones show a charging demand significantly above the average, which suggests that they are prime candidates for the construction of public charging infrastructure. They also show that over 80% of all current trips could be performed with electric vehicles without requiring any charging outside of the owner's home and note that much of the charging required for the remaining vehicles could be done while the owners are at their workplace.

In contrast to the exact methods used above, Ge et al. [107] employ a genetic algorithm to partition a planning area into zones and assign each of them a charging station of appropriate size, using the required energy expenditure as a quality criterion. Their algorithm is then evaluated on a test instance. Similarly, Hess et al. [136] describe a genetic algorithm for placing charging stations to minimize the total trip distances. They use a traffic simulator, modified to account for electric vehicles, to generate data for the inner city of Vienna, on which they evaluate their algorithm.

Wang et al. [250] describe a heuristic algorithm for finding good locations for charging stations serving private electric vehicles, considering both existing gas stations and entirely new spots as potential sites. Their approach considers a number of objectives including demand coverage, factors relating to the power grid and municipal planning factors (which seek to keep the stations away from places where they might impact other traffic). The algorithm is evaluated on data gathered from the city of Chengdu.

An integrated MIP model that optimizes both the location of charging stations and the routing of electric vehicles is given by Worley et al. [254], with the objective being the minimization of the total cost, which consists of the costs for building stations, charging vehicles and driving. Another MIP based algorithm for finding the optimal charging station locations is presented by Xu et al. [257], who consider customer accessibility (both spatial and temporal), number of charging slots and crime safety as relevant factors.

6.2.1.3 Location of Stations for Electric Taxi Cabs

Electric taxi cab stations represent a good combination of the two previous categories. Sellmair and Hamacher [223] consider the problem of selecting existing taxi stands as possible locations for charging stations and determining the number of charging points per station. By using simulation techniques, customer trips between taxis stands are generated. The simulation is based on the GPS data collected from five conventional taxis in the city of Munich in Germany. The simulation takes the state of charge into account for deciding whether trips can be accepted or not. An iterative heuristic approach is used to determine the number and location of the charging stations.

Asamer et al. [15] present a study based on operational data of a radio taxi provider in the city of Vienna in Austria. Positioning data of approximately 800 taxis over 12 weeks, one for each calendar month, is used. The authors aim to find locations for a limited number of charging stations dedicated to taxis. Instead of assuming taxi stands as possible locations, regions are considered and the exact locations within the selected areas are identified in a post-optimization phase, where various soft constraints need to be considered. The spatially-distributed charging demand is aggregated, meaning that start and end locations of taxi trips within each region are summed up. Based on this data, a set-covering approach is used to model the location problem with the goal of maximizing the coverage of the aggregated demands. The problem is modeled as a MIP and solved using the IBM CPLEX solver.

6.2.1.4 Location of Stations for Non-Electric Car-Sharing Systems

As noted in this section's introduction, the problem of finding the optimal locations of vehicle depots in conventional (i.e., non-electric) car-sharing systems is closely related to that of finding the locations of charging stations for electric vehicles, since the factors determining a station's quality are similar (e.g., proximity to areas of high demand). One key difference between these two problems is that models for conventional car-sharing usually do not consider the vehicles' fuel state, since gasoline-powered vehicles can be refilled comparatively quickly.

Correia and Antunes [68] describe MIP formulations that optimize the operator's profit by finding the optimal set of vehicle depots that should be opened, as well as their size and the allocation of vehicles among them. Three different models that maximize the operators' profit are studied, in which (i) the operator has full freedom to decide whether or not to accept a potential trip; (ii) all trips need to be accepted; or (iii) trips may only be rejected by the operator if no vehicle is available at the pick-up station. The authors evaluate their model on input data for the Lisbon area in Portugal, and conclude that the operator's profits decrease significantly when all trip requests must be fulfilled. In another publication, Correia et al. [69] analyze the effects of increased user flexibility on the operator's profit. They develop an MIP formulation that allows users to select one of several potential starting and ending vehicle depots for each trip, with the additional option of providing them with information about the availability of cars or parking spaces at the relevant depots. By applying the model to the Lisbon data set from their previous paper, the authors find that the flexible models improve vehicle usage, but increase walking and total travel times.

In contrast to the aforementioned publications, which deal with finding an optimal solution with respect to some measures of quality, others deal exclusively with the simulation and evaluation of solutions. Fassi et al. [91] evaluate the effects of several growth strategies (like increasing the size of stations and opening new ones) on the activity of stations and members, as well as the members' satisfaction with the service.

6.2.1.5 Summary, Open Problems and Possible Research Directions

The main objectives in the station location problems for (electric and non-electric) car-sharing systems are to minimize the total cost or maximize the total profit of the car-sharing companies. The characteristics of the location of charging stations for privately owned electric cars can be mainly considered in two categories: problems that aim to minimize total cost while satisfying all demand, and problems that aim to maximize demand coverage under budget constraints. Additionally, objectives pertaining to user satisfaction are sometimes considered. This includes, in addition to the aforementioned demand coverage, objectives like minimizing the walking distance of customers.

The objective of maximizing demand coverage in Ecar-sharing systems seems to be an open problem in the literature and has yet only been addressed in the context of electric taxi cabs [15]. As suggested by [251], multiple types of charging stations can be included in location decisions. Such models could also be extended to consider certain characteristics of the electric grid, like varying charging capacity throughout the day. Improved solutions are obtained when possible transfer of charging demand is considered by Cavadas et al. [54] for the stations dedicated to privately owned electric cars. Adaptation of this idea to the Ecar-sharing systems might be worthwhile to investigate. To better capture aspects related to the particular characteristics of electric cars (i.e., very limited range, long recharging times) integrated models combining strategic and operational aspects seem worth investigating. In that respect, we particularly refer to variants that include detailed tracking of battery-state and recharging times. The high degree of uncertainty in terms of energy usage for individual trips also suggests further investigations of robust or stochastic problem variants. Furthermore, explicitly capturing the trade-off between naturally arising conflicting objectives (such as long term investment costs, short term profits, relative number of accepted user requests) in terms of bi- or multi-objective problem variants seem worth further studies.

More generally, an aspect that is worth investigating is the study of inter-modal people transportation problems that include (electric) car-sharing systems, i.e., to study the integration of (electric) car-sharing with public transportation and other means of transportation. Besides, considering the likely relatively short distances of many car-sharing trips within cities, a study of the trade-off between vehicle cost and vehicle range seems relevant for the case of electric cars.

Another possible avenue of research would be the development of a flexible pricing scheme that considers the variation of demand throughout the network at different times. This might eventually lead to a system where relocation of vehicles is mostly user-based. It is, however, unclear whether such a system would find acceptance among its potential users.

6.2.2 Allocation of Vehicles to Existing Stations

Besides relocating vehicles between stations (as described in the next sections), most papers do not seem to explicitly optimize the assignment of vehicles to stations. On the contrary, it is typical that vehicles are considered as origin of a given demand and stations are built and dimensioned to satisfy that demand, see, e.g. [60, 107, 121]. Whenever the actual positions of vehicles throughout a certain planning period (typically a day) are considered in an approach (that, e.g., considers a location-routing problem combining the planning of stations or relocations), an (initial) allocation of vehicles is implicitly optimized by not fixing the (initial) status, see, e.g., aforementioned articles by Correia et al. [69] and Boyacı et al. [43]. On the contrary, other articles (such as Baouche et al. [22]) do not consider these temporal components, but simply design a set of stations (with their capacity) in order to be able to fulfill the demand corresponding to the set of vehicles. Clearly, the latter, which in turn is not so different from other classical assignment problems (p-center, set-covering), is more appropriate for car-sharing systems in which only round trips are allowed and issues such as relocation are not important.

One example of a model that considers the initial allocation of vehicles as a decision variable to be optimized is given by Nakayama et al. [197]. The authors describe a genetic algorithm to optimize, among other factors, the number of vehicles within the car-sharing system and their location at the beginning of each day, given a fixed set of charging stations with a similarly fixed number of parking spots. The algorithm is then evaluated on data from an electric car-sharing operator from Kyoto.

6.2.2.1 Summary, Open Problems and Possible Research Directions

Since the initial placement and allocation of vehicles to existing stations is rarely considered as an explicit optimization problem but rather assumed to be given, no particular objectives and general constraints have been identified.

An interesting aspect that needs further investigation concerns the integration of vehicle allocation with general location and relocation aspects.

6.3 Operational Problems

We consider here the optimization problem arising in the operational management of Ecar-sharing systems. Such problems may be grouped into two main classes. The first one is related to the within-day optimal relocation of vehicles while the second considers the possibility of exchanging the battery at charging stations so as to restore vehicle autonomy. We also consider some relevant operational problems that have potential connections with the management of Ecar-sharing systems, namely, the electric vehicle shortest path and vehicle routing problems.

TABLE 6.2: Classification of the literature related with vehicles relocation (UB: user-based relocation strategy, OB: operator-based relocation strategy).

Reference	Strategy	Objective	methodology
[25]	UB	min. relocation costs	Simulation
[64]	UB	max. revenue and max. user's benefit	Simulation
[157, 158]	OB	min. relocation cost and rejected demand	Exact/Heuristic/Simulation
[196]	OB	min. relocation costs	Exact
[170]	OB	min. relocation distance	Simulation
[150]	OB	max. profit	Exact/Simulation
[49]	OB	max. number of relocations served	Exact
[43]	OB	max. revenue and max. user's benefit	Exact

6.3.1 Relocation of Vehicles for Multiple-Stations Car-Sharing

During the last years, the offer of one-way trip mode has experienced an increased popularity in car-sharing services with fleets of conventional or electric vehicles. One-way car-sharing systems can be free-floating, in the absence of fixed parking spots, or *station-based*: in the latter case, reservations may be asked from the users. Since literature on free-floating services is very scarce, this section is focused on station-based systems. However, many issues described in this section apply to the free-floating case as well. The one-way option allows for a considerable increase in the number of potential customers interested in shared-use cars. This enhanced flexibility has a strong impact on the vehicle distribution in the service-provider network. Without the imposition of round-trips, an imbalance situation can occur and make the problem of ensuring vehicle availability in under-supplied stations a key issue for the system provider. In order to limit the unserviced trips and restrict economic losses of the car-sharing company, two types of relocation strategies may be implemented. In the first one, called *user-based* (UB) strategy, the relocation is decided by the customer itself, whereas in the second one, called *operator-based* (OB) strategy, relocation decisions are made by staff operators at a centralized or distributed level. The main characteristics of the papers examined here are presented in Table 6.2.

6.3.1.1 User-Based Strategies

From the system provider point of view, the organization of staff-relocation operations can carry an important economic load and cause operational difficulties. In order to alleviate such burden, Barth et al. [25] introduce two user-based relocation mechanisms called trip joining (or ride-sharing) and trip splitting. Reduced prices are offered to customers willing to accept these modifications of their trip mode. The trip demand data they consider is generated from the University of California-Riverside Campus fleet (UCR IntelliShare) historical database. The system offers trip joining when multiple users want to travel from one low-vehicle-quantity station to a high-vehicle-quantity station, and trip splitting in the opposite situation. Given the demand, a discrete-event time-step simulation model is presented. The simulation allows to calculate the reduction in operator-based relocations thanks to trip joining, trip splitting and the two techniques concurrently. Simulation results show that, in most cases, trip splitting proved to be more effective than trip joining in reducing the staff operators workload. Using these user-based techniques, a 42% reduction in the number of relocations is reported.

Clemente et al. [64] apply information and communication technology to the management of a one-way Ecar-sharing system. Real-time monitoring tools are used in order to propose economic incentives to the users, and help the rebalancing of vehicles in the network stations throughout the day. The authors used a timed Petri Net Framework to model the Ecar-sharing system. The customers response to the proposed trip alternatives modifies the random switches in the Petri Net. The proposed simulation model compares the "as-is" situation (no incentives), with two potential "to-be" strategies. In the "to-be" scenarios, users are encouraged to return cars as soon as possible (offline scenario) or to head to empty stations (online scenario); the latter situation requires the online monitoring of the system. Results on the Ecar-sharing system of Pordenone (Italy) are presented where the online scenario proves to be more profitable for the service provider. The authors conclude that relocation decisions rely on appropriate high-level strategic decisions; when such decisions are not accurately taken (e.g., the station fleet size), the relocation policy is not likely to be effective in solving the congestion problems.

