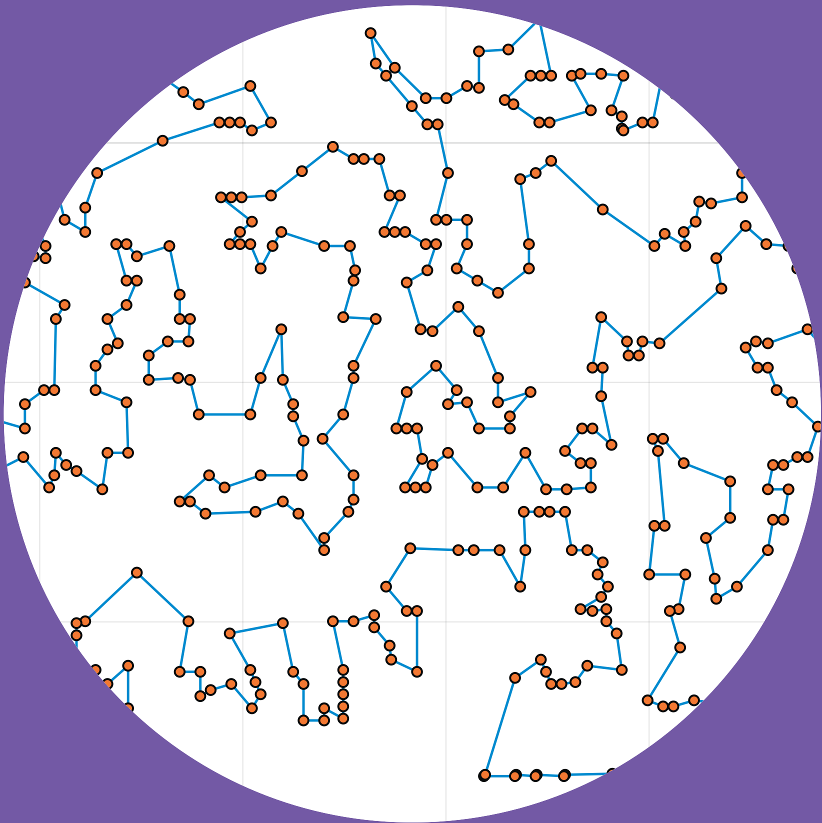
Department of Mathematics and Systems Analysis

Models and algorithms for vehicle routing, resource allocation, and multi-stage decision-making under uncertainty

Juho Andelmin



Aalto University

DOCTORAL
DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 82/2021

Models and algorithms for vehicle routing, resource allocation, and multi-stage decision-making under uncertainty

Juho Andelmin

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Science, at a public examination held at the lecture hall H304 of the school on 23 July 2021 at 10.

The public defense will be also organized via remote technology. Link: https://aalto.zoom.us/j/63059713880. Zoom quick guide: https://www.aalto.fi/en/services/zoom-quick-guide.

Aalto University
School of Science
Department of Mathematics and Systems Analysis

Supervising professor
Professor Ahti Salo, Aalto University, Finland

Thesis advisors
Dr. Enrico Bartolini, RWTH Aachen University, Germany
Associate Professor Fabricio Oliveira, Aalto University, Finland
Associate Professor Juuso Liesiö, Aalto University, Finland
Professor Ahti Salo, Aalto University, Finland

Preliminary examiners
Professor Jesper Larsen, Technical University of Denmark (DTU), Denmark
Professor Jeffrey T. Linderoth, University of Wisconsin-Madison, USA

Opponent
Professor M. Grazia Speranza, University of Brescia, Italy

Aalto University publication series
DOCTORAL DISSERTATIONS 82/2021

ISBN 978-952-64-0417-2 (printed)
ISBN 978-952-64-0418-9 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-64-0418-9

Unigrafia Oy
Helsinki 2021

Finland

Printed matter
4041-0619

Aalto University

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Abstract

Author
Juho Andelmin

Name of the doctoral dissertation
Models and algorithms for vehicle routing, resource allocation, and multi-stage decision-making under uncertainty

Publisher School of Science

Unit Department of Mathematics and Systems Analysis

Series Aalto University publication series DOCTORAL DISSERTATIONS 82/2021

Field of research Systems and Operations Research

Manuscript submitted 23 June 2021 **Date of the defence** 23 July 2021

Permission for public defence granted (date) 22 June 2021 **Language** English

☐ **Monograph** ☒ **Article dissertation** ☐ **Essay dissertation**

Abstract
In difficult optimization problems, strong formulations and algorithmic techniques that exploit the problem structure are often invaluable in designing efficient solution methods. Although microprocessors and generic solvers have reduced solution times, these tools are often not enough to solve hard problems of realistic size. To overcome these challenges, the standard practice has typically been to develop tailored, problem-specific algorithms. This summary chapter introduces efficient formulations and algorithms for three different optimization problems, each of which either serves as a basis for future extensions or unifies previous approaches under one framework.
First, two algorithms are developed for the green vehicle routing problem. Both algorithms rely on a novel multigraph reformulation that transforms refueling nodes into non-dominated refuel paths between customers. This transformation allows combining routing and refueling decisions with negligible overhead. Both algorithms serve as building blocks for developing new solution methods for generalizations of the problem. The effectiveness of the multigraph and the developed algorithms are demonstrated through computational evaluation.
Second, a new framework for centralized allocation of resources to a portfolio of decision-making units is developed. This framework can handle multiple objectives with incomplete preferences and compute all non-dominated portfolios satisfying these preferences. Each portfolio corresponds to a Pareto-optimal allocation of resources among the decision-making units that maximizes portfolio-level efficiency. The framework unifies several previous models that compute single solutions from the efficient frontier, possibly involving non-linear utilities and many kinds of production possibility sets. It also demonstrates that relying on conventional efficiency scores in guiding resource allocation decisions may lead to inefficiencies at the portfolio level.
Third, a novel *Decision Programming* approach is developed that contributes towards unifying stochastic programming and decision analysis within a single framework and relaxes two common assumptions in decision analysis: (i) perfect recall where all prior decisions are known when making a decision and (ii) regularity that assumes a total temporal order for decision variables. Decision Programming relies on a mixed-integer linear programming formulation that can handle both endogenous and exogenous uncertainties and can also optimize problems involving simultaneous decisions by agents unable to communicate with each other. The Decision Programming framework can be extended to incorporate deterministic and chance constraints, and it can be harnessed to compute all non-dominated solutions in presence of multiple value functions. Most importantly, it contributes towards approaches that can solve problems from both decision analysis and stochastic programming, and may thus facilitate collaboration between these two sub-disciplines in the future.

Keywords optimization, vehicle routing, resource allocation, decision-making under uncertainty

ISBN (printed) 978-952-64-0417-2 **ISBN (pdf)** 978-952-64-0418-9

ISSN (printed) 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2021

Pages 152 **urn** http://urn.fi/URN:ISBN:978-952-64-0418-9

A? **Aalto-yliopisto**

Aalto-yliopisto, PL 11000, 00076 Aalto www.aalto.fi

Tiivistelmä

Tekijä
Juho Andelmin

Väitöskirjan nimi
Malleja ja algoritmeja ajoneuvon reititykseen, resurssien allokointiin, sekä monijaksoiseen päätöksentekoon epävarmuuden alla

Julkaisija Perustieteiden korkeakoulu

Yksikkö Matematiikan ja systeemianalyysin laitos

Sarja Aalto University publication series DOCTORAL DISSERTATIONS 82/2021

Tutkimusala Systeemi- ja operaatiotutkimus

Käsikirjoituksen pvm 23.06.2021 **Väitöspäivä** 23.07.2021

Väittelyluvan myöntämispäivä 22.06.2021 **Kieli** Englanti

☐ **Monografia** ☒ **Artikkeliväitöskirja** ☐ **Esseeväitöskirja**

Tiivistelmä
Vaikeita optimointiongelmia ratkottaessa vahvat formulaatiot ja ongelman rakennetta hyödyntävät ratkaisutekniikat ovat usein keskeisiä. Vaikka mikroprosessorit ja yleiset ratkaisuohjelmat ovat nopeutuneet huomattavasti lyhyen ajan sisällä, ne eivät usein pysty yksinään ratkaisemaan isoja reaalimaailman ongelmia. Tyypillisesti ainoa vaihtoehto on ollut ongelmakohtaisten algoritmien kehittäminen jokaiselle ongelmalle erikseen. Tämä väitöskirjan tiivistelmä esittää tehokkaita malleja ja ratkaisumenetelmiä kolmelle optimointiongelmalle, jotka joko toimivat pohjana uusille laajennuksille, tai yhdistävät useita eri menetelmiä isommaksi viitekehykseksi.
Ensiksi on kehitetty kaksi algoritmia vihreälle ajoneuvon reititysongelmalle. Molemmat algoritmit hyödyntävät uutta monigraafi-reformulaatiota, mikä muuttaa tankkausasemia kuvaavat solmut ei-dominoiduksi energiapoluiksi asiakkaiden välillä. Tämä muunnos mahdollistaa sekä reititys- että tankkauspäätösten tekemisen samanaikaisesti ilman merkittävää laskennallista haittaa. Molemmat algoritmit toimivat myös rakennuspalikoina uusien ratkaisumenetelmien kehittämiseen ongelman laajennuksille. Monigraafi-reformulaatio sekä molemmat algoritmit ovat osoittautuneet tehokkaiksi laskennallisten testien avulla.
Toiseksi on kehitetty viitekehys keskitettyyn resurssien jakamiseen portfoliolle päätöksentekoyksiköitä. Tämä viitekehys pystyy käsittelemään useita päätösfunktioita ottamalla huomioon epätarkat preferenssit ja laskemaan kaikki ei-dominoidut portfoliot, mitkä toteuttavat kyseiset preferenssit. Jokainen ei-dominoitu portfolio vastaa Pareto-optimaalista resurssien jakoa päätöksentekoyksiköiden kesken, mikä maksimoi portfolion tehokkuuden. Kehitetty viitekehys yhdistää useita eri malleja, jotka itsekseen tuottavat yksittäisiä ratkaisuja tehokkaalta rintamalta, mahdollisesti sisältäen epälineaarisia hyötyfunktioita, sekä monen laisia tuotantomahdollisuusjoukkoja. Viitekehys myös osoittaa, että perinteisillä tehokkuusluvuilla ei voida luotettavasti ohjata resurssien allokointipäätöksiä maksimoidessa portfoliotason tehokkuutta.
Kolmanneksi on kehitetty *Decision Programming* menetelmä, mikä edistää stokastisen optimoinnin ja päätösanalyysin yhdistämistä samaan viitekehykseen, sekä relaksoi kaksi yleistä päätösanalyysiin liittyvää oletusta: (i) täydellinen muisti, missä aikaisemmat päätökset tiedetään uusia tehdessä, sekä (ii) säännöllisyys (engl. regularity), missä päätökset noudattavat lineaarista järjestystä. Menetelmä hyödyntää sekalukuista lineaarista optimointimallia, mikä pystyy käsittelemään sekä endo-, että eksogeenisia epävarmuuksia, ja ratkaisemaan ongelmia missä joukko agentteja, jotka eivät pysty kommunikoimaan keskenään, tekevät samanaikaisesti päätöksiä. Menetelmä voidaan yleistää huomioimaan deterministisiä- ja satunnaisrajoitteita, sekä laskemaan kaikki ei-dominoidut ratkaisut usealle tavoitefunktiolle. Mikä tärkeintä, se antaa edellytyksiä sille, että päätösanalyysin ja stokastisen optimoinnin asiantuntijat voivat tehdä yhteistyötä tulevaisuudessa.

