UNIVERSITY OF COPENHAGEN
FACULTY OF SCIENCE

PhD thesis

Henrik Carøe Bylling

Bilevel Optimization with Applications in Energy

Advisor: Trine Krogh Boomsma

Handed in: November 10, 2018

Henrik Carøe Bylling
Department of Mathematical Sciences
Faculty of Science
University of Copenhagen
Universitetsparken 5
2100 København Ø
bylling@math.ku.dk // hcbylling@gmail.com

This thesis has been submitted to the PhD School of The Faculty of Science, University of Copenhagen on November 10, 2018.

Supervisor:	Trine Krogh Boomsma, Associate Professor University of Copenhagen, Denmark
Co-Supervisor:	Salvador Pineda, Assistant Professor University of Málaga, Spain
Assessment Committee:	Giovanni Pantuso (chair), Associate Professor University of Copenhagen, Denmark
	Asgeir Thomasgard, Professor Norwegian University of Science and Technology, Norway
	Sonja Wogrin, Assistant Professor Universidad Pontificia Comillas, Spain

ISBN: 978-87-7078-885-4

Preface

This thesis has been prepared in fulfillment of the requirements for the PhD degree at the Department of Mathematical Sciences, Faculty of Science, University of Copenhagen. The work has been carried out under supervision of associate professor Salvador Pineda from September 2015 to May 2016 and associate professor Trine Krogh Boomsma from May 2016 to November 2018.

Acknowledgments

I have been fortunate to have two excellent supervisors during my PhD. First, I want to thank Salvador Pineda for encouraging me to apply for a PhD in the first place, for introducing me to the challenges of bilevel optimization and energy modeling, and for consistently pushing me to do my best. Second, I want to thank Trine Boomsma for stepping up as supervisor when Salva left to return to Spain, for guidance in both academia and career, and for the constant support and helpfulness.

During my PhD, I visited Professor Steven A. Gabriel at the University of Maryland, and we have since collaborated on two papers. I wish to thank Steve for the opportunity to work with him, for sharing his extensive knowledge and experience with me, and for welcoming me and my girlfriend with open arms in Maryland.

In the personal front, I want to thank my girlfriend Karoline for her support throughout the PhD. Also I want to thank both our families for their interest in and support for my PhD.

Contents

Summary

Bilevel optimization problems are very applicable to economic models in general and in energy in particular. However, they are theoretically NP-hard and, thus, solving them implies a large computational burden. This thesis tries to overcome the computational burden of bilevel optimization models, particularly in energy models. Hence, it allows for implementation of long-term strategic problems.

The thesis consists of five chapters. The first is an introduction to the methodologies of optimization and background of the energy market. It is followed by four self-contained chapters.

The second chapter *Efficiently solving Linear Bilevel Programming Problems using Off-the-shelf Optimization Software* provides an overview of existing state-of-the-art solution methods to bilevel optimization methods. Furthermore it presents a new method to linear bilevel optimization problems based on a reformulation to mathematical programs with equilibrium constraints (MPEC). First, a regularization method is used to solve the MPEC using an off-the-shelf, non-linear solver to find a local optimal solution. Local optimal information is then used to reduce the computational burden of solving a mixed-integer reformulation of the MPEC to global optimality using off-the-shelf mixed-integer solvers. Extensive numerical studies are presented using a wide range of randomly generated examples. The results show that our method outperforms existing state-of-the-art methods in terms of computational burden and global optimality.

The third chapter *The Impact of Short-term Variability and Uncertainty on Long-Term Power Planning* considers specific long-term models in power planning. This chapter investigates methods for aggregating data and reducing model size to obtain tractable yet close-to-optimal investment planning decisions. We consider including short-term variability and/or uncertainty and analyze under which circumstances these effects are relevant. In particular, we consider a generation expansion problem and compare various representations of short-term variability and uncertainty of demand and renewable supply. Numerical results are derived from a case study on the Danish power system. Our analysis shows that the inclusion of representative days is crucial for the feasibility and quality of long-term power planning decisions.