To the best of our knowledge, user-based relocation strategies are not currently implemented by car sharing providers. Although the aforementioned papers simulate the impact of such strategies on profit, their actual potential is yet to be evaluated. However, nowadays some incentives to users are proposed in order to reduce the workload of providers (e.g. car2go gives free riding time if the users re-fuels the car).

6.3.1.2 Operator-Based Strategies

Existing car sharing providers usually perform overnight relocation. In the literature different practical relocation methods are described. Examples of such techniques can be found in Barth et al. [25]:

- Moving EVs with a truck (troublesome in cities)

- Towing a single EV to a "service" car
- Transporting operators to relocation positions by using a "service" car

Notice that, unless otherwise stated, the following papers evaluate the benefits of introducing relocation during the daily service, regardless of the specific technique that will be implemented.

Contributions by Kek et al. [157] and Kek et al. [158] are motivated by the development of four shared-use vehicle companies in Singapore. The focus is on a multiple-station company that allows one-way trips; the customer also has the flexibility to modify the previously specified return station en-route. In the first paper, a relocation time-stepping simulation model is proposed and applied on a real set of shared-use vehicle data from commercial operations. Two operator-based relocation techniques are proposed. When service level is the main concern, the vehicle relocation from a neighboring station to an under-supplied station should be performed in shortest time (i.e., travel time to the over-supplied station and relocation duration). The inventory balancing strategy aims instead to relocate vehicles in order to gain an equilibrium in the vehicle distribution in the stations. Cost efficiency is the objective of such technique. The simulation model is validated with real commercial data trips over a typical one-month period. The performance is measured in terms of number of relocations; besides, Kek et al. [157] measures time in which parking slots in a station are either full (full port time) or empty (zero vehicle time). The simulated indicators show fidelity in replicating the trends occurring in the real situation; besides, they provide information on the potential cost savings which could be achieved without impacting the level of service. The authors observe that the individual change of the car-sharing systems parameters has no significant performance impact: this is due to the strong interrelation of operating parameter in such systems.

In Kek et al. [158], the authors present a three-phase optimization-trend-simulation (OTS) decision support system for car-sharing operators to determine a set of near-optimal manpower and operating parameters. A MIP in a time-space network determines the lowest-cost resource allocation and vehicle scheduling, given inputs on station characteristics, vehicle relocation costs and historical customer usage patterns. In the second phase of Trend Filtering, the suggested staff and vehicle activities output from phase one are filtered through several heuristics in order to produce a recommended set of operating parameters. Such output parameters are finally used in the relocation simulator previously described in [157]. The solution approach has been tested on real operational data from Singapore. Results show remarkable improvements in the system performance according to the proposed measure of effectiveness.

Considering the same case study of Kek et al. [157] and Kek et al. [158] in Singapore, in Nair and Miller-Hooks [196] the aim is finding a least-cost fleet redistribution plan such that most demand scenarios are satisfied. The probability distribution of users demand is defined by data collected with an Intelligent Transportation System infrastructure that enables monitoring of the trips. A stochastic MIP with joint chance constraints is formulated. The feasible region of the problem is non-convex. Two solution methods are presented: when demand at stations is correlated, an enumeration procedure based on the concept of p-efficient points is applicable; when the demand at each station is assumed to be independent, a cone-generation solution method is used. Solutions of the proposed case study proved to be robust in simulation studies.

Jorge et al. [150] present two methods for implementing operator-based relocation strategies. The strategic decision of location of stations is taken by adapting the model proposed in Correia and Antunes [68] to the case in which the demand between existing stations is not always satisfied. The first relocation method is based on a novel MIP formulation in a time-space network that aims to maximize the daily profit of the car-sharing system. The second method is a discrete event time-driven simulation for testing two real-time relocation policies. Such strategies consider different frequencies for checking whether a station is a supplier (vehicles in excess) or a demander (vehicles shortage). The two solution approaches were applied, independently and in a combined way, to several realistic scenarios in a case study in Lisbon. The optimized relocation decisions for these networks indicated significant potential profit gain with respect to the case of no relocation actions. The optimal solutions of the mathematical model provide upper bounds on the economic gains that are achievable with relocations since its input data are based on full knowledge of future daily trip demands. Even though trip reservation is necessary in the considered system, the simulation results based on real-time policies are remarkable.

Lee and Park [170] propose an operation planner for relocation staff operations in Ecar-sharing systems. The relocation scheme consists of three steps covering the relocation strategy, the action planning and the staff operation planning, respectively. The demand is estimated by using the extensive Jeju City dataset on actual trips consisting of pick-up and drop-off points collected from a taxi telematics system. Relocation is assumed to be carried out during non-operation hours. The third phase is the main focus of the paper. It implements the relocation staff operations (i.e., moving from an initial to a final station). Single relocation team scheduling is considered for simplicity. The scheduling phase is tackled by using a genetic algorithm in which the relocation distance is the main performance metric considered.

In Bruglieri et al. [49], the authors claim that relocation activities that rely on a truck for auto transport may not be practically implementable in urban environment, since stations may be hardly reachable by the trucks. To overcome this problem, they propose the use of folding bicycles for staff operators relocation movements from an under-supplied station (drop-off) to an over-supplied station (pick-up). Such relocation approach generates a specific pickup and delivery problem called the Electric Vehicle Relocation Problem (EVRP). Given a set of pick-up and drop-off requests defining the network graph, the relocation is formulated as a Vehicle Routing Problem aiming to maximize the total number of requests served. Their MIP model explicitly considers the battery degradation profile using linear assumption. The estimation of the demand has been performed by studying historical data on private car movements in the city of Milan, and restricting these data to the estimated percentage of users interested in using the car-sharing service. A car-sharing simulator has estimated the unbalances due to the projected travel demand. Computational results on realistic instances show that using two workers with a duty time of 5 hours is sufficient to satisfy a high percentage (about 86%) of the relocation requests.

Boyacı et al. [43] present an integrated (strategic, tactical and operational) framework to decide on the location of stations (see Section 6.2.1.1), on the number of parking slots to satisfy the uncertain user demand, on the assignment of users to slots and on the operator-based relocation actions. The considered Ecar-sharing system is one-way, non-free-floating and reservation-based: both the beginning and the ending station of the trip have to be specified. Demand centers represent sites that can be served by the same set of candidate stations; demands are obtained by an aggregation of orders of rentals, sharing the same set of origin and destination points and common departure and arrival time intervals. The considered graph is a time-space network. A set of scenarios is considered for coping with the stochasticity and seasonality of the demand. The authors develop a bi-objective MIP model. An aggregated model that uses the concept of virtual hubs is presented for the practical solution of instances based on the large-scale car-sharing system in Nice. Extensive sensitivity analysis for relevant parameters is performed. The model evaluates the trade-off between operator benefit and users' level of service, showing that the investment in relocation personnel is worthy both from the company and customers point of view.

6.3.1.3 Summary, Open Problems and Possible Research Directions

We now summarize the main constraints and optimization objectives considered in the literature for relocation in Ecar-sharing systems.

At each network node, each activity is restricted to begin after the previous one is completed (see [158]). Taking into account relocation action and maintenance activities, the number of available vehicles is updated during the operating day. A limit on the number of rejected demands and vehicle returns is imposed.

There are a number of capacity constraints present in these models. In [158] and [43], station capacity constraints are imposed: in each time discretization step, the sum of available and unavailable vehicles in a station can not exceed the station capacity.
In [158], [196] and [43], the authors limit the number of vehicles relocated out of a station with the number of vehicles available at the start of the planning period; also,

the number of vehicles relocated to a station cannot exceed the number of available slots. These conditions are called capacity constraints.

When time-space network representation is used (see [150]), the vehicle flow at each node in the time-space network must be preserved. The stations must have enough parking spaces for vehicles present at each minute. Flow conservation constraints are also considered in [49] and [43]. In [43], atom-coverage constraints are introduced. An atom is a small geographic area that is eligible to receive the car-sharing service. The number of operating parking spaces in all open stations constitutes an upper bound to the number of relocation actions.

In [196], the probabilistic level-of-service constraints state that the redistribution plan must result in inventories that satisfy p-proportion of all demand scenarios in the planning horizon. The resulting system is called a p-reliable system.

In some cases (see [49]) time windows for customers requests are present. Therefore, specific service limitations, such as imposing precedence constraints in the visit time of nodes and bounding the duration of a route are considered.

Finally, specific restrictions characterizing Ecar-sharing systems are imposed in [49] and [43]. In the first paper, the distance traveled by an electric vehicle is assumed to be linearly proportional to the residual charge: it is imposed that an electric vehicle needs to have minimum residual charge (level) in order to perform a trip. In the second paper, the electric vehicles are required to be recharged in the arriving station after each rental operation. In addition, the number of vehicles in the station should be greater than or equal to the number of vehicles requiring charging.

In this specific area there are several open research directions. Regarding the simulation approaches for the impact of user-based relocation strategies, [25] and [64] underline the interest of estimating user participation rate in the proposed relocation activities. The first paper suggests to collect extensive statistical data for making this forecast. The second one proposes a detailed behavioral analysis of the users willingness to accept real time trip suggestions that would permit a more precise trip pricing policy.

Other research directions are represented by integrating the relocation action in the strategic planning phase of car-sharing management and to investigate the adoption of real-time relocation policies. In addition, using multiple relocation teams and combining operator-based relocation approach with pricing policies on the parking stations offered to the users, all seem promising options.

Several papers have underlined the strong interrelation between the different levels of decision-making in car-sharing systems problems. As already mentioned, the strategic decision of the location of stations has a huge impact on the tactical and operational issues, such as the routing of the shared-use vehicle fleet, in order to satisfy users requests. An integrated modeling approach seems a promising line of future research.

Car-sharing problems might be considered as real-world application in which a location-routing scheme is directly present or at least identifiable. The location-routing problem is a research category that considers the integrated solution approaches for tackling location problems in which the tour planning aspects are strongly interrelated with the strategic decisions. To the best of our knowledge, in literature, car-sharing problems have not been explicitly stated in location-routing framework yet and we refer the reader to the survey by Nagy and Salhi [195], which provides a good introduction to the problem. More recently, Prodhon and Prins [211] updates the first survey presenting the multi-echelon problems and several other variants. Finally, the survey by Drexl and Schneider [82] proposes future research directions from the methodological and modeling point of view, such as the integration of revenue management in location-routing formulations.

6.3.2 Battery Swap

One main challenge for the large-scale spreading of battery-electric vehicles is their limited range and the fact that in contrast to traditional vehicles, re-charging operations take a significant amount of time (with the exception of expensive and not yet very widespread fast-charging stations such as Tesla Superchargers and CHAdeMO). Especially for long distance travel, overnight recharging is not sufficient. Thus, battery swapping (rather then recharging) has been considered as a viable alternative, in which the batteries are owned by a company and users simply exchange their currently used (nearly empty) battery with a fully charged one at predefined battery swapping stations (BSSs). A main advantage from a users perspective is that this process can be done in a few minutes (i.e., approximately in the same time frame needed for refueling a traditional car). Even if such technological approach is made difficult by the lack of standardization on batteries and by the huge investments required to set up the system, some interesting studies were presented in the literature.

Yang and Sun [258] study a location-routing problem arising in the delivery of goods to customers using a fleet of electric vehicles (EVs). Given a set of customer demands and of potential BSSs, the goal is to simultaneously determine the location of the battery swapping stations, the allocation of customers to EVs as well as that of EVs to BSSs. In addition, tours from the single depot to serve all customers are designed that consider the selected BSSs and the driving range of the vehicles. The objective is to minimize the total costs arising from the construction of BSSs and the service of the demands with the EVs. Energy consumption and maximum vehicle range are considered to be proportional to the traveled distance. Two flow-based integer programming models are proposed; only the second one allows to revisit BSSs (i.e., to pass at a station / customer multiple times). In addition, two heuristic approaches are studied. The first one is a tabu search that mainly focuses on the location of BSSs and uses a modified Clarke and Wright [63] savings algorithm to heuristically compute a set of routes based on the currently selected swapping stations. A radius-covering method is applied to find an initial set of BSSs. In addition, a hybrid heuristic combining various approaches (namely, modified sweep heuristic, iterated greedy and adaptive large neighborhood search), is described. The main idea is to initially ignore most of the constraints (i.e., battery driving range, BSS location) and subsequently refine a candidate solution to satisfy all conditions. Finally, a last phase aims at improving solutions that are already feasible for the considered problem. Computational experiments are performed using data sets from the CVRP in which all nodes are considered as potential BSSs. Results show that revisits often pay off. The influence of different maximum driving ranges is also analyzed.