Avainsanat optimointi, ajoneuvon reititys, resurssien allokointi, monijaksoinen päätöksenteko

ISBN (painettu) 978-952-64-0417-2 **ISBN (pdf)** 978-952-64-0418-9

ISSN (painettu) 1799-4934 **ISSN (pdf)** 1799-4942

Julkaisupaikka Helsinki **Painopaikka** Helsinki **Vuosi** 2021

Sivumäärä 152 **urn** http://urn.fi/URN:ISBN:978-952-64-0418-9

Preface

First of all I would like to express my gratitude to my supervisor Ahti Salo for all his support and encouragement to finally finish this dissertation. I also wish to thank my advisor Juuso Liesiö who hired me after finishing my B.Sc. degree to work on an interesting problem that several years later finally turned into an article. I wish to thank my advisor Fabricio Oliveira for our collaboration thus far and providing the means to keep researching new interesting topics. I also want to thank all my colleagues for making the working environment more enjoyable.

I would like to specifically thank my first supervisor and advisor Enrico Bartolini with whom I have worked the most since I started as a research assistant in Aalto. I learned more during the years I worked with Enrico that any other time period. I would also like to extend my thanks to Erik Neuvonen for his invaluable support.

Last but not least, I am indebted to my parents Anne and Pertti and my sisters Hilla and Johanna who have been extremely supportive and understanding by allowing me focus on research over these years.

Espoo, June 23, 2021,

Juho Andelmin

Contents

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I Juho Andelmin and Enrico Bartolini. An exact algorithm for the green vehicle routing problem. *Transportation Science,* July 2017.

II Juho Andelmin and Enrico Bartolini. A multi-start local search heuristic for the green vehicle routing problem based on a multigraph reformulation. *Computers & Operations Research,* September 2019.

III Juuso Liesiö, Juho Andelmin, and Ahti Salo. Efficient allocation of resources to a portfolio of decision making units. *European Journal of Operational Research,* October 2020.

IV Ahti Salo, Juho Andelmin, and Fabricio Oliveira. Decision Programming for mixed-integer multi-stage optimization under uncertainty. *Submitted Manuscript,* August 2020.

Author's Contribution

Publication I: "An exact algorithm for the green vehicle routing problem"

Andelmin and Bartolini are both primary authors. Bartolini proposed the initial core idea of the algorithm. Andelmin and Bartolini refined the algorithm subsequently, and they both worked on different parts of the implementation simultaneously using a common repository. Andelmin and Bartolini wrote the paper as a joint effort

Publication II: "A multi-start local search heuristic for the green vehicle routing problem based on a multigraph reformulation"

Andelmin is the primary author. He designed and implemented the matheuristic algorithm and performed computational tests. Bartolini provided suggestions on how to improve the algorithm and the exposition of the paper. Andelmin and Bartolini wrote the paper cooperatively.

Publication III: "Efficient allocation of resources to a portfolio of decision making units"

Liesiö is the primary author and proposed the core idea of the paper. Andelmin wrote the first version of the manuscript with Liesiö. Andelmin designed, implemented, and reported all computational tests. Andelmin wrote first versions of the proofs of all Theorems and Corollaries and helped write the paper with Liesiö and Salo.

Publication IV: "Decision Programming for mixed-integer multi-stage optimization under uncertainty"

Salo is the primary author and proposed the core idea. Andelmin proposed the main formulation plus valid cutting planes based on Oliveira's and Salo's initial models. Andelmin implemented all computational tests and reported the results. Andelmin wrote the first version of the proof of Theorem 1 and formulated the final CVaR model based on ideas from Salo and Oliveira. Andelmin helped write the paper with Salo and Oliveira.

List of Figures

List of Tables

Abbreviations

AFV	Alternative Fuel Vehicle
CVaR	Conditional Value-at-Risk
CO	Combinatorial Optimization
DM	Decision Maker
DMU	Decision Making Unit
ID	Influence Diagram
LIMID	Limited Memory Influence Diagram
LP	Linear Programming
ML	Machine Learning
MOP	Multi-Objective Programming
MILP	Mixed-Integer Linear Programming
MOLP	Multi-Objective Linear Programming
MSLS	Multi-Start Local Search
NP	Nondeterministic Polynomial Time
SP	Stochastic Programming
SPP	Set Partitioning Problem
VRP	Vehicle Routing Problem
G-VRP	Green Vehicle Routing Problem
ECVRP	Electric Capacitated Vehicle Routing Problem
E-RCSPP	Elementary Resource Constrained Shortest Path Problem
E-VRPTW	Electric Vehicle Routing Problem with Time Windows
E-VRPTW-PR	Electric Vehicle Routing Problem with Time Windows and Partial Recharges

Symbols

x	Decision variable vector $x \in \mathbb{R}^n$.
$f(x)$	Objective function $f : \mathbb{R}^n \to \mathbb{R}$.
$g(x)$	Constraint functions $g : \mathbb{R}^n \to \mathbb{R}^m$
N	Index set of customers $i \in N$.
$\mathscr{R}$	Index set of all feasible vehicle routes $r \in \mathscr{R}$.
c_r	Cost of a feasible vehicle route $r \in \mathscr{R}$.
x_r	Variable $x_r = 1$ if route r is in solution, $x_r = 0$ otherwise.
a_{ir}	Scalar $a_{ir} = 1$ if customer i is in route r, $a_{ir} = 0$ otherwise.
Ω	Index set of scenarios $\omega \in \Omega$.
$y(\omega)$	Second stage recourse variables $y(\omega)$ for all $\omega \in \Omega$.
$P(\Omega)$	Probability distribution of scenarios $\omega \in \Omega$.
α	Tail probability in CVaR computations $\alpha \in (0,1)$.

1. Introduction

1.1 Methodological Background

Mathematical optimization is concerned with finding the best solution from a set of alternative solutions subject to stated evaluation criteria. Optimization problems arise naturally in all quantitative fields of study from computer science and engineering to operations research and economics. Building blocks of an optimization problem are decision variables that take values within specified domains, constraint functions that form a *feasible region* which further restricts the decision variable domains, and one or more objective functions that are optimized by finding optimal solutions within the feasible region that either minimize or maximize the objective function values. Some optimization problems may also have uncertain stochastic elements either in the problem data or embedded in the problem structure as endogenous or exogenous uncertainties (Hellemo et al., 2018).

As an example of a single-objective problem, let $x \in \mathbb{R}^n$ be the vector of decision variables, $f : \mathbb{R}^n \to \mathbb{R}$ the objective function, and $g : \mathbb{R}^n \to \mathbb{R}^m$ the constraint functions (in vector form) of an optimization problem. This optimization problem can be formulated in general form as

$$\text{maximize} \quad f(x) \tag{1.1}$$

$$\text{subject to} \quad g(x) \leq 0, \tag{1.2}$$

$$x \in \mathbb{R}^n. \tag{1.3}$$

The goal is to maximize the objective function (1.1) such that its value becomes as large as possible with respect to the decision variable values (1.3). The constraint functions (1.2) impose limitations to the decision variables by restricting their domains to be inside the feasible region. The constraints (1.2) thus play a key role in determining the best course of action when searching for the optimal solution that gives the optimal (maximum) objective function value. Problems that have continuous variables, linear objective function, and linear constraint functions are called linear programming (LP) problems.

A maximization problem can always be transformed into an equivalent minimization problem (and vice versa) by multiplying the objective function by -1. The domain of some (or all) decision variables can also be restricted to take integer values $x \in \mathbb{Z}^n$. Problems that are linear and have both integer and continuous variables are called mixed-integer linear programming (MILP) problems. Some problems have decision variables $x \in \{0,1\}^n$ that take only binary values and typically involve finding the optimal element (or a subset of elements) from a discrete set of elements. Such problems are typically called combinatorial optimization problems (COPs). An important example COP is the set partitioning problem (SPP) in which a large set of elements is to be optimally divided into element-disjoint subsets such that every element is part of an exactly one subset. COPs can be solved using dynamic programming (Bellman, 1952) or formulated as MILPs and solved with any MILP solver.

A prominent example of a SPP which can be formulated as a MILP is the vehicle routing problem (VRP) in which a set of customers is to be served by a number of vehicles located at a depot. The objective is to design least-cost vehicle routes, each starting from and ending at the depot, such that every customer is served exactly once. In this case, the set of elements corresponds to the set of customers and the subsets correspond to different vehicle customer subsets that can be served by any vehicle. The objective is to find the least-cost partitioning of customers into subsets such that (i) each customer is in exactly one subset and (ii) the customers in each subset are served by a vehicle that visits them in such an order that minimizes the traveled distance (Balinski & Quandt, 1964). VRP generalizes the traveling salesman problem which is similar but has only one vehicle. Because the TSP is NP-hard, so is the VRP (Papadimitriou, 2003).

To illustrate the use of the SPP formulation, let $N = \{1,\dots,n\}$ be the set of n customers and $\mathscr{R}$ the set of all vehicle routes associated with all possible customer subsets visited in any given order. Define binary variables $x_r \in \{0,1\}$, for all $r \in \mathscr{R}$, so that $x_r = 1$ if the vehicle route r is part of the solution and $x_r = 0$ otherwise. Let c_r be the route costs and a_{ir} scalars so that $a_{ir} = 1$ if customer i is part of the route r and $a_{ir} = 0$ otherwise for all $r \in \mathscr{R}$ and $i \in N$. Using this notation, the VRP can be formulated as the SPP

$$\text{minimize} \quad \sum_{r \in \mathscr{R}} c_r x_r \tag{1.4}$$

$$\text{subject to} \quad \sum_{r \in \mathscr{R}} a_{ir} x_r = 1, \qquad \forall\, i \in N \tag{1.5}$$

$$x_r \in \{0,1\}, \qquad \forall\, r \in \mathscr{R} \tag{1.6}$$

The SPP formulation (1.4) – (1.6) has an exponential number of variables $x_r \in \mathscr{R}$. Thus, solving it directly is impractical except for very small problems with a few customers. However, using advanced techniques such as column generation, cutting planes, and dynamic programming to generate feasible vehicle routes, VRPs with up to 200 customers can be solved to optimality in reasonable times (see, e.g., Pessoa et al., 2020).