The fourth chapter *A Parametric Programming Approach to Bilevel Optimization with lower-level Variables in the upper Level* examines linearly constrained bilevel programming problems in which the upper-level objective function depends on both the lower-level primal and dual optimal solutions. We argue that a parametrized upper-level objective may be non-convex and even discontinuous. However, when the upper-level objective is affine in the lower-level primal optimal solution, the parametric function is piece-wise affine. We show how this property facilitates the application of parametric programming and demonstrate how the approach allows for decomposition of a separable lower-level problem. When the upper-level objective is bilinear in the lower-level primal and dual optimal solutions, we also provide an exact linearization method that reduces to a single-

level mixed-integer linear programming (MILP) formulation of the bilevel problem. We present two numerical case studies of strategic investment in electricity markets and we benchmark the proposed method against state-of-the-art MILP and non-linear solution methods for bilevel optimization problems. Results indicate computational advantages over several standard solvers. We furthermore show that the parametric programming approach succeeds in solving problems to global optimality for which standard methods can fail.

The fifth chapter *A Parametric Programming Approach to Bilevel Transmission Investment Problems in Power* is an extension of Chapter 4 where the proposed parametric programming method is applied to a transmission investment problem. Specifically, we formulate the stochastic transmission expansion problem of a merchant investor collecting congestion rents that are determined by the differences between nodal market prices. The corresponding bilevel program can be recast as an MPEC, but does not allow for linearization and reformulation by mixed-integer linear programming. Instead, the parametric programming approach from chapter 4 is adapted to handle binary upper-level variables.

Resumé

To-niveau optimeringsproblemer er meget anvendelige i økonomiske modeller generelt og især indenfor energi, men teoretisk set er de NP-svære og derfor er løsningen af dem beregningsmæssigt svært. Denne afhandling prøver at overvinde beregningsbyrden ved to-niveau optimeringsproblemer især inden for energimodeller og tillader som resultat implementeringen af langsigtede strategiske problemer.

Denne afhandling består af fem kapitler. Det første er en introduktion til optimeringsmetoder og baggrunden i energimarkeder. Derefter følger fire selvstændige kapitler.

Det andet kapitel med titlen *Efficiently solving Linear Bilevel Programming Problems using Off-the-shelf Optimization Software* giver et overblik over eksisterende state-of-the-art løsningsmetoder til to-niveau optimeringsproblemer. Derudover præsenteres en ny løsningsmetode til lineære, to-niveau optimeringsproblemer baseret på omformulering til et matematisk program med ligevægtsbetingelser (MPEC). Først bliver en reguleringsmetode brugt til at løse et MPEC med off-the-shelf heltalsløsningssoftware. Vi præsenterer omfattende numeriske studier med et bredt udvalg af tilfældigt genererede eksempler. Resultaterne viser at vores metode giver bedre resultater end eksisterende state-of-the-art metoder med hensyn til beregningsbyrden og globalt optimalitet.

Det tredje kapitel *The Impact of Short-term Variability and Uncertainty on Long-Term Power Planning* betragter specifikt langsigtede planmodeller med energi. Kapitlet undersøger metoder til at aggregere data og reducere størrelsen af modellen for at opnå løselige investeringsbeslutninger, som er tæt på optimale. Vi betragter inklusionen af kortsigtet variabilitet og/eller usikkerhed og analyserer under hvilke omstændigheder disse effekter er relevante. Specifikt betragter vi et problem om udvidelse af elproduktionen og sammenligner forskellige repræsentationer af forbrugets og den fornyende energis kortsigtet variabilitet og usikkerhed. Vi præsenterer numeriske resultater fra et casestudie i det danske elsystem. Vores analyse viser at inklusionen af repræsentative dage er afgørende for en mulig kvalitetsløsning i langsigtede energiplanlægningsbeslutninger.

Det fjerde kapitel *A Parametric Programming Approach to Bilevel Optimization with lower-level Variables in the upper Level* undersøger lineært betingede to-niveau programmeringsproblemer, hvor det øverste niveaus objektfunktion afhænger af både primære og duale optimale løsninger fra det nederste niveau. Vi argumenterer for at en parametriseret objektfunction i det øverste niveau kan være ikke-konveks og endda diskontinuert. Dog vil den parametriske funktion være stykkevis lineær hvis det øverste niveaus objektfuntion er affin med hensyn til den primære optimale løsning i det nederste niveau. Vi viser hvordan denne egenskab gør det muligt at anvende parametrisk programmering, og vi demonstrerer hvordan en sådan tilgang gør det muligt at dekomponere det nederste niveaus problem, hvis det er seperabelt. Hvis det øverste niveaus objektfunction er bilineær i det nederste niveaus primære og duale optimale løsninge, så giver vi også en eksakt lineariseringsmetode, der reducerer to-niveau problemet til et enkelt-niveaus, lineært heltalsproblem. Vi præsenterer to numeriske casestudier om strategiske investeringer i

elmarkeder, og vi benchmarker den foreslåede metode mod state-of-the-art heltals- og ikke-lineære metoder til løsning af to-niveau optimeringsproblemer. Resultaterne viser en beregningsmæssig fordel over adskillige standard løsningsmetoder, og vi viser endvidere at parametrisk programmeringstilgangen lykkedes med at løse problemer, som standardmetoder ikke kan.