Mak et al. [185] aim to optimize location and sizing of BSSs at strategic locations along a network of freeways. They argue that the strategic network decisions need to be taken before observing the actual demand. Therefore, they propose distribution-robust optimization problems where in a first phase the location of BSSs needs to be decided while the number of batteries stored at each BSS can be determined after the uncertain factors are realized. Two variants in which either the expected building and operating costs are minimized ("cost-concerned" model) or a robust estimate of the probability to meet a certain return-on-investment target is maximized ("goal-driven" model) are considered. Models based on mixed-integer second-order cone programming are derived and potential impacts of battery standardization and advancements on the deployment strategy are studied. Computational experiments are performed using instances based on the San Francisco Bay Area freeway network. It is also pointed out that there exist real world cases (Israel) in which the set of candidate BSSs corresponds to the set of existing gas stations and that upper bounds on the number of batteries per location need to be considered. This restriction arises from the capacity of the electrical grid. Furthermore, the number of arising swap-demanding EVs are treated by a Poisson process, the swapping is assumed to be instantaneous, and a heuristic first-in-first-out strategy for battery selection is considered.

Li [172] studies the scheduling of electric transit buses when either battery swapping or fast charging is employed. An exact branch-and-price algorithm (including stabilization and an initial construction heuristic) as well as heuristic variants based on truncated column generation, variable fixing, and local search are developed. A computational study is performed on instances that are based on publicly available real-world transit data. Besides comparing variants of the proposed algorithms, the results achieved are benchmarked against approaches for other types of buses (gas, diesel, hybrid). Despite the main disadvantage of electric buses, such as the need of deadhead travels to battery stations, the author concludes that the total operational costs of electric buses are smaller than those of the other options. The use of electric buses, therefore, represents a viable alternative also because they produce zero emissions during operation.

Other authors (see, e.g., Chen and Hua [59]) focus on the placement of battery swapping stations without discussing too many aspects that differ from the planning of other re-charging stations; we therefore refer to Section 6.2.1 for more details.

Another stream of research concerned with battery-swapping deals with the replacement of degraded batteries within a fleet of vehicles by new ones. Almuhtady et al. [7] study different swapping and replacement policies within maintenance of a fleet by a mathematical model as well as two metaheuristic approaches: genetic algorithm and simulated annealing. Experimental results using data inspired from real world are shown.

6.3.2.1 Summary, Open Problems and Possible Research Directions

Existing approaches in the literature are mainly concerned with either minimizing the total costs in installing (and possibly maintaining) battery-swapping stations. In addition, total routing costs are partially considered in case of classic vehicle routing applications. One exception to this trend is given by Mak et al. [185] who also consider a variant in which the probability to meet a certain return-on-investment goal is maximized. Most of the related works consider constraints limiting maximum travel ranges (whenever a location-routing problem is considered) and restrictions to relatively small sets of potential swapping stations (often only existing “traditional” gas stations). Besides, upper bounds on the numbers of batteries per location arising from limitations of the electric grid are considered (in particular if fast-charging is employed).

Open problems in this area include the appropriate integration of charging times within the overall models and the potential consideration of charging at different speeds instead of assuming a given number of available, charged batteries. Furthermore, integration of aging and replacing aspects of batteries (with respect distance traveled, charging cycles) into battery-swapping problems can be a relevant topic.

6.3.3 Electric Vehicle Shortest Path Problems

This section discusses optimal path problems involving electric vehicles – with focus on PBEVs – and their specifics. In the car-sharing context these problems might be relevant when the provider wants to estimate the energy consumption of customer trips or when navigation services are offered to customers.

In general one can think of many different practical problem variants of finding an efficient path from A to B while respecting the battery limits (lower and upper bound) of PBEVs. Among them, the following objectives might be relevant:

- minimize energy consumption,

- minimize travel time, and/or
- minimize total costs including costs for traveling, charging, drivers, etc.

Several additional aspects may be considered, e.g.:

- visits to charging stations,
- charging times,
- energy recuperation, i.e., negative energy values on arcs, and/or
- charging station capacities.

An extensive survey on EV shortest path problems and algorithms can be found in Pelletier et al. [205]. In the following, we review important works and extend this survey.

Artmeier et al. [13] minimize energy consumption while allowing recuperation. Since lower and upper bounds of the battery charge have to be respected, the resulting problem is a variant of the constrained shortest path problem that is NP-hard in general. However, here the optimized and constrained resource are the same, finally leading to a polynomial-time algorithm, i.e., a modified Bellman-Ford algorithm. Since the energy consumption on links also depends on the speed on the previous link on the selected path, applying the label-setting algorithm on the original graph is not possible. Thus, the authors describe the construction of an energy graph in which nodes are replicated for each velocity value on incoming arcs. Since the node degree in street network is three on average, the corresponding energy graph is not much larger than the original one.

Eisner et al. [85] extend the work by Artmeier et al. [13] by applying an adaptation of Johnson's potential shifting technique to obtain non-negative edge costs and finally run Dijkstra's algorithm to execute queries in polynomial time. Additionally, the idea of contraction hierarchies is used to further dramatically speed-up shortest path queries.

Sachenbacher et al. [219] also improve the work by Artmeier et al. [13] by considering an A*-related shortest path algorithm. They show that an energy consumption function depending on distance, elevation, and speed provides a consistent heuristic for the A* algorithm, i.e., an energy-optimal route can be found. Their approach significantly outperforms the standard Bellman-Ford and Johnson variants and additionally allows to use dynamic energy information at query-time.

Cassandras et al. [53] consider the problem of finding a path from A to B of a single PBEV with minimal total time while respecting the battery constraints and determining which and how long charging stations are visited. The total time includes both travel and charging times. A nonlinear MIP is presented and under several assumptions the authors transform it to an LP: i) at each node there is a charging station with a fixed charging rate, and ii) all energy consumption values on arcs are non-negative. The authors also study the path routing problem with multiple vehicles involving traffic congestion issues and assuming that all vehicles are controlled by a central system. Several non-linear MIPs are proposed to solve this problem.

Arslan et al. [12] deal with an NP-hard minimum-cost path problem for plug-in hybrid electric vehicles (PHEVs) (with both combustion and electric engine) with intermediate fueling/charging stations. They transform the original graph in a way that only origin, destination, and fueling/charging nodes are left. Edges represent the shortest paths between the corresponding nodes in the original graph. When considering only PBEVs, it is possible to find a minimum-cost path from A to B in this graph in polynomial time (e.g., by Dijkstra's algorithm), visiting fueling/charging stations if necessary. For PHEVs, the additional decision of choosing the driving mode makes the problem NP-hard. In an extended problem variant the authors additionally consider vehicle depreciation, stopping, and battery degradation costs. An exact MIP model with quadratic constraints, a dynamic programming and a shortest path based heuristic are presented to solve this problem.

6.3.3.1 Summary, Open Problems and Possible Research Directions

In earlier works, the main objective is to minimize the energy consumption on the total path. More recently, researchers often consider the minimization of the total travel time while respecting the energy limits, which might be more relevant in practical applications. Additionally, complex cost functions are used combining the (time-dependent) costs for traveling, charging, battery degradation, etc.

The most important common constraints are based on the physical limits of the battery of PBEVs. Because of the currently still quite small battery capacities, PBEVs quickly run out of energy. Recuperation, i.e., the recovery of energy when breaking, may compensate partly for this deficiency. This, however, leads to negative energy values on links and thus to more complicated optimization problems.

The systemic battery limits of PBEVs may also lead to further related constraints: If visits to a given set of charging stations are allowed, then corresponding charging times and station capacities have to be considered, which may also be time-dependent based on the overall state of the underlying electrical grid.

Many authors use simplified formulas to calculate the energy consumption on links. Here, more realistic (possibly nonlinear) functions involving a large number of influencing factors may be considered. For some applications, such detailed energy consumption models may not be needed, but nevertheless it should be clear which components mostly contribute to the energy consumption. A sensitivity analysis for a complex energy model might be performed to identify the crucial aspects.

Most works consider only a single vehicle and search for the best path in an egocentric point of view. For governmental stakeholders and local authorities, however, it might be more relevant to consider a global system optimum rather than a local egocentric optimum. Thus, more sophisticated models involving multiple vehicles and complex evaluation functions may be considered in the future.

Realistic energy consumption models and cost functions often involve nonlinear terms. Finding accurate linear approximations for these functions might be a way to finally obtain efficient solution approaches for these problems. Discretization might be a promising candidate to reach this goal.

6.3.4 Electric Vehicle Routing Problem

This section discusses works on vehicle routing problems in which traditional vehicles are either replaced by or mixed with PBEVs. Such problems might be relevant for car-sharing providers if navigation services are offered, which involve finding routes visiting a set of locations given by the customer.

Since the battery capacity of electric vehicles is strongly limited, it may be necessary to re-charge the battery along a single route, possibly multiple times. In the literature, this limitation is handled quite differently, as discussed in the next paragraphs. An early survey on sustainable VRP variants can be found in Lin et al. [176]. The survey by Pelletier et al. [205] summarizes several aspects of electric vehicles, i.e., different types of electric vehicles, market penetration, incentives, OR related works, and research perspectives. More details on the specifics of electric vehicles can be found in Pelletier et al. [204]. Since the survey by Pelletier et al. [205] is quite extensive, here we only discuss papers which are particularly relevant or not mentioned in the survey.

In the green VRP introduced by Erdoğan and Miller-Hooks [88], routes for alternative-fuel powered vehicles are determined. A compact MIP based on Miller-Tucker-Zemlin [191] subtour elimination constraints (Big-M) is presented, minimizing the traveled distance while considering the limited distance, possible visits to alternative fuel stations, and upper bounds on the number of tours and their duration. In contrast to classical VRP variants, vehicles are assumed to be uncapacitated here. Refueling time is assumed to be constant, which is usually not the case for electric vehicles. The authors also propose two construction heuristics to create feasible solutions. The results indicate that as the number of fuel stations increases, costs decrease for the same number of served customers, more customers can be served, and the total distance traveled decreases.

Van Duin et al. [243] examine the fleet size and mix Vehicle Routing Problem with Time Windows with special focus on different types of electric vehicles for goods distribution. The battery limitations are considered by setting a maximal tour length, which can be completed with a single battery charge, i.e., recharging at specific stations is not allowed. A compact MIP based on Big-M constraints is presented without solving the model. To find solutions for a case study in Amsterdam, the authors developed a simple construction heuristic that provides satisfying results in their application.

Schneider et al. [222] extend the green VRP by integrating time windows (VRPTW), customer demands, and capacity constraints to the problem, while focusing exclusively on PBEVs. As a result, recharging times depend on the vehicles battery charge when arrival at a recharging station, and assuming a full recharge. The authors consider a hierarchical objective function first minimizing the fleet size and second minimizing the total travel distance. A hybrid metaheuristic combining variable neighborhood search with tabu search yields small gaps compared to a compact MIP model with Big-M constraints solved by CPLEX.

Frank et al. [98] consider the same problem as Schneider et al. [222], but involve load-dependent energy consumption: each arc is associated with an energy consumption value both for an empty vehicle and a single load unit. Then, the total energy consumption on an arc is linearly dependent on the amount of cargo loaded. The authors provide several linear MIP models for this problem variant: i) a compact model with Big-M constraints, ii) a compact two/three-index-formulation with Big-M constraints allowing at most one charging station visit between two clients, and iii) a set-partitioning model. The same authors present in Preis et al. [209] a more detailed energy consumption model based on distance, altitude, load, and several vehicle properties. In a compact MIP model with Big-M constraints for the electric VRPTW, they minimize the total energy consumption. Additionally, the authors use tabu search heuristics to solve this problem.

Felipe et al. [92] also consider the same problem as Schneider et al. [222] except that i) partial recharges at charging stations are allowed, ii) different charging station technologies can be used at a station (faster charging is more expensive), and iii) the objective is to minimize the charging and battery cycle costs. A compact linear MIP model with Big-M constraints and a simulated annealing approach incorporating local search in several neighborhood structures are proposed.