Hard optimization problems are typically solved by developing problem-specific algorithms. However, when such problems become large enough, exact methods may not be able to compute even good approximate solutions in a reasonable time. In such cases, it may be necessary to develop heuristic solution methods instead, with the goal of producing "good enough" solutions without spending excessive amounts of time or computational resources. A major drawback of relying only on heuristic methods is that their unknown solution quality can be far from optimal. To alleviate this trade-off between solution quality and computation time, exact methods can often be transformed into heuristic ones by relaxing or modifying some components of the exact method so that instead of working towards global optimality, the focus is shifted into finding decent solutions with less computational effort. Such relaxed exact optimization methods can then not only compute feasible solutions, but also provide approximations of solution quality through optimality gaps which give upper bounds on how far the returned solutions are from an optimal solution. Specifically, the optimality gap corresponds to the percentage distance between the objective value of the solution and the best dual bound found by the algorithm.

A large class of heuristics uses combinations of rules and operators that produce feasible solutions fast, although without any optimality estimates. These methods are typically assessed by comparing their performance to exact methods on smaller problem instances that can be solved to optimality, and hoping that similar performance is achieved also with larger problems. Unfortunately, this is rarely the case. In fact, the opposite is typically true. There are some simple ways to increase the probability of finding better solutions through randomization. Multi-start local search

(MSLS) (see, e.g., Martí et al., 2013) is one such method that generates several different initial solutions by randomization, performs local search on each generated solution, and finally selects the best one. The MSLS can also handle problems with decomposable structures such as VRPs. First, all vehicle routes of each generated solution are stored into a route pool. Then, a new solution is computed by finding the best combination of routes in the route pool by modeling and solving the problem as a SPP (1.4) – (1.6) where $\mathscr{R}$ corresponds to the set of routes in the pool. These kinds of heuristics that utilize mathematical optimization are typically called *matheuristics*. Some authors further argue that the use of mathematical optimization as part of the heuristic method alone is not enough: to be classified as a matheuristic, the method should also provide an optimality gap or a thorough comparison with other methods (Voss et al., 2009).

Optimization problems can also have multiple objective functions $f: \mathbb{R}^n \to \mathbb{R}^s$ in which case the problem typically has several *non-dominated* solutions instead of a single optimal one. A solution $x \in \mathbb{R}^n$ dominates another solution $x' \in \mathbb{R}^n$ if both x and x' are feasible and $f_i(x) \geq f_i(x')$, for all $i = 1,\ldots,s$, and $f_i(x) > f_i(x)$ for at least one $i = 1,\ldots,s$, assuming that each objective function $f_i(x)$ for all $i = 1,\ldots,s$ is transformed into a maximization form. Non-dominated solutions are those that are not dominated by any other solution. All non-dominated solutions form the solution set to the problem. One possibility to choose a single solution from the non-dominated set is to use a utility function $U: \mathbb{R}^s \to \mathbb{R}$ that takes the objective function values of a non-dominated solution to compute its utility. The maximum-utility solution is then considered optimal.

Using the above notation, a general multi-objective optimization problem with objective functions $f: \mathbb{R}^n \to \mathbb{R}^s$ can be written as

$$\text{v-maximize} \quad f(x) \tag{1.7}$$

$$\text{subject to} \quad g(x) \leq 0, \tag{1.8}$$

$$x \in \mathbb{R}^n. \tag{1.9}$$

The only differences between the single-objective optimization problem (1.1) – (1.3) and the multi-objective one (1.7) – (1.9) are the 'v-maximize' notation which stands for *vector maximization*, and the number of objective functions which in this case is s instead of one. When the objectives (1.7) and constraint functions (1.8) of the problem are linear, it becomes a multi-objective LP (MOLP) problem (Ehrgott, 2005; Löhne, 2011),

In order to model real-life decisions more accurately, the corresponding problem formulations often involve multiple decision stages and stochastic elements. Indeed, uncertainty plays a significant role in many real-life decision problems that are difficult to model without problem-specific stochastic components. For example, the problem data at each stage may be defined over a discrete number of possible scenarios whose probabilities follow a specific probability distribution. Moreover, the probability distributions at different stages may be influenced endogenously by decisions made in the previous stages of the problem. The probability distributions may also change based on the timing of certain decisions, adding yet another layer of complexity. The former type of decision-dependent uncertainty has recently been categorized as Type-1 endogenous uncertainty, while the latter in which uncertainty is related to the timing of decisions is categorized as Type-2 endogenous uncertainty (Hellemo et al., 2018).

Multi-stage decision problems under uncertainty are typically modeled as stochastic programming (SP) problems (Ruszczyński & Shapiro, 2003; Birge & Louveaux, 2011). In conventional SP models, the uncertainty is typically assumed to be exogenous, meaning that probability distributions of uncertain realizations cannot be influenced by prior decisions. While this is reasonable in many contexts, such as one cannot influence the probability that it will rain the next day, there are many problems in which decisions have significant influence on the probability values of uncertain future events. For example, the probability of fully recovering from a bacterial infection in one week increases if the person makes the decision of taking antibiotics. Over the last decade, the number of studies involving endogenous uncertainties has been steadily increasing; however, the focus has been mostly on the Type-2 endogenous uncertainty related to timing of decisions (Goel & Grossmann, 2006; Solak et al., 2010; Gupta & Grossmann, 2014). SP problems with Type-1 decision-dependent endogenous uncertainty are much less studied (see, e.g., Hellemo et al., 2018).

To illustrate how simple stochastic elements can affect deterministic problems, consider an extension of the single-objective optimization problem (1.1) – (1.3) to a two-phase stochastic problem involving uncertain elements. Let $\Omega = \{\omega_1, \dots, \omega_{|\Omega|}\}$ be a discrete set of random outcomes where each $\omega \in \Omega$ corresponds to a *scenario* with corresponding scenario probabilities $p(\omega) \in P(\Omega)$, for all $\omega \in \Omega$, for the $|\Omega|$ different scenarios that can occur after optimizing the first-stage variable x. Moreover, let $y(\omega)$ denote the *second-stage* variables that represent corrective *recourse* actions for each scenario $\omega \in \Omega$. For example, the first-stage decision x could represent how many medical tests are needed to track the spread of COVID-19, and after observing the outcome of the realized scenario $\omega \in \Omega$, which could represent the fraction of people who were tested positive during a testing period, the recourse variables $y(\omega)$ could then either increase or decrease the testing capacity depending on the scenario ω.

The scenario-dependent problem data is typically denoted as $q(\omega)$, $T(\omega)$, $h(\omega)$, along with a matrix W, that all have appropriate dimensions. Also, in this case, both the first-stage and second-stage variables x and $y(\omega)$, for all $\omega \in \Omega$, must take integer values, because each variable corresponds to a number of tests which cannot be fractional. Using this notation, the two-stage SP problem can be formulated as

$$\text{maximize} \quad f(x) + \sum_{\omega \in \Omega} p(\omega) q(\omega) y(\omega) \tag{1.10}$$

$$\text{subject to} \quad g(x) \le 0, \tag{1.11}$$

$$h(\omega) = T(\omega)x + W y(\omega), \qquad \forall \omega \in \Omega \tag{1.12}$$

$$y(\omega) \in \mathbb{Z}, \qquad \forall \omega \in \Omega \tag{1.13}$$

$$x \in \mathbb{Z}_+ \tag{1.14}$$

The optimal solution to (1.10) – (1.14) determines the first-stage decision $x \in Z_+$ such that $f(x)$ plus the expected utility resulting from changes in testing capacity would be maximized over all the scenarios. While this simple example has only two stages, in general even two-stage SP problems pose major computational challenges, because their complexity increases rapidly when scenarios are produced by several parameters with different values in real-world applications (see, e.g., Boland et al., 2018).

Probabilities in the SP problem (1.10) – (1.14) are exogenous, as the probability distribution $P(\Omega) = \{p(\omega) \mid \omega \in \Omega\}$ remains the same regardless of the first-stage variable value $x \in \mathbb{Z}_+$. However, it is possible to induce a decision-dependent endogenous probability structure by making the probability distribution $P(\Omega)$ dependent on the first-stage variable x. One such approach would be to first restrict the domain of x such that it can take k different values $x_1, \ldots, x_k$. Then, instead of having just one probability distribution $P(\Omega)$, the problem would have k different probability distributions $P_1(\Omega), \ldots, P_k(\Omega)$, one for each possible value of $x_1, \ldots x_k$.

Figure 1.1 shows these two types of uncertainties in an *influence diagram* with three types of nodes. Squares represent decisions, circles represent probability distributions, and diamonds represent objective function values. Arcs between the nodes represent conditional and functional dependencies (Lauritzen & Nilsson, 2001). The left diagram is an example of exogenous uncertainty, while the right diagram exemplifies endogenous uncertainty where $P(\Omega)$ is conditionally dependent on x.

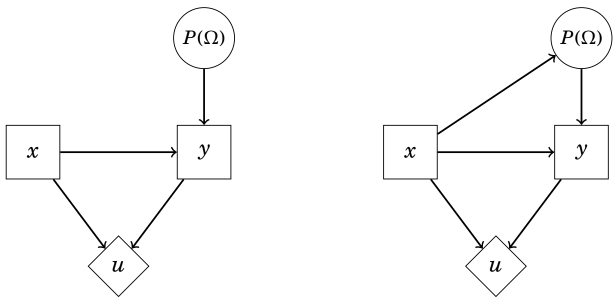


Figure 1.1. Left: an example of exogenous uncertainty $P(\Omega)$. Right: an example of decision-dependent endogenous uncertainty where $P(\Omega)$ is dependent on x.

1.2 Research Objectives

This introductory summary chapter presents the main contributions of Papers [I] – [IV]. The goal of the first chapter is to present a methodological overview of basic mathematical optimization concepts related to the problems in Papers [I] – [IV]. This prepares ground for the theoretical discussion about each individual problem and helps identify what is missing from the current literature that motivates the corresponding paper. Specifically, the objective is to:

1. Present methodological and theoretical overviews with a focus on the application areas addressed in the Papers [I] – [IV].
2. Present and discuss the most important contributions of these papers with respect to the current body of research.
3. Identify important research questions and potential gaps in the earlier literature.