Kapitel 5 *A Parametric Programming Approach to Bilevel Transmission Investment Problems in Power* er en udvidelse af kapitel 4, hvor den foreslåede parametrisk programmeringsmetode er anvendt til et investeringsproblem om transmission. Specifikt formulerer vi et stokastisk transmissionudvidelsesproblem fra en profitmaksimerende investors perspektiv. Investoren profiterer af trængselsafgiften givet ved forskellen mellem knudepunktspriser. Det tilsvarende to-niveaus optimeringsproblem kan blive omformuleret til et MPEC, men tillader ikke linearisering og omformulering til et lineært heltalsproblem. I stedet bliver den parametrisk programmeringstilgang fra kapitel 4 tilpasset til at håndtere binær variable i det øverste niveau.

List of Papers

This thesis is based on four papers:

Pineda, S., Bylling, H., and Morales, J. (2018). Efficiently solving linear bilevel programming problems using off-the-shelf optimization software. *Optimization and Engineering*, 19(1):187–211.

Bylling, H., Pineda, S., and Boomsma, T. (2018). The impact of short-term variability and uncertainty on long-term power planning. *Annals of Operations Research*, Accepted for publication.

Bylling, H., Gabriel, S., and Boomsma, T. (2018). A parametric programming approach to bilevel optimization with lower-level variables in the upper level. *Journal of the Operational Research Society*, 2nd round review.

Bylling, H., Boomsma, T., and Gabriel, S. (2018). A parametric programming approach to bilevel transmission investment problems in power. In *M. R. Hesamzadeh, J. Rosellon, I. V., editors, Transmission Network Investment in Liberalized Power Markets*. Springer (Lecture notes in Energy).

Chapter 1

Introduction

This chapter provides an overview of the four problems addressed in Chapters 2, 3, 4 and 5, including the relevant background and the applied methodology. First, we introduce optimization and bilevel optimization problems in general, as this subject is present in Chapters 2, 4 and 5. In Chapters 2 and 4, solution methods to bilevel optimization problems are presented in more detail. Second, we present a short overview of power markets and the connection to optimization. Two standard problems in power markets, the economic dispatch and optimal power flow, are presented. Finally, in Section 1.3, we introduce some applications of BPPs to power markets and the computational challenges these problems pose. As an example, a generation expansion problem is presented. The four main chapters are self-contained articles, except for the common reference list and appendix at the end of the thesis.

1.1 Bilevel Optimization

This section starts with defining optimization in general, before defining stochastic optimization and bilevel optimization.

1.1.1 Optimization

Optimization is the process of selecting a best element (or decision) in a set with regard to a specified criterion. Given a function $f : P \to \mathbb{R}$ with $P \subseteq \mathbb{R}^n$, we wish to find an element $x^* \in P$ such that $f(x^*) \leq f(x)$ for all $x \in P$ in the case of minimization, or $f(x^*) \geq f(x)$ for all $x \in P$ in the case of maximization. In general, this thesis considers minimization problems. Such an $x^* \in P$ is called an optimal solution while x is called the vector of decision variables. In constrained optimization, the feasible set P is often described by inequality and/or equality constraints, $g(x) \leq 0$ and/or $h(x) = 0$ with $g : \mathbb{R}^n \to \mathbb{R}^q$ and $h : \mathbb{R}^n \to \mathbb{R}^r$. The optimization problem is usually written as (Boyd and Vandenberghe, 2009)

$$\begin{aligned} \min \quad & f(x) && \text{(1.1a)} \\ \text{s.t.} \quad & g(x) \leq 0 && \text{(1.1b)} \\ & h(x) = 0. && \text{(1.1c)} \end{aligned}$$

Note that 0 is used for a vector of all zeroes of appropriate dimensions. The complexity of the optimization problem is determined by the functions f, g and h. For example, if f, g

and h are linear functions, (1.1) is a linear programming problem. If f and g are convex functions and h is an affine function, then (1.1) is a convex optimization problem. Note that linear optimization problems are a special case of convex optimization problems.