Goeke and Schneider [117] extend the work by Schneider et al. [222] by considering a mixed fleet with both traditional vehicles and PBEVs in the electric VRPTW. The main contribution of this article is that the energy consumption does not only depend on the distance but involves more parameters, i.e., travel speed, gradient of link, and current load. Here, the energy consumption may also be negative, allowing recuperation and recovery of energy on downward slopes and in breaking events. However, the battery is still fully recharged at a charging station visit. The authors provide a compact MIP model similar to the one in Schneider et al. [222] based on Big-M constraints but including nonlinear parts related to load-dependent energy consumption. Additionally, an Adaptive Large Neighborhood Search algorithm is presented. Tests are performed on newly generated instances and on the Solomon-based instances by Schneider et al. [222]. The authors also consider different objective functions not only involving the traveled distance, but also fuel and battery depreciation costs.

Hiermann et al. [137] tackle the same problem as Schneider et al. [222] but additionally consider a mixed fleet of different PBEVs varying in the load and battery capacity. A compact linear MIP model and an adaptive large neighborhood search are presented to solve this variant.

Desaulniers et al. [80] consider a generalization of the classical VRPTW using only electric vehicles: additional nodes represent charging stations that may be visited an arbitrary number of times. The authors also consider several special variants of this problem: i) at most one charging station can be visited on each route, and ii) at each charging station visit the battery is fully loaded. In the more general variant, there is no limit on the number of visited charging stations and the battery may also be partially loaded at a charging station. The results of these variants are compared, leading to the conclusion that in the unrestricted variant routing costs and the number of needed vehicles can be reduced. The authors present exact branch-price-and-cut approaches based on a classical set-partitioning formulation for the considered problem variants. Much effort is put into the development of efficient solution methods for the pricing subproblem, which often represents a performance bottleneck in these approaches. Mono- and bi-directional labeling algorithms are presented for the different variants, enhanced with acceleration strategies based on ng-route relaxations and reduced graphs. To decrease the integrality gap, two sets of valid inequalities defined on the route variables are added: i) the 2-path cuts, and ii) the subset row inequalities. The presented approaches are tested on a benchmark set introduced in Schneider et al. [222] and generated from the classical Solomon VRPTW instances. All instances can be solved in reasonable time. To the best of our knowledge, these approaches represent the computational state-of-the-art for many variants of the electric VRPTW.

Worley et al. [254] consider a combination of location of charging stations and routing of electric vehicles. They present a MIP model with variables for all route segments (no intermediate depot or charging stations) but do not mention how this model with an exponential number of variables is solved. The objective is to minimize the total costs consisting of the costs for building stations, charging vehicles, and driving.

Table 6.3 gives an overview of the different problem variants discussed in the last two sections.

TABLE 6.3: Classification of the literature related with EV routing problems (SP: shortest path problem, VRP: vehicle routing problem).

Reference	Type	Objective	Energy calculation	Charging	Methodology
[13, 85, 219]	SP	min. energy consumption	predefined	no	exact
[53]	SP	min. travel + charging time	predefined	partial	exact
[12]	SP	min. travel + charging costs	distance	full	exact/heuristic
[88]	VRP	min. distance	distance	constant	exact/heuristic
[243]	VRP	min. travel + vehicle + driver costs	distance	no	heuristic
[222]	VRP	min. distance	distance	full	exact/heuristic
[98]	VRP	min. distance	predefined + load	full	exact
[209]	VRP	min. energy	predefined + load	full	exact
[92]	VRP	min. charging + battery costs	distance	partial	exact/heuristic
[117]	VRP	min. distance/battery costs/energy + driver costs	predefined + load	full	heuristic
[137]	VRP	min. travel + vehicle costs	distance	full	exact/heuristic
[80]	VRP	min. distance	predefined	partial/full	exact
[254]	VRP	min. building + charging + travel costs	distance	full	exact

6.3.4.1 Summary, Open Problems and Possible Research Directions

Most works consider the minimization of the total traveled distance, or more generally the total costs including costs for traveling, fleet investments, battery degradation, etc. Often, the number of vehicles used is minimized in a hierarchical way (in contrast to a weighted objective or a multi-objective formulation). Some authors, however, focus on the minimization of the total energy consumption, which seems to be less relevant for practical needs.

Common for many problem variants is the consideration of customer demands, maximal vehicle load capacities, customer time windows, and clearly the highly restricted battery limits. In more strategic problems, the vehicle fleet is heterogeneous in terms of propulsion type (combustion/electric), battery size (if applicable), and/or load capacity.

Similar to Section 6.3.3, different (more or less detailed) energy consumption models are used. Additionally, for VRP variants it is relevant to also consider the current load for the energy consumption since it may change throughout the tour. The battery limits for PBEVs are considered differently: either simply the tour length is limited or the vehicles are allowed to visit charging stations within the tour. In the second case, different models for charging are implemented: (i) constant charging times, (ii) full charging based on the current state of charge, or (iii) partial charging. Different technologies and therefore charging speeds and capacities may be available at the stations to choose from.

In recent works, the researchers consider more integrated problem variants, e.g., by combining the location of charging stations with the routing part. Here, also the technology, the number of charging points, and the electric capacity may need to be decided for a new charging station.

There are existing models and exact approaches for load-dependent energy consumption. However, there seems to be some room for improvement in terms of model strength and efficiency of solution methods. Also more detailed energy consumption models may be considered in the VRPs, cf. Section 6.3.3.1.

When considering capacities and technologies of charging stations the corresponding electrical grid and its time-dependent load may be considered. In the area of smart energy grids, researchers brought up the idea of using PBEVs as a temporary energy storage to compensate high demands in peak hours Kempton et al. [160]. The integration of such features in existing VRP variants may lead to even more complicated problems but probably would also improve their relevance in real world applications. The combination of the location of charging stations and vehicle routing goes into a similar direction.

6.4 Conclusions

In this chapter, we reviewed the main optimization problems arising in the design and management of car-sharing systems based on electric vehicles. For each problem class, the relevant literature and the main practical issues arising from real-world applications are discussed.

The most relevant research directions for each problem are:

- Location problems (see 6.2.1.5)
  - Simultaneous consideration of different station types (e.g., slow and fast charging stations)
  - Incorporate detailed battery-state modeling in electric location-routing problems
- Relocation of vehicles for multiple-station car-sharing (see 6.3.1.3)
  - Assess users willingness to modify the trip when incentives are offered
  - Investigate on the integration of user-based techniques in staff relocation
  - Use real time information for online relocation
- Electric vehicle shortest path problems (see 6.3.3.1)
  - Use more realistic functions to calculate the vehicle's energy consumption
  - Find system-optimal paths in complex traffic networks rather than optimal paths in an egocentric point of view
- Electric vehicle routing problems (see 6.3.4.1)
  - Use more practically relevant objective functions
  - Use more realistic energy consumption models, e.g., involving the vehicle's load

- Consider the (time-dependent) capacity and load of charging stations and the underlying electrical grid

Besides from tackling each of these problems individually, the study of combined approaches (e.g., simultaneously optimizing the location of charging stations and relocation decisions) is a worthwhile goal for future research.

Many open problems are discussed, indicating Ecar-sharing systems as a rich and promising research area for optimization methods.

Acknowledgements

This research is performed within the European project e4-share (Models for Ecological, Economical, Efficient, Electric Car-Sharing) funded by FFG (Austria) under grant 847350, INNOVIRIS (Belgium) and MIUR (Italy) via the Joint Programme Initiative Urban Europe. See `http://www.univie.ac.at/e4-share/` for more details.

Bibliography

[1] Regulation (EC) no 2150/2002 of the European Parliament and of the council of 25 November 2002 on waste statistics (text with EEA relevance). *Official Journal L 332*, pages 1–36, 2002. URL http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32002R2150&from=en.

[2] Decreto legislativo 3 aprile 2006, n. 152. *Gazzetta Ufficiale*, (88), 2006. URL http://www.albonazionalegestoriambientali.it/Download/it/NormativaNazionale/014-DLGS152_03.04.2006_EstrattoParteIV_Versione_Integrata_2010.pdf.

[3] Abrate, G., Erbetta, F., Fraquelli, G., and Vannoni, D. The costs of disposal and recycling: an application to italian municipal solid waste services. *Regional Studies*, 48(5):896–909, 2014.

[4] Achterberg, T. SCIP: solving constraint integer programs. *Mathematical Programming Computation*, 1(1):1–41, 2009.

[5] Aïvodji, U. M., Gambs, S., Huguet, M.-J., and Killijian, M.-O. Privacy-preserving carpooling. In *Odysseus 2015 - 6th International Workshop on Freight Transportation and Logistics, Ajaccio*, May 2015.

[6] Alizadeh, F. and Goldfarb, D. Second-order cone programming. *Mathematical programming*, 95(1):3–51, 2003.

[7] Almuhtady, A., Lee, S., Romeijn, E., Wynblatt, M., and Ni, J. A degradation-informed battery-swapping policy for fleets of electric or hybrid-electric vehicles. *Transportation Science*, 48(4):609–618, 2014. doi: 10.1287/trsc.2013.0494.

[8] Andersen, E. D. and Andersen, K. D. The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*, pages 197–232. Springer, 2000.

[9] Applegate, D. L., Cook, W. J., Dash, S., and Johnson, D. S. A practical guide to discrete optimization, 2014.

[10] Apt, K. *Principles of constraint programming*. Cambridge University Press, 2003.

[11] Arora, S. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *Journal of the ACM (JACM)*, 45(5):753–782, 1998.

[12] Arslan, O., Yıldız, B., and Karasan, O. E. Minimum cost path problem for plug-in hybrid electric vehicles. Technical report, Bilkent University, Department of Industrial Engineering, Bilkent, Ankara, 2014. URL http://www.optimization-online.org/DB_HTML/2014/02/4230.html.

[13] Artmeier, A., Haselmayr, J., Leucker, M., and Sachenbacher, M. The optimal routing problem in the context of battery-powered electric vehicles. In *Second International Workshop on Constraint Reasoning and Optimization for Computational Sustainability*, Bologna, Italy, 2010.

[14] Asahiro, Y., Horiyama, T., Makino, K., Ono, H., Sakuma, T., and Yamashita, M. How to collect balls moving in the Euclidean plane. *Discrete Applied Mathematics*, 154(16):2247 – 2262, 2006. ISSN 0166-218X. doi: http://dx.doi.org/10.1016/j.dam.2006.04.020. URL http://www.sciencedirect.com/science/article/pii/S0166218X06001995. Discrete Algorithms and Optimization, in Honor of Professor Toshihide Ibaraki at His Retirement from Kyoto UniversityInternational Symposium on Discrete Algorithms and Optimization.

[15] Asamer, J., Reinthaler, M., Ruthmair, M., Straub, M., and Puchinger, J. Optimizing charging station locations for urban taxi providers. *under revision*, 2015.

[16] Atamtürk, A. and Narayanan, V. Conic mixed-integer rounding cuts. *Mathematical Programming*, 122(1):1–20, 2010.

[17] Atamtürk, A., Berenguer, G., and Shen, Z.-J. A conic integer programming approach to stochastic joint location-inventory problems. *Operations Research*, 60(2):366–381, 2012.

[18] Avriel, M. and Williams, A. The value of information and stochastic programming. *Operations Research*, 18(5):947–954, 1970.

[19] Baetz, B. W., Pas, E. I., and Neebe, A. W. Trash management: sizing and timing decisions for incineration and landfill facilities. *Interfaces*, pages 52–61, 1989.

[20] Balinski, M. L. Integer programming: methods, uses, computations. *Management Science*, 12(3):253–313, 1965.

[21] Bandara, N. J., Hettiaratchi, J. P. A., Wirasinghe, S. C., and Pilapiiya, S. Relation of waste generation and composition to socio-economic factors: a case study. *Environmental Monitoring and Assessment*, pages 31–39, 2007.

[22] Baouche, F., Billot, R., Trigui, R., and El Faouzi, N.-E. Efficient allocation of electric vehicles charging stations: Optimization model and application to a dense urban network. *IEEE Intelligent Transportation Systems Magazine*, 6(3): 33–43, 2014. ISSN 1939-1390. doi: 10.1109/mits.2014.2324023.

[23] Barnes, J. W., Wiley, V. D., Moore, J. T., and Ryer, D. M. Solving the aerial fleet refueling problem using group theoretic tabu search. *Mathematical and computer modelling*, 39(6):617–640, 2004.