1.3 Problem Classification

Section 1.1 describes the basic building blocks of the problems studied in Papers [I] – [IV]. Table 1.1 provides a classification of these problems and the optimization techniques (detailed in Section 1.1) that are used in the corresponding papers. In addition, Table 1.2 provides a more detailed account of the types of problems studied in each paper.

Papers [I] – [IV] develop new algorithmic ideas and formulations for important problems in green logistics, centralized resource allocation, and discrete multi-stage stochastic optimization problems. While these problems could be classified into different areas of operations research, the Papers [I] – [IV] showcase a number of general modeling techniques that can be utilized across a wider variety of optimization problems.

Table 1.1. Classification of problems and optimization techniques used in Papers [I] – [IV]. Alternative fuels include, for example, bio-diesel, electricity, and hydrogen.

	Problem class	Optimization techniques
Paper [I]	Alternative fuel vehicle routing	Mixed-integer linear programming Combinatorial optimization Dynamic programming Linear programming
Paper [II]	Alternative fuel vehicle routing	Mixed-integer linear programming Combinatorial optimization
Paper [III]	Centralized resource allocation	Multi-objective linear programming Linear programming
Paper [IV]	Multi-stage stochastic optimization	Mixed-integer linear programming Multi-objective programming Linear programming

Table 1.2. Characteristics of problems in Papers [I] – [IV]

	Objectives		Variables		Solution methods		
	Single	Multiple	Continuous	Discrete	Exact	Heuristic	Stochastic
Paper [I]	X		X	X	X		
Paper [II]	X			X		X	
Paper [III]		X	X	X	X		
Paper [IV]	X	X	X	X	X		X

1.4 Organization of this Summary Chapter

The rest of this summary chapter is organized as follows. Chapter 2 gives a brief theoretical background on the studied problems and discusses how the current theory can benefit from new ideas. Chapter 3 presents these ideas and summarizes research contributions of the papers. Chapter 4 provides concluding remarks on the problems studied in Papers [I] – [IV] and ideas for future research.

2. Theoretical Background

2.1 Vehicle Routing Problems for Alternative Fuel Vehicles

Vehicle routing problems (VRPs) constitute a large class of combinatorial optimization (CO) problems that involve designing least-cost delivery routes for a fleet of vehicles to serve a set of customers. Several VRP variants have been studied extensively in the past 60 years (Laporte, 2009). However, most VRP formulations that have been introduced are best suited for gasoline- and diesel-powered vehicles with long driving ranges, relatively short refueling times, and widespread refueling infrastructures.

Issues concerning fuel consumption, refueling delays, and their interplay with different operational constraints that describe real-life applications have been mostly overlooked. Specifically, the majority of VRP variants do not consider possibilities to refuel when designing optimal routing plans. This makes it difficult to apply conventional VRP formulations to routing problems involving alternative fuel vehicles (AFVs) with limited fueling infrastructures and shorter driving ranges. Without appropriate systems to design routing plans that involve refueling stops, users may experience *range anxiety* – the fear of running out of fuel en route – and become reluctant to travel longer distances (Franke et al., 2012). Range anxiety is regarded as one of the main barriers to large-scale adoption of electric vehicles (Eberle & Von Helmolt, 2010; Philip & Wiederer, 2010).

New vehicle routing models are therefore needed in navigation support and route planning to (i) mitigate range anxiety for private users and (ii) ensure a reliable and adequate level of service for organizations planning to adopt AFVs. To tackle these issues, Erdoğan & Miller-Hooks (2012) introduce the green VRP (G-VRP) that provides a modeling framework involving refueling stations and constraints that monitor vehicles' fuel consumption. The G-VRP has become the standard model for AFV routing which can be augmented with additional constraints such as time windows and customer demands (see, e.g., Schneider et al., 2014). The G-VRP itself is inspired by Bard et al. (1998) who introduces *intermediate facilities* where vehicles can restock supplies without returning to the depot.

As the G-VRP can be considered the core model that captures the combinatorial essence of AFV routing problems – specifically its extensions that include additional constrains – it is crucial to develop and analyze exact algorithms for the G-VRP in order to understand the complexities of this class of problems. Exact algorithms for the G-VRP are also useful for evaluating the quality of heuristic algorithms that serve as building blocks for developing new heuristics (and exact methods that can utilize such heuristics) to generalizations of the G-VRP.

From a modeling perspective, the standard formulation for the G-VRP, proposed by (Erdoğan & Miller-Hooks, 2012), requires making as many copies of each refueling station node as there are possible times for visiting each individual station. This is needed to maintain flow balance in the model and to distinguish between different vehicle routes. However, this also increases the number of variables and constraints significantly, which makes it more challenging to develop exact solution methods that guarantee optimality without limiting the number of possible stops to refueling stations. Typically, a trade-off has to be made between two extremes to either (i) allow all possible refueling stops to ensure optimality or (ii) limit the number of such stops to one per station to obtain a model with significantly less variables and constraints.

An alternative formulation for the electric VRP with time windows, customer demands, and partial recharges (E-VRPTW-PR) is introduced by Andelmin (2014) who generalizes the G-VRP and the E-VRPTW by Schneider et al. (2014) by allowing AFVs to partially refuel instead of fully refueling each AFV upon visiting a refueling station. It is also worth mentioning that unlike the studies in the literature that credit Felipe et al. (2014) for introducing the partial refueling (see, e.g., Asghari & Mirzapour Al-e-hashem, 2021), it was first introduced by Andelmin (2014). Moreover, this E-VRPTW-PR formulation is significantly stronger compared to the standard G-VRP formulation by Erdoğan & Miller-Hooks (2012) and the E-VRPTW formulation introduced by Schneider et al. (2014). A direct comparison is possible, because the E-VRPTW-PR generalizes both the G-VRP and the E-VRPTW and can thus solve instances of both problems straightforwardly. The improved E-VRPTW-PR formulation is based on a multigraph in which each arc corresponds to a sequence of non-dominated refueling stops between two customers. A similar problem-specific formulation would be beneficial for the G-VRP, because it avoids the multiplication of refueling station nodes, preserves optimality, and combines routing and refueling decisions by simply choosing different arcs between two customers or a customer and a depot.

Since the introduction of the G-VRP, there has been an increasing number of studies on the G-VRP and particularly its generalizations (for recent surveys, see, e.g., Lin et al., 2014; Dammak et al., 2019; Schiffer et al., 2019; and Asghari & Mirzapour Al-e-hashem, 2021). While most of these studies consider heuristic solution methods, exact algorithms have received much less attention. A possible reason for this could be the increased problem complexity arising from the route duration constraints, inclusion of refueling stations, and the need to keep track of vehicles' fuel levels. Vehicle routing problems with route duration constraints (or equivalently route distance constraints) are known to be extremely difficult, as evidenced by the problem instance CMT13 (Christofides et al., 1979) with 120 customers and route duration constraints whose optimality has not been proven at the time of this writing despite decades of research.

The first exact algorithm for the G-VRP is by Koç & Karaoglan (2016) who develop a "heuristic based exact solution approach" for the G-VRP, although their method limits the number of consecutive refueling stations visits to one. Nevertheless, even with this limitation on refueling station stops, the results of this study are highly informative and suggest that the G-VRP is extremely difficult to solve. Indeed, the authors' exact branch-and-cut algorithm is unable to solve 18 of the 40 benchmark instances by Erdoğan & Miller-Hooks (2012) with 6 – 20 customers and 2 – 10 refueling stations to optimality within 1 hour of computation time.

Another exact method is introduced by Desaulniers et al. (2016) who develop a branch-price-and-cut algorithm for the E-VRPTW. The authors study 4 different variants that involve either full or partial refueling, and either limiting the number of refueling stops per route to 1 or allowing multiple stops per route. The authors demonstrate the effectiveness of their algorithm by optimally solving instances with up to 100 customers and 21 refueling stations. As their algorithm is a generalization of the G-VRP, it can also solve the G-VRP in a straightforward manner. However, no computational results on any of the G-VRP instances is provided.

As the authors mention, their algorithm for the E-VRPTW is particularly efficient with narrow customer time windows. The G-VRP, on the other hand, has a maximum time limit for every vehicle route which translates into wide time windows for every customer. Thus, it is possible that the algorithm by Desaulniers et al. (2016) is unable to solve G-VRP instances efficiently, because it relies on a labeling method whose performance benefits from narrow time windows. Specifically, more strict customer time windows limit the number of possible ways to visit customer subsets, thus rendering a significant number of otherwise feasible vehicle routes infeasible and providing more opportunities for the labeling algorithm to discard partial tours early on when generating new vehicle routes. It is difficult to find similar, easily exploitable structures from the G-VRP.

In a more recent study, Tahami et al. (2020) develop 3 new formulations and an exact branch-and-cut algorithm for a generalization of the G-VRP, called electric capacitated VRP (ECVRP), in which route duration constraints are removed and customer demands and vehicle capacities are added instead. The authors also design a set of new benchmark instances for the ECVRP and demonstrate the effectiveness of their algorithm by solving instances with up to 100 customers and 21 refueling stations to optimality. The authors also consider an extension of the ECVRP by including route duration constraints – which are essential part of the G-VRP – and conclude that the problem becomes significantly more difficult: the numerical results indicate that the addition of route duration constraints limit the problem size their exact method can solve down to 50 customers compared to the 100 customers without these constraints. Specifically, with the route duration constraints, their algorithm is unable to solve any of the 60 customer instances, and even finding a feasible solution proves to be difficult within 3 hours of computation time. Finally, as the authors point out, their exact method can also solve G-VRP instances in a straightforward way by treating the capacity constraints as route duration ones. However, much like Desaulniers et al. (2016), no meaningful comparison with other exact G-VRP algorithms is provided. This further reinforces that the G-VRP is more difficult than its more constrained generalizations E-VRPTW and ECVRP studied in (Desaulniers et al., 2016) and (Tahami et al., 2020), respectively. The results of the exact branch-and-cut algorithm for the G-VRP by Koç & Karaoglan (2016) provide further evidence in support of this conclusion.

To further appreciate the difficulty of solving the G-VRP with respect to its more constrained generalizations with time windows and customer demands, the stronger multigraph reformulation by Andelmin (2014) can solve most of the small E-VRPTW test instances by Schneider et al. (2014) with 5 – 15 customers and 3 – 7 refueling stations in a few seconds using a commercial mixed-integer linear programming (MILP) solver. However, the same formulation cannot close the optimality gap for many of the

G-VRP benchmark instances by Erdoğan & Miller-Hooks (2012) with 6 – 20 customers and 2 – 10 refueling stations within two hours of computing time using the same MILP solver and computational environment. This difference in problem difficulty can be attributed to the significantly larger solution space of the G-VRP compared to the more tightly constrained E-VRPTW and E-VRPTW-RP.