1.1.2 Stochastic Optimization

As part of the three functions f, g and h there are fixed, problem-dependent parameters that are exogenous to the optimization model. The optimal solution naturally depend on these parameters. In some cases, however, these parameters are stochastic and uncertain, and rather than a fixed value, we have a probability distribution to describe them. As an example, stochastic parameters in this thesis are production levels of non-dispatchable renewable power sources in Chapter 3 and uncertain demand in Chapter 5. To cope with this uncertainty, we introduce stochastic optimization. A simple and common model of this type is a two-stage stochastic optimization problem where the decision variables are divided into two time dependent stages, a first stage and a second stage (Birge and Louveaux, 2011). The first-stage variables are decisions taken before realization of the stochastic parameters while the second-stage variables are decisions taken after the outcome of the stochastic parameters is known.

To describe a two-stage stochastic programming problem in more detail, let ω be the outcome of random events and Ω be the set of all outcomes. The second-stage random data is dependent on ω and denoted by random variable $\xi(\omega)$ with a corresponding probability measure. For simplicity, we consider a linear two-stage stochastic program. The general formulation is given by (Birge and Louveaux, 2011):

$$\min \quad c^T x + E_\xi[\min q(\omega)^T y(\omega)] \tag{1.2a}$$
$$\text{s.t.} \quad Ax = b \tag{1.2b}$$
$$T(\omega)x + Wy(\omega) = h(\omega) \tag{1.2c}$$
$$x \geq 0, y(\omega) \geq 0. \tag{1.2d}$$

Here, $x \in \mathbb{R}^{n_1}$ is the first-stage decision variables, $y(\omega) \in \mathbb{R}^{n_2}$ is the second-stage decision variables. $\xi(\omega) = (q(\omega), h(\omega), T_1(\omega), T_2(\omega), \ldots)$ contains the second-stage data, where $T_1(\omega), T_2(\omega), \ldots$ are the rows of $T(\omega)$. The objective function in (1.2a) minimizes the first-stage costs, given by $c^T x$, and the expected (optimal) second-stage costs, given by $E_\xi[\min q(\omega)^T y(\omega)]$. Note, that in the second-stage expected costs, there is also a minimization, since we have decision $y(\omega)$ to be made after the realization of the random event. We have first-stage constraints (1.2b) and second stage constraints (1.2c). The parameters in the latter depend on the random event ω and the first-stage decision x.

In stochastic programming, the stochastic parameters are usually assumed with discrete probability distributions, i.e., where the outcomes of the parameters can be described as a finite number of scenarios with corresponding probabilities. We write the K scenarios as $\xi_1, \xi_2, \ldots, \xi_K$ with corresponding probabilities $\rho_1, \rho_2, \ldots, \rho_K$. The expectation in the (1.2a) then becomes a weighted sum over all scenarios and the two-stage stochastic program can be solved as one large linear programming problem. In the cases where ξ has infinitely many outcomes, scenarios can be generated to match the moments of the probability distribution, enabling an approximated solution using the above solution approach (Birge and Louveaux, 2011).

1.1.3 Bilevel Optimization

In this thesis, we focus on bilevel optimization problems, defined as optimization problems for which some of the variables are constrained to be optimal solutions to another optimization problem (Dempe et al., 2015). A bilevel optimization problem has two levels of optimization which we write as

$$\min f(x, y^*) \tag{1.3a}$$
$$\text{s.t. } g_1(x, y^*) \leq 0 \tag{1.3b}$$
$$h_1(x, y^*) = 0 \tag{1.3c}$$
$$y^* \in \text{argmin}\{f_2(x, y) \tag{1.3d}$$
$$\text{s.t.} \quad g_2(x, y) \leq 0 \tag{1.3e}$$
$$h_2(x, y) = 0\}. \tag{1.3f}$$

The decision variables are divided into two sets, upper-level variables $x \in \mathbb{R}^{n_1}$ and lower-level variables $y \in \mathbb{R}^{n_2}$. We likewise have two sets of constraints: the upper-level constraints (1.3b) and (1.3c) and the lower-level constraints (1.3e) and (1.3f). However, the problem has a hierarchical structure, as the upper-level problem includes the optimal solutions to the lower-level problem, as indicated in the constraint (1.3d) but not vice versa. Instead, in the lower-level problem, the upper-level variables are fixed parameters and not decision variables.