[24] Barth, M. and Todd, M. Simulation model performance analysis of a multiple station shared vehicle system. *Transportation Research Part C: Emerging Technologies*, 7(4):237–259, August 1999. ISSN 0968-090X. doi: 10.1016/s0968-090x(99) 00021-2.

[25] Barth, M., Todd, M., and Xue, L. User-based vehicle relocation techniques for multiple-station shared-use vehicle systems. In *Transportation Research Board, 80th Annual Meeting*, 2004. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.360.8614.

[26] Beale, E. M. L. On minimizing a convex function subject to linear inequalities. *Journal of the Royal Statistical Society. Series B (Methodological)*, 17(2):173–184, 1955. ISSN 00359246. URL http://www.jstor.org/stable/2983952.

[27] Beale, E. M. L. and Tomlin, J. A. Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables. *OR*, 69(447-454):99, 1970.

[28] Beigl, P., Lebersorger, S., and Salhofer, S. Modelling municipal solid waste generation: A review. *Waste Management*, 28(1):200–214, 2007.

[29] Belloni, A., Lima, A. D. S., Maceira, M. P., and Sagastizábal, C. A. Bundle relaxation and primal recovery in unit commitment problems. The Brazilian case. *Annals of Operations Research*, 120(1-4):21–44, 2003.

[30] Belotti, P. *COUENNE: a user's manual.* URL http://www.coin-or.org/Couenne/couenne-user-manual.pdf.

[31] Belotti, P., Kirches, C., Leyffer, S., Linderoth, J., Luedtke, J., and Mahajan, A. Mixed-integer nonlinear optimization. *Acta Numerica*, 22:1–131, 5 2013. ISSN 1474-0508. doi: 10.1017/S0962492913000032. URL http://journals.cambridge.org/article_S0962492913000032.

[32] Ben-Tal, A. and Nemirovski, A. On polyhedral approximations of the second-order cone. *Mathematics of Operations Research*, 26(2):193–205, 2001.

[33] Benchimol, P., Hoeve, W.-J., Régin, J.-C., Rousseau, L.-M., and Rueher, M. Improved filtering for weighted circuit constraints. *Constraints*, 17(3):205–233,

2012. ISSN 1383-7133. doi: 10.1007/s10601-012-9119-x. URL http://dx.doi.org/10.1007/s10601-012-9119-x.

[34] Benders, J. F. Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 4(1):238–252, 1962.

[35] Benson, H. Y. and Saglam, U. Mixed-integer second-order cone programming: A survey. *Tutorials in Operations Research*, pages 13–36, 2013.

[36] Bertsekas, D. P. *Nonlinear Programming*. Athena Scientific, 2nd edition, Sept. 1999. ISBN 1886529000.

[37] Birge, J. R. The value of the stochastic solution in stochastic linear programs with fixed recourse. *Mathematical programming*, 24(1):314–325, 1982.

[38] Birge, J. R. and Louveaux, F. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

[39] Bit-Monnot, A., Artigues, C., Huguet, M.-J., and Killijian, M.-O. Carpooling: the 2 Synchronization Points Shortest Paths Problem. In Frigioni, D. and Stiller, S., editors, *13th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, volume 33 of *OpenAccess Series in Informatics (OASIcs)*, pages 150–163, Dagstuhl, Germany, 2013. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[40] Bonami, P. and Lejeune, M. A. An exact solution approach for portfolio optimization problems under stochastic and integer constraints. *Operations research*, 57(3):650–670, 2009.

[41] Bortoff, S. A. Path planning for UAVs. In *American Control Conference, 2000. Proceedings of the 2000*, volume 1, pages 364–368, Sep 2000. doi: 10.1109/ACC.2000.878915.

[42] Boschetti, M. and Maniezzo, V. Benders decomposition, Lagrangean Relaxation and metaheuristic design. *Journal of Heuristics*, 15(3):283–312, 2007. ISSN 1572-9397. doi: 10.1007/s10732-007-9064-9. URL http://dx.doi.org/10.1007/s10732-007-9064-9.

[43] Boyacı, B., Zografos, K. G., and Geroliminis, N. An optimization framework for the development of efficient one-way car-sharing systems. *European Journal of Operational Research*, 240(3):718–733, February 2015. ISSN 0377-2217. doi: 10.1016/j.ejor.2014.07.020.

[44] Boyd, S. and Mutapcic, A. Subgradient methods. *Lecture notes of EE364b, Stanford University, Winter Quarter*, 2007, 2006.

[45] Boyd, S. and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

[46] Brandenberg, R. and Roth, L. New algorithms for k-center and extensions. In *Combinatorial Optimization and Applications*, pages 64–78. Springer, 2008.

[47] Brandstätter, G., Gambella, C., Leitner, M., Malaguti, E., Masini, F., Puchinger, J., Ruthmair, M., and Vigo, D. Overview of optimization problems in electric car-sharing system design and management. In Doerner, K. and Feichtinger, G., editors, *Dynamic Perspectives on Managerial Decision Making*. Springer, Berlin, 2016. To appear.

[48] Bruck, B. P., Incerti, V., Iori, M., and Vignoli, M. On the development of a practical carpooling application. In *Odysseus 2015 - 6th International Workshop on Freight Transportation and Logistics, Ajaccio*, May 2015.

[49] Bruglieri, M., Colorni, A., and Luè, A. The vehicle relocation problem for the one-way electric vehicle sharing: An application to the Milan case. *Procedia - Social and Behavioral Sciences*, 111:18–27, February 2014. ISSN 1877-0428. doi: 10.1016/j.sbspro.2014.01.034.

[50] Burer, S. and Letchford, A. N. Non-convex mixed-integer nonlinear programming: A survey. *Surveys in Operations Research and Management Science*, 17 (2):97 – 106, 2012. ISSN 1876-7354. doi: http://dx.doi.org/10.1016/j.sorms.2012.08.001. URL http://www.sciencedirect.com/science/article/pii/S1876735412000037.

[51] Callan, S. J. and Thomas, J. M. Economies of scale and scope: A cost analysis of municipal solid waste services. *Land Economics*, 77(4):548–560, 2001.

[52] Caprara, A., Monaci, M., Toth, P., and Guida, P. L. A Lagrangian heuristic algorithm for a real-world train timetabling problem. *Discrete Applied Mathematics*, 154(5):738 – 753, 2006. ISSN 0166-218X. doi: http://dx.doi.org/10.1016/j.dam.2005.05.026. URL http://www.sciencedirect.com/science/article/pii/S0166218X05003045.

[53] Cassandras, C. G., Wang, T., and Pourazarm, S. Energy-aware vehicle routing in networks with charging nodes. Technical report, Division of Systems Engineering and Center for Information and Systems Engineering, Boston University, January 2014. URL http://arxiv.org/abs/1401.6478.

[54] Cavadas, J., Correia, G. H., and Gouveia, J. A MIP model for locating slow-charging stations for electric vehicles in urban areas accounting for driver tours. *Transportation Research Part E: Logistics and Transportation Review*, 75:188–201, March 2015. ISSN 1366-5545. doi: 10.1016/j.tre.2014.11.005.

[55] Cepolina, E. M. and Farina, A. A new shared vehicle system for urban areas. *Transportation Research Part C: Emerging Technologies*, 21(1):230–243, April 2012. ISSN 0968-090X. doi: 10.1016/j.trc.2011.10.005.

[56] Çezik, M. T. and Iyengar, G. Cuts for mixed 0-1 conic programming. *Mathematical Programming*, 104(1):179–202, 2005.

[57] Chandrasekaran, S., Golub, G., Gu, M., and Sayed, A. Parameter estimation in the presence of bounded data uncertainties. *SIAM Journal on Matrix Analysis and Applications*, 19(1):235–252, 1998.

[58] Charnes, A. and Cooper, W. W. Chance-constrained programming. *Management science*, 6(1):73–79, 1959.

[59] Chen, C. and Hua, G. Optimal deployment of electric vehicle charging and battery swapping stations based on gas station network. *International Journal of Control and Automation*, 7(5):247–258, 2014. URL http://www.paper.edu.cn/en_releasepaper/downPaper/201401-450.

[60] Chen, T. D., Kockelman, K. M., Murray, W. J., and Khan, M. The electric vehicle charging station location problem: a parking-based assignment method for seattle. In *92nd Annual Meeting of the Transportation Research Board. Washington DC, USA*, 2013. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.300.2883.

[61] Cheng, Y., Drewes, S., Philipp, A., and Pesavento, M. Joint network optimization and beamforming for coordinated multi-point transmission using mixed integer programming. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 3217–3220. IEEE, 2012.

[62] Choubey, N. S. Moving target travelling salesman problem using genetic algorithm. *International Journal of Computer Applications*, 70(2):30–34, may 2013. doi: 10.5120/11937-7726. URL http://dx.doi.org/10.5120/11937-7726.

[63] Clarke, G. and Wright, J. W. Scheduling of vehicles from a central depot to a number of delivery points. *Operations research*, 12(4):568–581, 1964.

[64] Clemente, M., Fanti, M. P., Mangini, A. M., and Ukovich, W. The vehicle relocation problem in car sharing systems: Modeling and simulation in a Petri Net framework. *Lecture Notes in Computer Science*, 7927:250–269, 2013. ISSN 1611-3349. doi: 10.1007/978-3-642-38697-8_14.

[65] Coelho, L. C. and Laporte, G. The exact solution of several classes of inventory-routing problems. *Computers & Operations Research*, 40(2):558–565, 2013.

[66] Commission, E. Environment: Commission urges Italy to resolve Campania waste management situation. *Press Release*, (IP/11/1102), 2011.

[67] Conic Programming Solvers. Conic programming solvers cited in gamsworld.org. URL http://www.gamsworld.org/cone/solvers.htm.

[68] Correia, G. H. and Antunes, A. P. Optimization approach to depot location and trip selection in one-way carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):233–247, January 2012. ISSN 1366-5545. doi: 10.1016/j.tre.2011.06.003.

[69] Correia, G. H., Jorge, D. R., and Antunes, D. M. The added value of accounting for users' flexibility and information on the potential of a station-based one-way car-sharing system: An application in Lisbon, Portugal. *Journal of Intelligent Transportation Systems*, 18(3):299–308, June 2014. ISSN 1547-2442. doi: 10.1080/15472450.2013.836928.

[70] Cplex Manual. *IBM ILOG CPLEX Optimization Studio CPLEX User's Manual*. URL http://public.dhe.ibm.com/software/products/Decision_Optimization/docs/pdf/usrcplex.pdf. Version 12 Release 6.

[71] Crainic, T. G., Frangioni, A., and Gendron, B. Bundle-based relaxation methods for multicommodity capacitated fixed charge network design. *Discrete Applied Mathematics*, 112(1–3):73 – 99, 2001. ISSN 0166-218X. doi: http://dx.doi.org/10.1016/S0166-218X(00)00310-3. URL http://www.sciencedirect.com/science/article/pii/S0166218X00003103. Combinatorial Optimization Symposium, Selected Papers.

[72] Croxton, K. L., Gendron, B., and Magnanti, T. L. A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science*, 49(9):1268–1273, 2003.

[73] Dantzig, G., Fulkerson, R., and Johnson, S. Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, 2(4): 393–410, 1954.

[74] Dantzig, G. B. Linear programming under uncertainty. *Management science*, 1 (3-4):197–206, 1955.

[75] Dantzig, G. B. *Linear programming and extensions*. Princeton University Press, 1998.

[76] Dantzig, G. B. and Ramser, J. H. The truck dispatching problem. *Management Science*, 6(1):80–91, 1959.

[77] Dantzig, G. B. and Wolfe, P. Decomposition principle for linear programs. *Operations research*, 8(1):101–111, 1960.

[78] Daskalopoulos, E., Badr, O., and Probert, S. D. Municipal solid waste: a prediction methodology for the generation rate and composition in the European Union countries and the United States of America. *Resources, Conservation and Recycling*, 24(2):155–166, 1998.

[79] Desaulniers, G., Desrosiers, J., and Solomon, M. M. *Column generation*, volume 5. Springer Science & Business Media, 2006.

[80] Desaulniers, G., Errico, F., Irnich, S., and Schneider, M. Exact algorithms for electric vehicle-routing problems with time windows. Technical report, Darmstadt Technical University, 2014.

[81] Drewes, S. and Ulbrich, S. *Mixed integer second order cone programming*. PhD thesis, Technische Universität Darmstadt, 2009. Verlag Dr. Hut.