2.2 Efficient Centralized Resource Allocation

Data envelopment analysis (DEA) (see, e.g. Cooper et al., 2000) is often used to evaluate efficiencies of decision making units (DMUs) that consume several inputs and produce several outputs. Specifically, standard measures of efficiency analysis are typically related to how well a DMU utilizes its resources. A simplified way to measure this would be to divide the DMU's output production by its input consumption or vice versa. In case of multiple inputs and outputs, both the inputs and outputs would have to be aggregated into single measures by giving each input and output a specific weight reflecting its importance and taking weighted sums of the inputs and outputs separately. In this simplified example, a decision maker (DM) wishing to decrease input resources would maximize the benefit to cost ratio (i.e., outputs divided by inputs), while a DM wishing to increase output production would instead minimize the cost to benefit ratio (W. Cook & Hassan, 2020). A greater ratio in each case would indicate that the corresponding DMU is more efficient than one with a smaller ratio.

However, relying on simple efficiency ratios alone is likely not sufficient. Moreover, some of the data may be qualitative instead of quantitative and cannot be straightforwardly transformed into an aggregate measure. To handle different types of inputs and outputs and provide a non-parametric framework for estimating DMUs' production possibilities, more sophisticated models to quantify efficiency have been developed in the DEA literature. The two most cited ones are the CCR by Charnes et al. (1978) and the BCC by Banker et al. (1984). These two models have similar characteristics, but they differ in how the efficiency of a DMU is measured with respect to the other DMUs. The axioms of the CCR and the BCC models each give rise to a different production possibility set (PPS) that corresponds to the possible input/output mixes that the DMUs can attain.

Generally, the selection of the underlying PPS depends on how well it represents DMUs' input and output changes in the application at hand, although the current (observed) input/output mixes of the DMUs typically define the constraints forming the efficient frontier that *envelops* the input and output values that are attainable. Instead of selecting a single PPS, it is likely beneficial to analyze the problem with respect to more than one PPS to get a more comprehensive view of the DMUs' efficiencies and how their input/output mixes behave under different choices of the PPS. Also, instead of computing solutions from the efficient frontier by, for instance, maximizing the DMU efficiencies, it is worthwhile to explore how the input resources consumed by the DMUs change over a number of different alternative solutions and including additional constraints that, for example, limit how much the DMUs' inputs are allowed to increase and/or decrease.

Korhonen & Syrjänen (2004) are among the first to study the similarities between DEA and multi-objective linear programming (MOLP). The authors investigate combining the DEA methodology by including traditional DEA efficiency scores into a MOLP formulation that involves additional constraints preventing individual DMUs from increasing their efficiency. They consider a resource allocation problem in a centralized context where a single DM (or several DMs) control a portfolio of DMUs that correspond to, for example, supermarkets whose inputs are man-hours and floor area and outputs are profit and sales. The objective is to (re-)allocate input resources among the DMUs efficiently by simultaneously maximizing the sums of different outputs and minimizing the sums of different inputs over all the DMUs in the portfolio. The solution set of this multi-objective problem consists of all non-dominated DMU portfolios with respect to the portfolio-level inputs and outputs (i.e., the sums of all different inputs/outputs over all DMUs in the portfolio).

In the example by Korhonen & Syrjänen (2004), it is assumed that no inefficient DMU can increase its efficiency score after resource allocation. While this assumption may seem unrealistic for many practical cases, the resulting non-dominated DMU portfolios can nevertheless provide reasonable short-term approximations. At the other extreme, many of the related studies typically compute portfolio-level solutions from the efficient frontier without limiting the increases in DMUs' efficiency scores (see, e.g., W. D. Cook & Seiford, 2009). In the thus obtained solutions, all inefficient DMUs end up with perfect efficiency scores either by (i) decreasing their input resources while expecting the same output production, or (ii) increasing their (expected) output production without making use of additional input resources. Both approaches seem unrealistic for most practical cases, especially in short-term. In comparison, not allowing the DMUs increase their efficiency scores may lead to better approximations. A more sensible but application dependent approach would be to control DMUs' efficiency scores individually (see, e.g., Aristizábal-Torres et al., 2017).

At present, most research on centralized resource allocation in the context of efficiency analysis typically involves scalarization techniques to first transform the multi-objective portfolio-level problem into a single-objective one, and then proceeds by computing solutions from the efficient frontier (see, e.g., Nasrabadi et al., 2012; Fang, 2013; Lozano, 2014). Also, some of these methods may be interactive such that the DM can change certain parameters during the computation of non-dominated DMU portfolios. For example, the Pareto race method by Korhonen & Wallenius (1988) allows the DM to investigate the efficient frontier of a multi-objective problem. The CUT method by Argyris et al. (2014) operates on a discrete set of multi-criteria alternatives and relies cutting planes, derived from repeated pairwise comparisons by the DM, to exclude a subset of the alternatives. Despite these advances, obtaining a more comprehensive view of all possible non-dominated solutions can be impractical in these methods. Indeed, a unified framework that computes all non-dominated solutions and presents them to the DM in a visually meaningful way has been missing, possibly due to the lack of efficient exact multi-objective optimization algorithms that can handle problems with several objectives and decision variables. Such an approach would make it possible to illustrate, for example, the ranges of all possible resource values that each DMU can achieve over all non-dominated DMU portfolios.

Recent advances in multi-objective programming (MOP) solvers, specifically, the implementation of Benson-type algorithms (Benson, 1998) by Löhne & Weißing (2017) and Dörfler et al. (2020) for MOLP and convex MOP problems, respectively, operate in the objective space rather than in the decision variable space and appear extremely promising for solving large multi-objective problems. While the algorithm for the convex MOP problems necessarily induces approximation errors, these errors are absolute rather than relative and can therefore be adjusted by setting a maximum tolerance value to obtain an appropriate numerical precision. A smaller tolerance leads to an increased computation time, but also gives a better representation of the true non-dominated solution set.

Using an efficient MOP solver also offers further possibilities to augment the resource allocation model by different kinds of constraints. For example, DM's preference information on the importance of unit increases between different outputs can be translated into linear constraints in the decision variable space. Such information could also be included by directly modifying the objective functions instead. As an example, if the preferences are linear and form a compact polyhedron V, the problem can be transformed into an equivalent MOP in which the objective functions are multiplied by the extreme points $\mathbf{ext}(V)$ of the polyhedron V.

Because the introduction of additional constraints restricts the solution set, it also narrows down the ranges of at least some DMUs' possible resource values, thus giving the DM more conclusive recommendations on which DMUs should receive more input resources and identify possible DMUs whose input resources are to be decreased over all remaining non-dominated DMU portfolios. Such a framework would be extremely useful, because it would provide significantly more information to the DM than standard models from the current literature that compute and present only a few solutions without the capability of visualizing the entire solution space and the corresponding DMU- and portfolio-level information.

Finally, while DMUs' efficiency scores under different DEA models in a centralized context can be, in most cases, computed using an LP solver, it may not be entirely obvious how these efficiency scores could be utilized in practice. For example, in the case of allocating additional input resources to a portfolio of DMUs in which the goal is to maximize all portfolio-level outputs while minimizing all portfolio-level inputs, it might seem obvious that the maximum output production would be achieved by allocating the additional resources to the most efficient DMUs as long as they remain inside the PPS. While this strategy appears sensible when maximizing the outputs of single DMUs separately, the use of efficiency scores to guide portfolio-level resource allocation may not be as straightforward. Evaluating the impact of using efficiency scores to guide resource allocation decisions when all DMUs are controlled by a single DM seems not to have been studied in the current literature.

2.3 Discrete Multi-Stage Stochastic Optimization

Multi-stage decision problems under uncertainty are typically studied in the context of stochastic programming (SP) with exogenous uncertainties (Birge & Louveaux, 2011), or in decision analysis (Abbas & Howard, 2015) with two simplifying assumptions: (i) perfect recall, that is, all earlier information is remembered across time when making later decision, and (ii) regularity, meaning that a total temporal order must exist over all decision variables (see, e.g., Koller & Friedman, 2009). While these problems can include decision variables with continuous domains and probability distributions, here the focus is specifically on discrete stochastic decision problems that are modeled with a discrete number of actions at each decision stage and a discrete number of chance events at each chance stage with corresponding probability distributions that sum to one. In addition, these problems have a discrete number of consequences evaluated by utility functions at one or more value stages.

Different outcomes at chance, decision, and value stages are typically called states of the corresponding stage. Also, in an influence diagram representation, stages are typically called nodes in accordance with their graph-like appearance. On the other hand, in a decision tree representation states are typically referred to as nodes at different stages of the tree, while each node in a corresponding decision diagram represents a stage of the decision tree. For concrete examples of both representations and their relations, see (Call & Miller, 1990; Abbas & Howard, 2015).

Discrete multi-stage stochastic optimization problems in the contexts of standard SP and decision analysis consider different kinds of applications and solution methods. Indeed, it can be surprisingly difficult to transform SP problems to decision analysis framework and vice versa. The SP approach typically relies on MILP formulations while the standard approach with decision analysis has usually been dynamic programming. Moreover, experts in one of these two disciplines do not necessarily have deep knowledge of the other one, although both approaches are tackling problems with similar characteristics. Thus, contributing towards a unified framework that could be used to model and solve problems from both disciplines seems worthwhile. Such a framework could bring experts from both domains closer and thus facilitate collaboration and emergence of novel ideas.

The multi-stage nature of these problems in the SP framework typically implies that such a problem involves a sequence of discrete time steps such that at each time step a decision is made and a chance outcome is observed. The goal is to optimize a given objective function with respect to decisions made at the first time step and recourse decisions made at later time steps. This class of SP problems conventionally involves exogenous uncertainties, meaning that probability distributions at chance nodes are unaffected by prior decisions (Ruszczyński & Shapiro, 2003).

In decision analysis, on the other hand, these problems typically involve not only exogenous but also endogenous uncertainties in that probability distributions at chance nodes may be dependent on prior decisions and change accordingly based on the decisions that influence the corresponding chance stages. Figure 1.1 demonstrates the difference between these two types of uncertainties based on the simplified 2-stage stochastic programming example problem from Section 1.1.