[82] Drexl, M. and Schneider, M. A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241(2):283–308, 2015. ISSN 0377-2217. doi: 10.1016/j.ejor.2014.08.030.

[83] Duran, M. A. and Grossmann, I. E. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical programming*, 36(3): 307–339, 1986.

[84] Dyson, B. and Chang, N.-B. Forecasting municipal solid waste generation in a fast-growing urban region with system dynamics modeling. *Waste Management*, 25(7):669–679, 2005.

[85] Eisner, J., Funke, S., and Storandt, S. Optimal route planning for electric vehicles in large networks. In *25th AAAI Conference on Artificial Intelligence*, 2011. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3637.

[86] El Ghaoui, L. and Lebret, H. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 18(4):1035–1064, 1997.

[87] Elhedhli, S., Li, L., Gzara, M., and Naoum-Sawaya, J. A branch-and-price algorithm for the bin packing problem with conflicts. *INFORMS Journal on Computing*, 23(3):404–415, 2011.

[88] Erdoğan, S. and Miller-Hooks, E. A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):100–114, January 2012. ISSN 1366-5545. doi: 10.1016/j.tre.2011.08.001.

[89] Escudero, L. F., Garín, A., Merino, M., and Pérez, G. The value of the stochastic solution in multistage problems. *Top*, 15(1):48–64, 2007.

[90] Farquhar, G. Leachate: production and characterization. *Canadian Journal of Civil Engineering*, 16(3):317–325, 1989.

[91] Fassi, A. E., Awasthi, A., and Viviani, M. Evaluation of carsharing network's growth strategies through discrete event simulation. *Expert Systems with Applications*, 39(8):6692–6705, June 2012. ISSN 0957-4174. doi: 10.1016/j.eswa.2011.11.071.

[92] Felipe, Á., Ortuño, M. T., Righini, G., and Tirado, G. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transportation Research Part E: Logistics and Transportation Review*, 71(0):111–128, 2014. ISSN 1366-5545. doi: 10.1016/j.tre.2014.09.003.

[93] Fischetti, M., González, J. J. S., and Toth, P. Experiments with a multi-commodity formulation for the symmetric capacitated vehicle routing problem. In *Proceedings of the 3rd Meeting of the EURO Working Group on Transportation, 169–173*. Citeseer, 1995.

[94] Fisher, M. L. The Lagrangian Relaxation method for solving integer programming problems. *Management science*, 50(12_supplement):1861–1871, 2004.

[95] Fourer, R., Gay, D. M., and Kernighan, B. W. A modeling language for mathematical programming. *Management Science*, 36(5):519–554, 1990.

[96] Frade, I., Ribeiro, A., Gonçalves, G., and Antunes, A. P. Optimal location of charging stations for electric vehicles in a neighborhood in Lisbon, Portugal. *Transportation Research Record: Journal of the Transportation Research Board*, 2252:91–98, December 2011. ISSN 0361-1981. doi: 10.3141/2252-12.

[97] Frangioni, A. About Lagrangian methods in integer optimization. *Annals of Operations Research*, 139(1):163–193, 2005. doi: 10.1007/s10479-005-3447-9. URL http://dx.doi.org/10.1007/s10479-005-3447-9.

[98] Frank, S., Preis, H., and Nachtigall, K. On the modeling of recharging stops in context of vehicle routing problems. In Huisman, D., Louwerse, I., and Wagelmans, A. P., editors, *Operations Research Proceedings 2013*, Operations Research Proceedings, page 129–135. Springer Science + Business Media, 2014. doi: 10.1007/978-3-319-07001-8_18.

[99] Gambella, C., Lodi, A., and Vigo, D. Exact solutions of the carrier-vehicle traveling salesman problem. Technical Report OR-15-10, DEI - Università di Bologna, 2015.

[100] Gambella, C., Ghaddar, B., and Naoum-Sawaya, J. The interceptor vehicle routing problem: formulation and branch & price algorithm. Technical Report OR-16-2, DEI - Università di Bologna, 2016.

[101] Gambella, C., Maggioni, F., and Vigo, D. A solid waste management problem with stochastic parameters at a tactical planning level. Technical Report OR-16-3, DEI - Università di Bologna, 2016.

[102] Gambella, C., Parriani, T., Pozzi, M., Vigo, D., Gordini, A., Ravaglia, L., Lombardi, F., Guarnieri, A., and Fabrizio, S. Waste flow optimization: An application in the italian contest. Technical Report OR-16-1, DEI - Università di Bologna, 2016.

[103] Garone, E., Naldi, R., Casavola, A., and Frazzoli, E. Planning algorithms for a class of heterogenous multi-vehicle systems. In *Proceedings of the 8th IFAC Symposium on Nonlinear Control Systems*, pages 969–974. Univ. of Bologna, Italy, 2010.

[104] Garone, E., Naldi, R., and Casavola, A. Traveling salesman problem for a class of carrier-vehicle systems. *Journal of Guidance, Control, and Dynamics*, 34(4): 1272–1276, 2011.

[105] Garone, E., Determe, J.-F., and Naldi, R. A travelling salesman problem for a class of heterogeneous multi-vehicle systems. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 1166–1171, Dec 2012. doi: 10.1109/CDC.2012.6426488.

[106] Garone, E., Determe, J.-F., and Naldi, R. Generalized traveling salesman problem for carrier-vehicle systems. *Journal of guidance, control, and dynamics*, 37 (3):766–774, 2014.

[107] Ge, S., Feng, L., and Liu, H. The planning of electric vehicle charging station based on grid partition method. In *2011 International Conference on Electrical and Control Engineering (ICECE)*. IEEE, September 2011. ISBN http://id.crossref.org/isbn/978-1-4244-8162-0. doi: 10.1109/iceceng.2011.6057636.

[108] Geoffrion, A. M. Generalized Benders decomposition. *Journal of optimization theory and applications*, 10(4):237–260, 1972.

[109] Geoffrion, A. M. Lagrangean relaxation for integer programming. *Mathematical Programming Study*, 2:82–114, 1974.

[110] Ghaddar, B., Naoum-Sawaya, J., Kishimoto, A., Taheri, N., and Eck, B. A Lagrangian Decomposition approach for the pump scheduling problem in water networks. *European Journal of Operational Research*, 241(2):490 – 501, 2015. ISSN 0377-2217. doi: http://dx.doi.org/10.1016/j.ejor.2014.08.033. URL http://www.sciencedirect.com/science/article/pii/S0377221714007103.

[111] Ghiani, G., Laporte, G., and Musmanno, R. *Introduction to logistics systems planning and control*. John Wiley & Sons, 2004.

[112] Ghiani, G., Guerrieroa, E., Mannia, A., Mannia, E., and Potenza, A. Simultaneous personnel and vehicle shift scheduling in the waste management sector. *Waste management*, 33(7):1589–1594, 2013.

[113] Ghiani, G., Laganà, D., Manni, E., Musmanno, R., and Vigo, D. Operations research in solid waste management: A survey of strategic and tactical issues. *Computers & Operations Research*, 44:22–32, 2014.

[114] Ghiani, G., Manni, A., Manni, E., and Toraldo, M. The impact of an efficient collection sites location on the zoning phase in municipal solid waste management. *Waste Management*, 34(11):1949–1956, 2014.

[115] Giugliano, M., Cernuschi, S., Grosso, M., and Rigamonti, L. Material and energy recovery in integrated waste management systems. An evaluation based on life cycle assessment. *Waste management*, 31(9):2092–2101, 2011.

[116] Glineur, F. Computational experiments with a linear approximation of second-order cone optimization. 2000.

[117] Goeke, D. and Schneider, M. Routing a mixed fleet of electric and conventional vehicles. *European Journal of Operational Research*, 245(1):81–99, 2015.

[118] Goldfarb, D. The simplex method for conic programming. *CORC Report*, 5, 2002.

[119] Goldfarb, D. and Yin, W. Second-order cone programming methods for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 27 (2):622–645, 2005.

[120] Gonzaga, C. C. Path-following methods for linear programming. *SIAM review*, 34(2):167–224, 1992.

[121] González, J., Alvaro, R., Gamallo, C., Fuentes, M., Fraile-Ardanuy, J., Knapen, L., and Janssens, D. Determining electric vehicle charging point locations considering drivers' daily activities. *Procedia Computer Science*, 32(0):647–654, 2014. ISSN 1877-0509. doi: 10.1016/j.procs.2014.05.472. URL http://www.sciencedirect.com/science/article/pii/S1877050914006723. The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014), the 4th International Conference on Sustainable Energy Information Technology (SEIT-2014).

[122] Google Optimization Tools. URL https://developers.google.com/optimization/.

[123] Griggs, B. J., Parnell, G. S., and Lehmkuhl, L. J. An air mission planning algorithm using decision analysis and mixed integer programming. *Operations Research*, 45(5):662–676, 1997.

[124] Guignard, M. Lagrangean relaxation. *Top*, 11(2):151–200, 2003. ISSN 1863-8279. doi: 10.1007/BF02579036. URL http://dx.doi.org/10.1007/BF02579036.

[125] Guignard, M. and Kim, S. Lagrangean decomposition: A model yielding stronger Lagrangean bounds. *Mathematical Programming*, 39(2):215–228, 1987. ISSN 1436-4646. doi: 10.1007/BF02592954. URL http://dx.doi.org/10.1007/BF02592954.

[126] Guignard, M. and Kim, S. Lagrangean decomposition for integer programming: theory and applications. *Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle*, 21(4):307–323, 1987.

[127] Guo, P., Huang, G. H., and He, L. ISMISIP: an inexact stochastic mixed integer linear semi-infinite approach for solid waste management and planning under uncertainty. *Journal of Air & Waste Management Association*, 2008.

[128] Gurobi Manual. *GUROBI OPTIMIZER REFERENCE MANUAL*. URL https://www.gurobi.com/documentation/6.5/refman.pdf. Version 6.5.

[129] Gutin, G. and Punnen, A. P. *The traveling salesman problem and its variations*, volume 12. Springer Science & Business Media, 2002.

[130] Hammar and Nilsson. Approximation results for kinetic variants of TSP. *Discrete & Computational Geometry*, 27(4):635–651, 2002. ISSN 0179-5376. doi: 10.1007/s00454-001-0081-4. URL http://dx.doi.org/10.1007/s00454-001-0081-4.

[131] Heilmann, W.-R. A note on chance-constrained programming. *Journal of the Operational Research Society*, pages 533–537, 1983.

[132] Held, M. and Karp, R. M. The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18(6):1138–1162, 1970.

[133] Held, M. and Karp, R. M. The traveling-salesman problem and minimum spanning trees: Part ii. *Mathematical Programming*, 1(1):6–25, 1971. ISSN 1436-4646. doi: 10.1007/BF01584070. URL http://dx.doi.org/10.1007/BF01584070.

[134] Helvig, C. S., Robins, G., and Zelikovsky, A. The moving-target traveling salesman problem. *Journal of Algorithms*, 49(1):153–174, 2003.

[135] Hera, G. La gestione dei rifiuti del gruppo Hera. 2008.

[136] Hess, A., Malandrino, F., Reinhardt, M. B., Casetti, C., Hummel, K. A., and Barceló-Ordinas, J. M. Optimal deployment of charging stations for electric vehicular networks. In *Proceedings of the First Workshop on Urban Networking*, UrbaNe '12, pages 1–6, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1781-8. doi: 10.1145/2413236.2413238.

[137] Hiermann, G., Puchinger, J., and Hartl, R. F. The electric fleet size and mix vehicle routing problem with time windows and recharging stations. Technical report, Austrian Institute of Technology, 2014.

[138] Hijazi, H., Bonami, P., and Ouorou, A. Robust delay-constrained routing in telecommunications. *Annals of Operations Research*, 206(1):163–181, 2013.

[139] Holmberg, K. and Yuan, D. A Lagrangian heuristic based branch-and-bound approach for the capacitated network design problem. *Operations Research*, 48

(3):461–481, 2000. ISSN 0030364X, 15265463. URL http://www.jstor.org/stable/222850.

[140] Hosni, H., Naoum-Sawaya, J., and Artail, H. The shared-taxi problem: Formulation and solution methods. *Transportation Research Part B: Methodological*, 70:303–318, 2014.

[141] Høyland, K. and Wallace, S. W. Generating scenario trees for multistage decision problems. *Management Science*, 47(2):295–307, 2001.

[142] Huang, G., Baetz, B. W., and Patry, G. G. A grey linear programming approach for municipal solid waste management planning under uncertainty. *Civil Engineering Systems*, 9(4):319–335, 1992.

[143] Huang, G. H., Baetz, B. W., and Patry, G. G. Trash-flow allocation: planning under uncertainty. *Interfaces*, 28(6):36–55, 1998.

[144] Hurkens, C. A. and Woeginger, G. J. On the nearest neighbor rule for the traveling salesman problem. *Operations Research Letters*, 32(1):1 – 4, 2004. ISSN 0167-6377. doi: http://dx.doi.org/10.1016/S0167-6377(03)00093-2. URL http://www.sciencedirect.com/science/article/pii/S0167637703000932.

[145] Ibrahim, M. A., Göransson, G., Kaczala, F., Hogland, W., and Marques, M. Characterization of municipal solid waste temporary storage sites: Risks posed to surrounding areas as a consequence of fire incidents. *Waste management*, 33 (11):2296–2306, 2013.

[146] ISPRA. Special Waste Report - Edition 2012. *ISPRA Reports*, 202, 2014.

[147] ISPRA. Municipal Waste Report-Edition 2014. *ISPRA Reports*, 202, 2014.

[148] Jiang, Q., Sarker, R., and Abbass, H. Tracking moving targets in the non-stationary travelling salesman problem. *Complexity International*, pages 171–179, 2005.

[149] Jorge, D. and Correia, G. H. Carsharing systems demand estimation and defined operations: a literature review. *European Journal of Transport and Infrastructure Research*, 13(3):201–220, 2013. URL http://www.ejtir.tudelft.nl/issues/2013_03/pdf/2013_03_02.pdf.

[150] Jorge, D., Correia, G. H., and Barnhart, C. Comparing optimal relocation operations with simulated relocation policies in one-way carsharing systems. *IEEE Transactions on Intelligent Transportation Systems*, 15(4):1667–1675, August 2014. ISSN 1558-0016. doi: 10.1109/tits.2014.2304358.

[151] Kall, P. and Wallace, S. W. *Stochastic Programming*. Chichester, 1994.

[152] Karimi, B., Fatemi Ghomi, S. M. T., and Wilson, J. M. The capacitated lot sizing problem: a review of models and algorithms. *Omega*, 31(5):365–378, 2003.

[153] Kaut, M. *Scenario tree generation for stochastic programming: Cases from finance*. PhD thesis, Norwegian University of Science and Technology, 2003.

[154] Kaut, M. and Wallace, S. W. Evaluation of scenario-generation methods for stochastic programming. *Pacific Journal of Optimization*, 3(2):257–271, 2007.

[155] Kaut, M., Vladimirou, H., Wallace, S. W., and Zenios, S. A. Stability analysis of portfolio management with conditional value-at-risk. *Quantitative Finance*, 7 (4):397–409, 2007.

[156] Kaya, L. G. and Hooker, J. N. A filter for the circuit constraint. In *Principles and Practice of Constraint Programming-CP 2006*, pages 706–710. Springer, 2006.

[157] Kek, A. G., Cheu, R. L., and Chor, M. Relocation simulation model for multiple-station shared-use vehicle systems. *Transportation Research Record*, 1986(1): 81–88, January 2006. ISSN 0361-1981. doi: 10.3141/1986-13.

[158] Kek, A. G., Cheu, R. L., Meng, Q., and Fung, C. H. A decision support system for vehicle relocation operations in carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 45(1):149–158, January 2009. ISSN 1366-5545. doi: 10.1016/j.tre.2008.02.008.

[159] Kelley Jr., J. E. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8(4):703–712, 1960.

[160] Kempton, W., Tomic, J., Letendre, S., Brooks, A., and Lipman, T. Vehicle-to-grid power: battery, hybrid, and fuel cell vehicles as resources for distributed electric power in california. Technical Report UCD-ITS-RR-01-03, University of Delaware, 2001.

[161] Kitamura, R. Sharing electric vehicles in kyoto: Kyoto public car system. {*IATSS*} *Research*, 26(1):86–89, 2002. ISSN 0386-1112. doi: 10.1016/S0386-1112(14)60085-6. URL http://www.sciencedirect.com/science/article/pii/S0386111214600856.

[162] Kjeldsen, P., Barlaz, M. A., Rooker, A. P., Baun, A., Ledin, A., and Christensen, T. H. Present and long-term composition of MSW landfill leachate: a review. *Critical reviews in environmental science and technology*, 32(4):297–336, 2002.

[163] Klauco, M., Blazek, S., Kvasnica, M., and Fikar, M. Mixed-integer SOCP formulation of the path planning problem for heterogeneous multi-vehicle systems. In *Control Conference (ECC), 2014 European*, pages 1474–1479. IEEE, 2014.

[164] Kohl, N. and Madsen, O. B. G. An optimization algorithm for the vehicle routing problem with time windows based on Lagrangian Relaxation. *Operations Research*, 45(3):395–406, 1997.

[165] Kulikowska, D. and Klimiuk, E. The effect of landfill age on municipal leachate composition. *Bioresource Technology*, 99(13):5981–5985, 2008.

[166] Land, A. H. and Doig, A. G. An automatic method of solving discrete programming problems. *Econometrica*, 28(3):497–520, 1960. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1910129.

[167] Laporte, G. The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(2):231–247, 1992.

[168] Lawler, E. L. A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management Science*, 18(7):401–405, 1972.

[169] Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., and Shmoys, D. B., editors. *The Traveling Salesman Problem. A Guided Tour of Combinatorial Optimization.* Wiley-Interscience Series in Discrete Mathematics. Wiley, Chichester, 1985.

[170] Lee, J. and Park, G.-L. Planning of relocation staff operations in electric vehicle sharing systems. In *Lecture Notes in Computer Science*, volume 7803, page 256–265. Springer Science + Business Media, 2013. ISBN http://id.crossref.org/isbn/978-3-642-36543-0. doi: 10.1007/978-3-642-36543-0_27.

[171] Létocart, L., Nagih, A., and Touati-Moungla, N. Dantzig-Wolfe and Lagrangian decompositions in integer linear programming. *International Journal of Mathematics in Operational Research*, 4(3):247–262, 2012.

[172] Li, J.-Q. Transit bus scheduling with limited energy. *Transportation Science*, 48 (4):521–539, 2014. doi: 10.1287/trsc.2013.0468.

[173] Li, Y. and Huang, G. An inexact two-stage mixed integer linear programming method for solid waste management in the city of regina. *Journal of Environmental Management*, 81(3):188 – 209, 2006. ISSN 0301-4797. doi: http://dx.doi.org/10.1016/j.jenvman.2005.10.007.

[174] Li, Y., Huang, G. H., Nie, X., and Nie, S. A two-stage fuzzy robust integer programming approach for capacity planning of environmental management systems. *European Journal of Operational Research*, 189(2):399–420, 2008.

[175] Li, Y. P. and Huang, G. H. Minimax regret analysis for municipal solid waste management: an interval-stochastic programming approach. *Journal of Air & Waste Management Association*, 2006.

[176] Lin, C., Choy, K., Ho, G., Chung, S., and Lam, H. Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, 41 (4, Part 1):1118–1138, 2014. ISSN 0957-4174. doi: 10.1016/j.eswa.2013.07.107.

[177] Liu, S. and Lin, Y. *Grey information: theory and practical applications*. Springer Science & Business Media, 2006.

[178] Lobo, M. S., Vandenberghe, L., Boyd, S., and Lebret, H. Applications of second-order cone programming. *Linear algebra and its applications*, 284(1):193–228, 1998.

[179] Louveaux, F. V. A solution method for multistage stochastic programs with recourse with application to an energy investment problem. *Operations Research*, 28(4):889–902, 1980.

[180] Louveaux, F. V. and Peeters, D. A dual-based procedure for stochastic facility location. *Operations research*, 40(3):564–573, 1992.

[181] Luè, A., Colorni, A., Nocerino, R., and Paruscio, V. Green move: An innovative electric vehicle-sharing system. *Procedia - Social and Behavioral Sciences*, 48(0):2978–2987, 2012. ISSN 1877-0428. doi: 10.1016/j.sbspro.2012.06.1265. URL http://www.sciencedirect.com/science/article/pii/S187704281203008X. Transport Research Arena 2012.

[182] Maggioni, F., Potra, F., Bertocchi, M., and Allevi, E. Stochastic second-order cone programming in mobile ad hoc networks. *Journal of Optimization Theory and Applications*, 143(2):309–328, 2009. ISSN 0022-3239.

[183] Maggioni, F., Bertocchi, M., Giacometti, R., Vespucci, M. T., Innorta, M., and Allevi, E. A stochastic optimization model for gas retail with temperature scenarios and oil price parameters. *IMA Journal of Management Mathematics*, 21(2):149–163, 2010. doi: 10.1093/imaman/dpp011. URL http://imaman.oxfordjournals.org/content/21/2/149.abstract.

[184] Maggioni, F., Allevi, E., and Bertocchi, M. Bounds in multistage linear stochastic programming. *Journal of Optimization Theory and Applications*, 163(1):200–229, 2013.

[185] Mak, H.-Y., Rong, Y., and Shen, Z.-J. M. Infrastructure planning for electric vehicles with battery swapping. *Management Science*, 59(7):1557–1575, 2013. doi: 10.1287/mnsc.1120.1672. URL http://dx.doi.org/10.1287/mnsc.1120.1672.

[186] Mallick, M., Vo, B.-N., Kirubarajan, T., and Arulampalam, S. Introduction to the issue on multitarget tracking. *IEEE Journal of Selected Topics in Signal Processing*, 7(3):373–375, June 2013. doi: 10.1109/JSTSP.2013.2254034.

[187] Maqsood, I. and Huang, G. H. Two-stage interval-stochastic programming model for waste management under uncertainty. *Journal of the Air & Waste Management Association*, 2003.

[188] Mata-Alvarez, J., Mace, S., and Llabres, P. Anaerobic digestion of organic solid wastes. An overview of research achievements and perspectives. *Bioresource technology*, 74(1):3–16, 2000.

[189] Mazzanti, M. and Zoboli, R. Municipal waste Kuznets curves: Evidence on socio-economic drivers and policy effectiveness from the EU. *Environmental and Resource Economics*, 44(2):203–230, 2009.

[190] Millard-Ball, A., Murray, G., Ter Schure, J., Fox, C., and Burkhardt, J. Carsharing: Where and how it succeeds. Technical Report TCRP Report 108, TRB, Washington D.C., 2005.

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

[192] Misener, R. and Floudas, C. A. ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations. *Journal of Global Optimization*, 2014. doi: 10.1007/s10898-014-0166-2.

[193] Mittelmann, H. D. The state-of-the-art in conic optimization software. In *Handbook on Semidefinite, Conic and Polynomial Optimization*, pages 671–686. Springer, 2012.

[194] Murray, R. M. Recent research in cooperative control of multivehicle systems. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):571–583, 2007.

[195] Nagy, G. and Salhi, S. Location-routing: Issues, models and methods. *European Journal of Operational Research*, 177(2):649–672, 2007. ISSN 0377-2217. doi: 10.1016/j.ejor.2006.04.004.

[196] Nair, R. and Miller-Hooks, E. Fleet management for vehicle sharing operations. *Transportation Science*, 45(4):524–540, November 2011. ISSN 1526-5447. doi: 10.1287/trsc.1100.0347.

[197] Nakayama, S., Yamamoto, T., and Kitamura, R. Simulation analysis for the management of an electric vehicle-sharing system: Case of the Kyoto public-car system. *Transportation Research Record*, 1791(1):99–104, January 2002. ISSN 0361-1981. doi: 10.3141/1791-15.

[198] Nemhauser, G. L. and Wolsey, L. A. *Integer and Combinatorial Optimization*. John Wiley and Sons, New York, 1988.

[199] Nemirovski, A. Advances in convex optimization: conic programming. In *Proceedings of the International Congress of Mathematicians Madrid, August 22–30, 2006*, pages 413–444, 2007.

[200] Nesterov, Y. and Nemirovski, A. *Interior-point polynomial algorithms in convex programming*, volume 13. SIAM, 1994.

[201] Noël, M.-C. and Smeers, Y. Nested decomposition of multistage nonlinear programs with recourse. *Mathematical Programming*, 37(2):131–152, 1987.