Discrete multi-stage decision problems under uncertainty are typically represented by an influence diagram (ID), which has an intuitive graph structure with directed arcs between specific nodes (Howard & Matheson, 2005). Another possibility is to use a decision tree which is a directed acyclic graph whose depth equals the number of stages (Call & Miller, 1990). While both representations have their strengths, it is more convenient to represent large problems as an ID due to its capability of describing problems more concisely, whereas a decision tree representation grows exponentially with the number of stages. IDs can also represent information structures between nodes intuitively by drawing directed arcs between those nodes that have either a conditional or a functional relationship. However, IDs lack some crucial information that is inherently embodied in decision trees: probability distributions of uncertain realizations at chance nodes, possible actions that can be taken at decision nodes, and different outcomes at value nodes (see, e.g., Call & Miller, 1990). On the other hand, decision trees, unlike IDs, cannot represent conditional or functional relationships between different stages that are represented by arcs between the nodes in the corresponding ID. An example ID representing a discrete stochastic decision problem with multiple stages (see pp. 1237 in Lauritzen & Nilsson, 2001) is presented in Figure 2.1.

In this ID, chance nodes h_i represents prior probabilities of being ill at the beginning of each period $i \in \{1,\ldots,4\}$ while u_4 represents the consequences of being ill or healthy at the end of period 4. The remaining nodes correspond to testing outcomes t_i, treatment decisions d_i, and treatment costs u_i, for $i \in \{1,2,3\}$. The objective is to maximize the utility of the consequences at u_4 by deciding each month $i \in \{1,2,3\}$ based on uncertain test results t_i whether to treat ($d_i = 1$) or not ($d_i = 0$). A generalization of this problem which includes testing costs is presented by Hölsä (2020).

Limited memory influence diagram (LIMID) is an interesting generalization of the ID (see Lauritzen & Nilsson, 2001 for a comprehensive definition). In decision problems represented by LIMIDs, all information arcs between nodes must be represented explicitly since perfect recall, the assumption that all prior decisions and chance outcomes are known, no longer holds. Instead, with LIMIDs it is possible to "forget" information from previous stages and information is preserved only between those nodes that are connected by a direct arc (i.e., parent nodes and their immediate predecessors). Thus, each ID is a special case of a corresponding LIMID in which perfect recall is enforced by drawing all arcs explicitly.

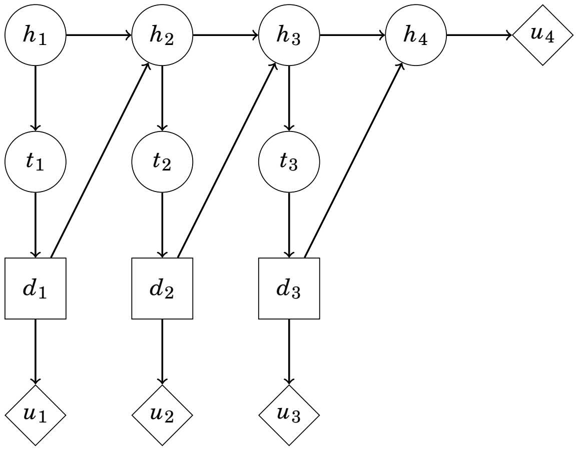


Figure 2.1. An example ID (Lauritzen & Nilsson, 2001). Circles represent chance nodes, squares represent decision nodes, and diamonds represent value nodes.

LIMIDs also relax the regularity assumption which requires a total temporal order between all decision nodes. An example problem not satisfying the regularity assumption involves multiple simultaneous decisions made by agents that cannot communicate with each other. In this example, determining a unique strict total order among all decision nodes is impossible and relaxing the total recall and regularity assumptions makes the decision problem significantly harder to solve. In particular, conventional solution methods for IDs that rely on regularity and total recall such as dynamic programming and message passing algorithms cannot find provably optimal solutions to LIMIDs in general. A simple example problem with simultaneous decisions and imperfect recall represented by a LIMID is shown in Figure 2.2. Note that while the two decisions A^1 and A^2 made simultaneously by the two individual agents are unable to communicate due to the imperfect recall assumption, they both eventually strive to maximize a common utility U.

Also, when the ID in Figure 2.1 is defined as a LIMID, the problem structure changes radically. For instance, in the LIMID representation, the probability distribution of uncertain realizations at the chance node h_2 is influenced only by the chance node h_1 and decision node d_1. This type of uncertainty where prior decisions affect the probability distribution of uncertain realizations is referred to as endogenous. In the same example, the chance node h_1 is not influenced by any prior decision nodes: its probability distribution is independent of the decision variables. Thus, while the uncertainty from h_1 influences the system, specifically, the probability distributions at t_1 and h_2, the system has no influence over h_1. Uncertainty of this kind is referred to as exogenous (see, e.g., Dupacová, 2006). It is worth noting that endogenous uncertainty can also be temporal, meaning that it is related to the timing of decisions rather than prior decisions affecting probability distributions at chance nodes (Herrala, 2020).

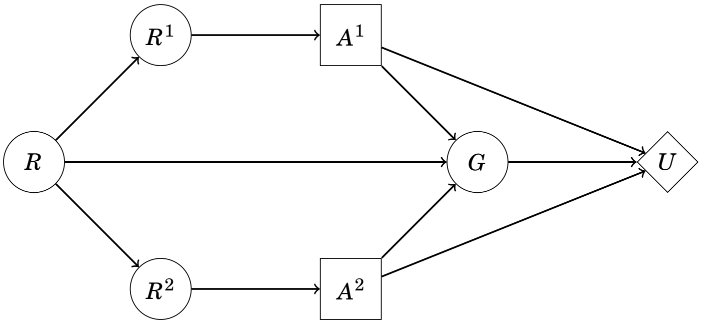


Figure 2.2. LIMID with simultaneous decisions A^1 and A^2 corresponding to two agents unable to communicate, yet striving to maximize the common utility U.

Discrete multi-stage decision problems represented by LIMIDs are further classified as *soluble* and *insoluble* (see Lauritzen & Nilsson, 2001 for the definition of soluble). In general, LIMIDs that satisfy the regularity and total recall assumptions either directly or through problem modifications can be classified as soluble. Specifically, an ID is a special case of LIMID that is soluble. Solubility allows the corresponding decision problem to be solved to optimality with conventional methods that rely on sequentially solving local optimization problems at different nodes. These methods include dynamic programming and different message passing algorithms (Koller & Friedman, 2009). However, for insoluble decision problems (such as in Figure 2.1), the corresponding LIMID is NP-hard, in the same vein as finding optimal (or even an approximate) solutions to IDs with discrete variables is NP-hard in general (Mauá et al., 2013).

3. Research Contributions

The main contributions of the papers [I] – [IV] are in Table 3.1.

1. Paper [I] studies the green vehicle routing problem (G-VRP) that consists of designing optimal routing plans for alternative fuel vehicles (AFVs) to serve a set of customers. An exact algorithm is developed for the G-VRP that can optimally solve problem instances with up to 110 customers and 28 refueling stations in a reasonable time. Moreover, the average optimality gap over instances with 200 – 300 customers is on average ~0.67%. The algorithm is tailored to a novel multigraph reformulation that first removes all refueling nodes and then pre-computes all non-dominated paths between any two customers that may visit any number of refueling stations in between. The algorithm first computes a tight lower bound using a dual ascent method that combines Lagrangian relaxation, subgradient optimization, and column generation. The dual ascent first uses a state-space relaxation based on the ng-paths by Baldacci et al. (2011) before generating elementary vehicle routes to speed up computing a near-optimal dual solution. This is followed by a Simplex-based cut and column generation procedure which is hot-started using the feasible routes generated during the dual ascent method. Number of generated columns is also limited at different stages of algorithm (see, e.g., Larsen, 2004) to improve convergence. Different sets of cutting planes are generated, including subset-row cuts (Jepsen et al., 2008), weak subset-row cuts (Baldacci et al., 2011), and k-path cuts (Laporte et al., 1985). Specifically, a formal proof connecting $k-$path cuts and rank-1 Chvátal-Gomory cuts (Fischetti & Lodi, 2007) is established and a formulation for separating maximally violated $k-$path cuts is introduced. An early version of the algorithm presented in Paper [II] is used to compute upper bounds.
2. Paper [II] also studies the G-VRP and develops a multi-start local search (MSLS) matheuristic for the problem. The MSLS utilizes a

similar multigraph reformulation introduced in paper [I] and develops new variants of conventional heuristic operators by adapting them to work directly on the multigraph. The MSLS uses these operators in a heuristic column generation scheme in which each column corresponds to a feasible route and stores the generated columns in a route pool. This is followed by solving a set partitioning problem in order to find an optimal subset of routes in the route pool that constitutes a feasible G-VRP solution. The novelty of this approach comes from the ability to use the new variants of the conventional heuristic operators to make routing- and refueling decisions simultaneously when constructing new routes by simply selecting different arcs between customers or a customer and a depot. Computational tests demonstrate the effectiveness of the developed MSLS algorithm and how to conveniently trade-off computation time and solution quality by changing a single parameter. The multigraph reformulation is inspired by Andelmin (2014) who introduces a multigraph reformulation for the electric VRP with time windows, customer demands, and partial recharges for the first time along with a strong mathematical formulation.

3. Paper [III] develops a framework for computing all non-dominated resource allocations to a portfolio of decision making units (DMUs) in the context of efficiency analysis. These non-dominated DMU portfolios are computed based on preferences of a decision maker (DM) on, for example, unit value changes of certain inputs and outputs while maximizing the sums of different outputs and minimizing the sums of different inputs over all DMUs in the portfolio. To obtain the whole non-dominated set of DMU portfolios, the problem is solved as a multi-objective programming problem using an efficient Benson-type algorithm (Benson, 1998) that operates in objective space and identifies all non-dominated extreme points and facets of the efficient frontier Hamel et al. (2014). This information is used to compute and illustrate, at individual DMU- and portfolio-level, the ranges of attainable input and output values over all non-dominated DMU portfolios.

   Most importantly, based on the data from previous case studies by Korhonen & Syrjänen (2004) and Lozano (2014), an interesting observation is made in Paper [III] on how resources are allocated among the DMUs in non-dominated DMU portfolios. When maximizing portfolio-level efficiency, resource allocation recommendations are not consistent with conventional efficiency analysis scores. Specifically, in all case studies, resources are taken from both efficient and inefficient DMUs in all non-dominated DMU portfolios, while extra resources are also

Table 3.1. Contributions of Papers [I] – [IV].