[202] Nowak, I. *Relaxation and decomposition methods for mixed integer nonlinear programming*, volume 152. Springer Science & Business Media, 2006.

[203] Papadimitriou, C. H. and Steiglitz, K. *Combinatorial optimization: algorithms and complexity.* Courier Corporation, 1982.

[204] Pelletier, S., Jabali, O., and Laporte, G. Battery electric vehicles for goods distribution: A survey of vehicle technology, market penetration, incentives and practices. Technical report, CIRRELT, Montréal, Canada, 2014.

[205] Pelletier, S., Jabali, O., and Laporte, G. Goods distribution with electric vehicles: Review and research perspectives. *Transportation Science*, 2015. accepted.

[206] Pereira, M. and Pinto, L. Stochastic optimization of a multireservoir hydroelectric system: a decomposition approach. *Water resources research*, 21(6):779–792, 1985.

[207] Pflug, G. C. Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical programming*, 89(2):251–271, 2001.

[208] Pınar, M. Ç. Mixed-integer second-order cone programming for lower hedging of american contingent claims in incomplete markets. *Optimization Letters*, 7(1): 63–78, 2013.

[209] Preis, H., Frank, S., and Nachtigall, K. Energy-optimized routing of electric vehicles in urban delivery systems. In Helber, S., Breitner, M., Rösch, D., Schön, C., Graf von der Schulenburg, J.-M., Sibbertsen, P., Steinbach, M., Weber, S., and Wolter, A., editors, *Operations Research Proceedings 2012*, Operations Research Proceedings, page 583–588. Springer Science + Business Media, November 2013. doi: 10.1007/978-3-319-00795-3_87.

[210] Prekopa, A. On probabilistic constrained programming. In *Proceedings of the Princeton symposium on mathematical programming*, pages 113–138. Princeton University Press Princeton, NJ, 1970.

[211] Prodhon, C. and Prins, C. A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1):1–17, 2014. ISSN 0377-2217. doi: 10.1016/j.ejor.2014.01.005.

[212] Quesada, I. and Grossmann, I. E. An LP/NLP based branch and bound algorithm for convex MINLP optimization problems. *Computers & chemical engineering*, 16(10):937–947, 1992.

[213] Renou, S., Givaudan, J., Poulain, S., Dirassouyan, F., and Moulin, P. Landfill leachate treatment: Review and opportunity. *Journal of hazardous materials*, 150(3):468–493, 2008.

[214] Ribeiro, C. and Minoux, M. Solving hard constrained shortest path problems by Lagrangean relaxation and branch-and-bound algorithms. *Methods of Operations Research*, 53:303–316, 1986.

[215] Rimaityte, I., Ruzgas, T., Denafas, G., Racys, V., and Martuzevicius, D. Forecasting municipal solid waste generation in a fast-growing urban region with system dynamics modeling. *Waste Management & Research*, pages 89–98, 2012.

[216] Rosenthal, R. E. *GAMS-a user's guide*, 2004.

[217] Ruszczyński, A. A regularized decomposition method for minimizing a sum of polyhedral functions. *Mathematical programming*, 35(3):309–333, 1986.

[218] Ryan, D. M. and Foster, B. A. An integer programming approach to scheduling. *Computer scheduling of public transport urban passenger vehicle and crew scheduling*, pages 269–280, 1981.

[219] Sachenbacher, M., Leucker, M., Artmeier, A., and Haselmayr, J. Efficient energy-optimal routing for electric vehicles. In *25th AAAI Conference on Artificial Intelligence*, 2011. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3735.

[220] Sahinidis, N. *BARON user manual v. 15.6.5*. URL http://www.minlp.com/downloads/docs/baron%20manual.pdf.

[221] Schmöller, S. and Bogenberger, K. Analyzing external factors on the spatial and temporal demand of car sharing systems. *Procedia - Social and Behavioral Sciences*, 111:8–17, February 2014. ISSN 1877-0428. doi: 10.1016/j.sbspro.2014.01.033.

[222] Schneider, M., Stenger, A., and Goeke, D. The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, 48(4): 500–520, November 2014. ISSN 1526-5447. doi: 10.1287/trsc.2013.0490.

[223] Sellmair, R. and Hamacher, T. Method of optimization for the infrastructure of charging station for electric taxis. In *Proceedings of the 93rd Annual Meeting of the Transportation Research Board*, 2014.

[224] Shafik, N. and Bandyopadhyay, S. *Economic growth and environmental quality: time-series and cross-country evidence*, volume 904. World Bank Publications, 1992.

[225] Shaheen, S., Sperling, D., and Wagner, C. Carsharing in europe and North America: Past, present and future. *Transportation Quarterly*, 52(3):35–52, 1998.

[226] Shevchik, V. Multicommodity flow problem. *Journal of Mathematical Sciences*, 65(1):1462–1464, 1993.

[227] Shivaswamy, P. K., Bhattacharyya, C., and Smola, A. J. Second order cone programming approaches for handling missing and uncertain data. *Journal of Machine Learning Research*, 7:1283–1314, Dec. 2006. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1248547.1248594.

[228] Silva, F. J. F. and De la Figuera, D. A capacitated facility location problem with constrained backlogging probabilities. *International journal of production research*, 45(21):5117–5134, 2007.

[229] Singh, G. K., Gupta, K., and Chaudhary, S. Solid waste management: Its sources, collection, transportation and recycling. *International Journal of Environmental Science & Development*, 2014.

[230] Smith, J. C. and Taskın, Z. C. A tutorial guide to mixed-integer programming models and solution techniques. 2007.

[231] Stieber, A., Fügenschuh, A., Epp, M., Knapp, M., and Rothe, H. The multiple traveling salesmen problem with moving targets. *Optimization Letters*, pages 1–15, 2014.

[232] Stillwater, T., Mokhtarian, P., and Shaheen, S. Carsharing and the built environment: A GIS-based study of one U.S. operator. In *Transportation Research Record*, volume 2110, pages 27–34, January 2009. URL http://www.its.ucdavis.edu/wp-content/themes/ucdavis/pubs/download_pdf.php?id=1503.

[233] Sun, W., An, C., Li, G., and Lv, Y. Applications of inexact programming methods to waste management under uncertainty: current status and future directions. *Environmental Systems Research*, 3(1):15, 2014. doi: 10.1186/s40068-014-0015-9. URL http://dx.doi.org/10.1186/s40068-014-0015-9.

[234] Sundar, K. and Rathinam, S. Algorithms for routing an unmanned aerial vehicle in the presence of refueling depots. *IEEE Transactions on Automation Science and Engineering*, 11(1):287–294, Jan 2014. ISSN 1545-5955. doi: 10.1109/TASE.2013.2279544.

[235] Tarjan, R. Depth first search and linear graph algorithms. *SIAM Journal on Computing*, 1972.

[236] Thomas, P. R., Bhandari, U., Bullock, S., Richardson, T. S., and du Bois, J. L. Advances in air to air refuelling. *Progress in Aerospace Sciences*, 71: 14 – 35, 2014. ISSN 0376-0421. doi: http://dx.doi.org/10.1016/j.paerosci.2014.07.001. URL http://www.sciencedirect.com/science/article/pii/S0376042114000657.

[237] Todd, M. J. Potential-reduction methods in mathematical programming. *Mathematical Programming*, 76(1):3–45, 1997.

[238] Tomlin, J. A suggested extension of special ordered sets to non-separable non-convex programming problems. *North-Holland Mathematics Studies*, 59:359–370, 1981.

[239] Torino, M. Studio della raccolta multimateriale leggero di rifiuti solidi urbani. Master's thesis, Università Bicocca, 2005.

[240] Toth, P. and Vigo, D. *Vehicle Routing: Problems, Methods, and Applications, Second Edition*. SIAM, 2014.

[241] Trebouet, D., Schlumpf, J., Jaouen, P., and Quemeneur, F. Stabilized landfill leachate treatment by combined physicochemical–nanofiltration processes. *Water Research*, 35(12):2935–2942, 2001.

[242] Tseng, P. Second-order cone programming relaxation of sensor network localization. *SIAM Journal on Optimization*, 18(1):156–185, 2007.

[243] Van Duin, J., Tavasszy, L. A., and Quak, H. Towards e(lectric)-urban freight: first promising steps in the electric vehicle revolution. *European Transport / Trasporti Europei*, 54(9):1–19, 2013. Published online.

[244] Van Slyke, R. M. and Wets, R. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.

[245] Varone, S. and Aissat, K. Multi-modal transportation with public transport and ride-sharing. In *ICEIS 2015*, 2015.

[246] Vielma, J. P., Ahmed, S., and Nemhauser, G. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Operations research*, 58(2):303–315, 2010.

[247] Vigo, D., Bonoli, A., Gricinella, A. C., and Zarri, G. *Key Elements for Optimal Integrated Urban Solid Waste Management - International Experiences*. Esculapio, 2007.

[248] Wagner, J. and Bilitewski, B. The temporary storage of municipal solid waste–recommendations for a safe operation of interim storage facilities. *Waste management*, 29(5):1693–1701, 2009.

[249] Wallace, M. Practical applications of constraint programming. *Constraints*, 1 (1-2):139–168, 1996.

[250] Wang, H., Huang, Q., Zhang, C., and Xia, A. A novel approach for the layout of electric vehicle charging station. In *The 2010 International Conference on Apperceiving Computing and Intelligence Analysis Proceeding*. IEEE, December 2010. ISBN http://id.crossref.org/isbn/978-1-4244-8025-8. doi: 10.1109/icacia.2010.5709852.

[251] Wang, Y.-W. and Lin, C.-C. Locating multiple types of recharging stations for battery-powered electric vehicle transport. *Transportation Research Part E: Logistics and Transportation Review*, 58:76–87, November 2013. ISSN 1366-5545. doi: 10.1016/j.tre.2013.07.003.

[252] Westerlund, T. and Pettersson, F. An extended cutting plane method for solving convex MINLP problems. *Computers & Chemical Engineering*, 19:131–136, 1995.

[253] Wilson, E., McDougall, F., and Willmore, J. Euro-trash: searching europe for a more sustainable approach to waste management. *Resources, Conservation and Recycling*, 31(4):327 – 346, 2001. ISSN 0921-3449. doi: http://dx.doi.org/10.1016/S0921-3449(00)00089-6. URL http://www.sciencedirect.com/science/article/pii/S0921344900000896.

[254] Worley, O., Klabjan, D., and Sweda, T. M. Simultaneous vehicle routing and charging station siting for commercial electric vehicles. In *2012 IEEE International Electric Vehicle Conference*. IEEE, March 2012. ISBN http://id.crossref.org/isbn/978-1-4673-1562-3. doi: 10.1109/ievc.2012.6183279.

[255] Xpress-Mosel Manual. *Xpress-Mosel Reference manual*. URL http://www.maths.ed.ac.uk/hall/Xpress/FICO_Docs/mosel/mosel_lang/dhtml/moselref.html. Release 3.4.

[256] Xpress-Optimizer Manual. *Xpress-Optimizer Reference manual*. URL http://www.fico.com/en/node/8140?file=5097. Release 20.00.

[257] Xu, K., Yi, P., and Kandukuri, Y. Location selection of charging stations for battery electric vehicles in an urban area. *International Journal of Engineering Research and Science & Technology*, 2(3):15–23, 2013. URL http://www.ijerst.com/ijerstadmin/upload/IJEETC_51ffe12fbeda5.pdf.

[258] Yang, J. and Sun, H. Battery swap station location-routing problem with capacitated electric vehicles. *Computers & Operations Research*, 55:217–232, July 2014. ISSN 0305-0548. doi: 10.1016/j.cor.2014.07.003.

[259] Yeomans, J., Huang, G., and Yoogalingam, R. Combining simulation with evolutionary algorithms for optimal planning under uncertainty: an application

to municipal solid waste management planning in the regional municipality of Hamilton-Wentworth. *Journal of Environmental Informatics*, 2(1):11–30, 2003.

[260] Yeomans, J. S. Solid waste planning under uncertainty using evolutionary simulation-optimization. *Socio-Economic Planning Sciences*, 41(1):38–60, 2007.

[261] Zhang, Y., Huang, G. H., and He, L. A multi-echelon supply chain model for municipal solid waste management system. *Waste management*, 34(2):553–561, 2014.

[262] Zhao, X. and Luh, P. New bundle methods for solving Lagrangian Relaxation dual problems. *Journal of Optimization Theory and Applications*, 113(2):373–397, 2002.