	Research objectives	Methodologies	Main contributions
Paper [I]: An exact algorithm for the green vehicle routing problem	Design an exact algorithm for solving large AFV routing problems optimally. Establish a standard for future research and aid heuristic developers calibrate their work against the optimal solutions.	Dual ascent combining Lagrangian relaxation, subgradient optimization, and column generation. Simplex-based cut and column generation. Rank-1 Chvátal-Gomory cut separation for generating k-path cuts. Dynamic programming algorithm for generating columns. MILP for solving reduced SP problems.	First-rate exact algorithm for the G-VRP. Problems with up to 109 customers and 28 stations solved to optimality. With 200 – 300 customer instances, ~0.67% average optimality gap. Proof that connects rank-1 Chvátal-Gomory cuts and k-path cuts. Formulation of separation problem to compute k-path cuts.
Paper [II]: A multi-start local search heuristic for the green vehicle routing problem based on a multigraph reformulation	Develop new building blocks by modifying conventional operators to work directly in the multigraph, allowing simultaneous routing and refueling decisions. Design first-class metaheuristic to demonstrate the new operators' efficiency and promote their use in future research.	Problem decomposition based on a multigraph reformulation. Heuristic column generation using problem-specific local search operators. Multi-start variable neighbourhood search. MILP for optimally combining generated columns by solving an SP problem.	New tailored variants of conventional local search operators that work directly in the multigraph, thus allowing simultaneous routing and refueling decisions. Demonstrating the effectiveness of the multigraph as a basis for future AFV routing algorithms.
Paper [III]: Efficient allocation of resources to a portfolio of decision making units	Develop a framework for portfolio-level resource allocation that includes DMs' preferences. Unify earlier approaches by computing not single but all non-dominated portfolios.	MOLP for computing all non-dominated portfolios. LP to compute and present all possible resource and output ranges over all allocation portfolios.	Framework that unifies prior centralized resource allocation models. Demonstrating the unreliability of conventional efficiency scores in centralized resource allocation.
Paper [IV]: Decision Programming for mixed-integer multi-stage optimization under uncertainty	Combine decision analysis and mathematical programming into a single framework to solve multi-stage decision problems with endogenous uncertainties.	MILP for computing optimal decision strategies strengthened by probability cuts. Exact MOP algorithm for computing all non-dominated decision strategies with more than one utility using valid inequalities to discard dominated solutions.	Framework that can be used to formulate and solve problems from both decision analysis and stochastic programming as MILPs. The MILP optimally solves LIMIDs with endogenous uncertainties and accommodates chance constraints and risk measures such as CVaR.

given to both efficient and inefficient DMUs in the same problems. This happens regardless of whether the DMU efficiencies are allowed to increase or not. This crucial observation suggests that relying on conventional efficiency scores in guiding resource allocation decisions can cause inefficiencies and lead to dominated DMU portfolios.

4. Paper [IV] develops the *Decision Programming* framework for discrete multi-stage decision-making problems under endogenous and exogenous uncertainties that can solve problems from both stochastic programming (SP) and decision analysis using a novel MILP formulation. The new MILP formulation can incorporate different kinds of risk measures, such as conditional value-at-risk (CVaR), directly by adding constraints and/or modifying the objective function. Moreover, in presence of multiple objectives, the MILP formulation can be used as a basis for solving all non-dominated decision strategies with the help of valid inequalities that discard dominated decision strategies until there are no dominated ones remaining. Finally, the MILP formulation provides a more flexible modeling tool compared to standard methods in decision analysis such as dynamic programming to model different kinds of constraints including the CVaR and other risk measures.

The solution methods of the optimization problems in Papers [I] – [IV] seem different, but they use some common modeling techniques, such as exploiting the problem structure by pre-computing parts of the problem and reformulating it in terms of the pre-computed elements from the original model. This is done, for example, in Paper [I] by removing all refueling nodes and instead pre-computing all non-dominated paths between any two customers that may visit any number of refueling stations in between. Similarly, in Paper [II], the results of the heuristic column generation in the first two phases are capitalized in the final phase by formulating a set partitioning problem in order to find the best combination of columns that constitute a feasible solution. In Paper [III], all extreme points of the information set corresponding to the DM's preferences are first computed and arranged as columns of a matrix. The objective functions are then multiplied by this extreme point matrix which gives a simplified formulation. Finally, in Paper [IV], the model becomes much stronger by first pre-computing the probabilities and utilities of all paths, and then reformulating the problem by exploiting this path structure. Thus, while the solution methods in Papers [I] – [IV] may seem different, similar algorithmic techniques are utilized in each paper to facilitate solving the corresponding problems.

4. Discussion

This introductory summary discusses Papers [I] – [IV], each of which focuses on a different optimization problem except Papers [I] and [II] whose main topic is the green vehicle routing problem (G-VRP) by Erdoğan & Miller-Hooks (2012). The problems in Papers [I] – [IV] and their solution methods are presented from two different viewpoints of mathematical optimization. Specifically, in Papers [I] and [II], the focus is on the algorithmic ideas, computational performance, and implementation details. In Papers [III] and [IV], the focus is more on the properties of the corresponding frameworks, with a special emphasis on the problem formulations.

The intention behind Papers [I] and [II] is to construct new efficient solution methods for the G-VRP, both of which combine different algorithmic components that are executed repeatedly. It is therefore not possible to provide formulations that would capture all the complexities of the corresponding algorithms. However, Andelmin (2014) introduces a strong formulation based on a similar but more complex multigraph transformation for the electric VRP with time windows, customer demands, and partial recharges. Because this formulation generalizes the G-VRP, it can also solve G-VRP instances in a straightforward way.

In Papers [III] and [IV], the emphasis is on advancing formulations rather than specialized solution approaches. Specifically, Paper [III], which focuses on the efficient allocation of resources to a portfolio of decision-making units (DMUs), first introduces a new formulation that establishes a connection between the efficient frontier and the efficient solution set of the corresponding multi-objective linear programming (MOLP) problem. The problem is then extended to include decision maker's preferences and formulated as another MOLP whose solution set constitutes all non-dominated DMU portfolios and corresponds exactly to the non-dominated frontier. Thus, both MOLP formulations play key roles in conveying their corresponding information, which are then formalized into theoretical proofs. Most importantly, the LP formulations computing the ranges of possible input/output values based on the extreme points and facets of efficient or non-dominated DMU portfolios demonstrate through examples from the literature how conventional efficiency scores are not reliable in guiding resource allocation decisions. Overall, the main focus is not on developing efficient solution methods but, rather, presenting the new framework and demonstrating the importance of these new ideas.

Similarly, in Paper [IV], which introduces the Decision Programming framework, no special algorithms are developed to solve the new mixed-integer linear programming (MILP) formulations. A MILP solver is used instead that relies on a generic branch-and-cut method. Cutting planes derived from the problem structure are also added through the callback interface of the MILP solver. Specifically, Paper [IV] introduces the Decision Programming framework and its different extensions for the first time. To describe such a seemingly simple but surprisingly complex framework as clearly as possible requires careful planning in terms of notation and definitions. Moreover, because the MILP formulations in Paper [IV] can be challenging to interpret in a straightforward way, it is also necessary to prove their correctness. Therefore, the main focus in Paper [IV] is not to develop a fast algorithm – just presenting the Decision Programming framework alone exceeds the typical 30 page limit.

Although the G-VRP in Papers [I] – [II], the efficient resource allocation approach in Paper [III], and the Decision Programming framework in Paper [IV] have been treated separately in this summary, some applications could benefit from combining at least two of these different types of problems or solution methods. Considering all possible combinations suggests some new ideas and avenues for future research. The final paragraphs of this summary chapter are devoted to discussing the most prominent combinations that could be studied in future.

Most importantly, because the MILP formulation for the Decision Programming framework in Paper [IV] has not yet been studied much, there exists considerable potential for future improvements. The most promising approach involves developing a decomposition method that can reduce both the memory consumption and computation time. Specifically, due to its similarities with many combinatorial optimization problems, the MILP formulation can be transformed into becoming amenable to a column generation approach, possibly coupled with strong cutting plane generators, as further research reveals more about the structure of the formulation. Because a column generation-based algorithm would generate promising strategies dynamically, a full state-space representation would not be necessary, much like in Paper [I] that utilizes cut and column generation.

Regarding the development of new heuristic algorithms for the G-VRP or its generalizations, possibly based on the multigraph reformulation that uses new variants of conventional heuristic operators, it is crucial to assess the performance of each individual operator or a set of operators. This assessment is typically necessary to demonstrate that the rationale behind selecting the operators for the heuristic is sound and quantifiable to some extent. Another reason for such an assessment is to decide which operators are not contributing enough given their time consumption and can be discarded. Some of the important criteria to consider are the number of times each operator is called, the number of times each call results in a better solution, and the average computing time of each operator call. The average improvement of each improving operator call could also be recorded. After running the heuristic over benchmark instances, one possibility to assess the operator performance could be to apply centralized efficiency analysis as in Paper [III]. The DMUs would correspond to different operators or operator sets. The input resource could be the total computing time. Outputs could then be the average improvement in terms of solution quality and the success rate measured as the number of calls that improve a solution divided by the total number of calls. In this setting, not allowing the DMUs to increase their efficiency could be a reasonable assumption, because it is unlikely that the success rate would change drastically. In an optimal DMU portfolio, some operators would get more computing time, while others would have less. In practice, this can be achieved by either increasing or decreasing the number of times an operator is called to achieve the target computing time. To achieve possible computing time ranges over all efficient or non-dominated DMU portfolios, random number scaling gives the correct ranges of all DMUs.

Further possibility could be to consider the G-VRP in either Paper [I] or [II] and the Decision Programming in Paper [IV]. An interesting combination could be to consider the G-VRP under endogenous uncertainties. One such application could be to add timing-related uncertainty to travel times between cities such that the probability distribution of different travel times between every two cities would change depending on the current time of the day. For example, longer travel times would be more likely during the rush hour in the morning when people drive to work and in the afternoon when they typically return from work. Since the G-VRP has a route duration constraint, adding travel time uncertainties would make the problem extremely challenging but more realistic. Other types of uncertainties could be added as well related to, for instance, customer service times or refueling delays.

Finally, a combination of efficient allocation of resources in Paper [III] and Decision Programming in Paper [IV] could be as follows. The problem could be to allocate resources to a portfolio of decision-making units under uncertainties about how well the allocated resources will be utilized by different DMUs. For example, it could be that the probabilities of DMUs producing more or less than the predicted amounts of different outputs may depend on the different levels of possible input resources that can be allocated to each DMU. Moreover, the problem could have multiple stages in which resources are allocated to a number of DMUs and after they generate outputs, more resources could be bought and re-allocated to the same and/or different sets of DMUs. One could also consider continuous probability distributions and continuous decision spaces, with each decision corresponding to the amount of resources allocated.

References

Abbas, A. E., & Howard, R. A. (2015). *Foundations of decision analysis*. Pearson Education Limited, England.

Andelmin, J. (2014). *Optimal routing of electric vehicles* (Master's thesis, Aalto University School of Science). Available at `http://urn.fi/URN:NBN:fi:aalto-201408292544`.

Argyris, N., Morton, A., & Figueira, J. R. (2014). CUT: A multicriteria approach for concavifiable preferences. *Operations Research*, *62*(3), 633–642.

Aristizábal-Torres, D., Castro-Peña, M. Y., Echeverri-Gutiérrez, P. A., & Valencia-Salazar, J. D. (2017). Assessing the efficiency of science, technology and innovation using data envelopment analysis (DEA): The case of Colombia. *Dyna*, *84*(202), 215–220.

Asghari, M., & Mirzapour Al-e-hashem, S. M. J. (2021). Green vehicle routing problem: A state-of-the-art review. *International Journal of Production Economics*, *231*, 107899.

Baldacci, R., Mingozzi, A., & Roberti, R. (2011). New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, *59*(5), 1269–1283.

Balinski, M. L., & Quandt, R. E. (1964). On an integer program for a delivery problem. *Operations Research*, *12*(2), 300–304.

Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management Science*, *30*(9), 1078–1092.

Bard, J. F., Huang, L., Dror, M., & Jaillet, P. (1998). A branch and cut algorithm for the VRP with satellite facilities. *IIE Transactions*, *30*(9), 821–834.

Bellman, R. (1952). On the theory of dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America*, *38*(8), 716.

Benson, H. (1998). An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem. *Journal of Global Optimization*, *13*, 1-24.

Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Heidelberg, Dordrecht, London, New York.

Boland, N., Christiansen, J., Dandurand, B., Eberhard, A., Linderoth, J., Luedtke, J., & Oliveira, F. (2018). Combining progressive hedging with a Frank–Wolfe method to compute Lagrangian dual bounds in stochastic mixed-integer programming. *SIAM Journal on Optimization*, *28*(2), 1312–1336.

Call, H. J., & Miller, W. A. (1990). A comparison of approaches and implementations for automating decision analysis. *Reliability Engineering & System Safety*, *30*(1-3), 115–162.

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research*, *2*(6), 429–444.

Christofides, N., Mingozzi, A., & Toth, P. (1979). *Combinatorial optimization*. John Wiley & Sons, Chichester, UK.

Cook, W., & Hassan, A. (2020). Project portfolio selection under uncertainty: A DEA methodology using predicted and actual frontiers. *Journal of Management Research*, *12*(3), 58–71.

Cook, W. D., & Seiford, L. M. (2009). Data envelopment analysis (DEA) – thirty years on. *European Journal of Operational Research*, *192*(1), 1–17.

Cooper, W. W., Seiford, L. M., & Tone, K. (2000). Data envelopment analysis. *Handbook on Data Envelopment Analysis*, 1–40.

Dammak, N., Dhouib, S., et al. (2019). A review of optimal routing problem for electric vehicle. In *International Colloquium on Logistics and Supply*

Chain Management (LOGISTIQUA) (pp. 1–5). France · Montreuil - Paris, June 12 – 14, 2019.

Desaulniers, G., Errico, F., Irnich, S., & Schneider, M. (2016). Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, *64*(6), 1388–1405.

Dörfler, D., Löhne, A., Schneider, C., & Weißing, B. (2020). A Benson-type algorithm for bounded convex vector optimization problems with vertex selection. *arXiv preprint arXiv:2006.15600*.

Dupacová, J. (2006). Optimization under exogenous and endogenous uncertainty. *University of West Bohemia in Pilsen*, Technical Report.

Eberle, U., & Von Helmolt, R. (2010). Sustainable transportation based on electric vehicle concepts: A brief overview. *Energy & Environmental Science*, *3*(6), 689–699.

Ehrgott, M. (2005). *Multicriteria optimization* (Vol. 491). Springer Berlin · Heidelberg, Germany.

Erdoğan, S., & Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, *48*(1), 100–114.

Fang, L. (2013). A generalized DEA model for centralized resource allocation. *European Journal of Operational Research*, *228*(2), 405–412.

Felipe, Á., Ortuño, M. T., Righini, G., & Tirado, G. (2014). A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transportation Research Part E: Logistics and Transportation Review*, *71*, 111–128.

Fischetti, M., & Lodi, A. (2007). Optimizing over the first Chvátal closure. *Mathematical Programming*, *110*(1), 3–20.

Franke, T., Neumann, I., Bühler, F., Cocron, P., & Krems, J. F. (2012). Experiencing range in an electric vehicle: Understanding psychological barriers. *Applied Psychology*, *61*(3), 368–391.

Goel, V., & Grossmann, I. E. (2006). A class of stochastic programs with decision dependent uncertainty. *Mathematical Programming*, *108*(2-3), 355–394.

Gupta, V., & Grossmann, I. E. (2014). A new decomposition algorithm for multistage stochastic programs with endogenous uncertainties. *Computers & Chemical Engineering*, *62*, 62–79.

Hamel, A. H., Löhne, A., & Rudloff, B. (2014). Benson type algorithms for linear vector optimization and applications. *Journal of Global Optimization*, *59*(4), 811–836.

Hellemo, L., Barton, P. I., & Tomasgard, A. (2018). Decision-dependent probabilities in stochastic programs with recourse. *Computational Management Science*, *15*(3-4), 369–395.

Herrala, O. (2020). *An efficient strategy for solving stochastic programming problems under endogenous and exogenous uncertainties* (Master's thesis, Aalto University School of Science). Available at http://urn.fi/URN:NBN:fi:aalto-202003222575.

Hölsä, O. (2020). *Decision programming framework for evaluating testing costs of disease-prone pigs* (Bachelor's thesis, Aalto University School of Electrical Engineering). Available at http://urn.fi/URN:NBN:fi:aalto-202009295618.

Howard, R. A., & Matheson, J. E. (2005). Influence diagrams. *Decision Analysis*, *2*(3), 127-143.

Jepsen, M., Petersen, B., Spoorendonk, S., & Pisinger, D. (2008). Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research*, *56*(2), 497–511.

Koç, Ç., & Karaoglan, I. (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing*, *39*, 154–164.

Koller, D., & Friedman, N. (2009). *Probabilistic graphical models: Principles and techniques*. MIT Press. Cambridge Massachusetts, London England.

Korhonen, P., & Syrjänen, M. (2004). Resource allocation based on efficiency analysis. *Management Science*, *50*(8), 1134–1144.

Korhonen, P., & Wallenius, J. (1988). A Pareto race. *Naval Research Logistics (NRL)*, *35*(6), 615–623.

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

Laporte, G., Nobert, Y., & Desrochers, M. (1985). Optimal routing under capacity and distance restrictions. *Operations Research*, *33*(5), 1050–1073.

Larsen, J. (2004). Refinements of the column generation process for the vehicle routing problem with time windows. *Journal of Systems Science and Systems Engineering*, *13*(3), 326–341.

Lauritzen, S. L., & Nilsson, D. (2001). Representing and solving decision problems with limited information. *Management Science*, *47*(9), 1235–1251.

Lin, C., Choy, K. L., Ho, G. T., Chung, S. H., & Lam, H. (2014). Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, *41*(4), 1118–1138.

Löhne, A., & Weißing, B. (2017). The vector linear program solver Bensolve – Notes on theoretical background. *European Journal of Operational Research*, *260*(3), 807–813.

Lozano, S. (2014). Company-wide production planning using a multiple technology dea approach. *Journal of the Operational Research Society*, *65*(5), 723–734.

Löhne, A. (2011). *Vector optimization with infimum and supremum* (First ed.). Springer Heidelberg, Dordrecht, London, New York.

Martí, R., Resende, M. G., & Ribeiro, C. C. (2013). Multi-start methods for combinatorial optimization. *European Journal of Operational Research*, *226*(1), 1–8.

Mauá, D. D., De Campos, C. P., & Zaffalon, M. (2013). On the complexity of solving polytree-shaped limited memory influence diagrams with binary variables. *Artificial Intelligence*, *205*, 30–38.

Nasrabadi, N., Dehnokhalaji, A., Kiani, N. A., Korhonen, P. J., & Wallenius, J. (2012). Resource allocation for performance improvement. *Annals of Operations Research*, *196*(1), 459–468.

Papadimitriou, C. H. (2003). *Computational complexity*. John Wiley and Sons Ltd. United Kingdom.

Pessoa, A., Sadykov, R., Uchoa, E., & Vanderbeck, F. (2020). A generic exact solver for vehicle routing and related problems. *Mathematical Programming*, *183*(1), 483–523.

Philip, R., & Wiederer, A. (2010). Policy options for electric vehicle charging infrastructure in C40 cities. *Clinton Climate Initiative*, Harvard Kennedy School, Cambridge Massachusetts.

Ruszczyński, A., & Shapiro, A. (2003). *Stochastic programming*. Elsevier North Holland, Amsterdam.

Schiffer, M., Schneider, M., Walther, G., & Laporte, G. (2019). Vehicle routing and location routing with intermediate stops: A review. *Transportation Science*, *53*(2), 319–343.

Schneider, M., Stenger, A., & Goeke, D. (2014). The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, *48*(4), 500–520.

Solak, S., Clarke, J.-P. B., Johnson, E. L., & Barnes, E. R. (2010). Optimization of R&D project portfolios under endogenous uncertainty. *European Journal of Operational Research*, *207*(1), 420–433.

Tahami, H., Rabadi, G., & Haouari, M. (2020). Exact approaches for routing capacitated electric vehicles. *Transportation Research Part E: Logistics and Transportation Review*, *144*, 102126.

Voss, S., Maniezzo, V., & Stützle, T. (2009). *Matheuristics: Hybridizing metaheuristics and mathematical programming* (Vol. 10). Springer Heidelberg, Dordrecht, London, New York.

The dissertation of Juho Andelmin contains valuable research at a high level. This is also indicated by the fact that the three papers that are already published are published in highly esteemed top journals of the community. It is also my belief that the fourth paper is ready for publication resulting in all four publications of the dissertation being in top journals within Operations Research. This is a strong mark of excellence. The papers are all nicely written, thorough, and good at explaining the problem and the contribution. Each of the papers advances the research in their given area. In conclusion, this dissertation clearly deserves to be published. It is in my belief comparable to the top-level of dissertations within our academic community. It shows the mastery of a broad level of problems and solution techniques and deep insight into the problems being researched in each of the four papers.

Professor Jesper Larsen

Aalto-DD 82/2021

ISBN 978-952-64-0417-2 (printed)
ISBN 978-952-64-0418-9 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)

Aalto University
School of Science
Department of Mathematics and Systems Analysis
www.aalto.fi

BUSINESS + ECONOMY

ART + DESIGN + ARCHITECTURE

SCIENCE + TECHNOLOGY

CROSSOVER

DOCTORAL DISSERTATIONS