With a bilevel optimization problem as (1.3), the constraint region is defined as (Bard, 1998)

$$\Omega = \{(x, y) \in \mathbb{R}^{n_1 \times n_2} : g_1(x, y) \leq 0, h_1(x, y) = 0, g_2(x, y) \leq 0, h_2(x, y) = 0\}, \tag{1.4}$$

i.e., the points that satisfy both the upper-level and lower-level constraints. However, since we have two optimization problems, we rather consider the projection of the constraint region on $\mathbb{R}^{n_1}$ defined as the upper-level feasible region which is

$$\Omega(X) = \{x \in \mathbb{R}^{n_1} : \exists y : (x, y) \in \Omega\}, \tag{1.5}$$

i.e., the set of upper-level variables which renders the lower-level problem feasible. As the lower-level considers the upper-level variable as a parameter, the lower-level feasible region is defined for x fixed as

$$\Omega(x) = \{y \in \mathbb{R}^{n_2} : g_2(x, y) \leq 0, h_2(x, y) = 0\}. \tag{1.6}$$

As indicated by this definition, the optimal solution of the lower-level depends on the upper-level variables. That is, for each feasible x, the so-called lower-level rational reaction set is

$$\Pi(x) = \{y^* \in \mathbb{R}^{n_2} : y^* \in \text{argmin}(f_2(x, y) : y \in \Omega(x))\}, \tag{1.7}$$

i.e., the set of optimal solutions for a fixed x. Finally, the feasible region of (1.3) is also called the inducible (or induced) region (IR) and is defined as (Bard, 1998)

$$IR = \{(x, y) \in \mathbb{R}^{n_1 \times n_2} : x \in \Omega(X), y \in \Pi(x)\}. \tag{1.8}$$

The IR is defined as the set of feasible upper-level variables x and the corresponding lower-level optimal solutions, given by the set $\Pi(x)$.

With this definition, the bilevel optimization problem (1.3) can be reformulated as

$$\min \quad f_1(x, y) \tag{1.9a}$$
$$\text{s.t.} \quad (x, y) \in IR. \tag{1.9b}$$

For the bilevel optimization problem to be well-defined, we consider the lower-level rational reaction set $\Pi(x)$ in further detail. This is the set of solutions to an optimization problem, and thus not necessarily a singleton for a fixed x. However, the problem (1.3) is not well-defined for multiple solutions to the lower-level problem. To overcome this, two solutions exist: the pessimistic and the optimistic solutions (Dempe et al., 2015). The optimistic solution to a bilevel optimization problem is determined by a solution which minimizes the upper-level objective function. This is easy to implement since no restrictions have to be made on $\Pi(x)$. The pessimistic solution considers a $y^* \in \Pi(x)$ that maximizes the upper-level objective function, that is, the worst-case solution to the lower-level problem given an upper-level decision x.

For a solution to a bilevel optimization problem to exist, we impose the following assumptions: f_1 is continuous, $\Pi(x)$ is bounded and Ω is non-empty and bounded (Bard, 1998). These are natural assumptions as the opposite could render the bilevel optimization problem infeasible or unbounded. With these assumptions, the IR can be proven to be a bounded and closed set. With IR as a bounded and closed set, (1.9) is equivalent to minimizing a continuous function over a compact set and the existence of a solution is given by Weierstrass' extreme value theorem (Bard, 1998).

The history of bilevel optimization problems traces back to the so-called Stackelberg games (Stackelberg, 1934) describing a strategic game in economics with two players: a leader who moves first and a follower who moves subsequently. These two players solve the upper-level and lower-level optimization problems in the bilevel optimization problem, respectively, and as such the terms are used interchangeably in this thesis. In a game theoretic sense, the leader has perfect information about the follower, and thus can anticipate the follower's action. From the point of view of the leader, the problem then chooses x and thereby the follower chooses y and, as a result, the joint solution is the best possible for the leader.

1.1.4 Computational Complexity

In computational complexity theory, computational problems are evaluated by their inherent difficulty and are subsequently divided into complexity classes. Two main complexity classes are P and NP (Cormen et al., 2009). P is defined as the class of problems for which computational time is polynomial in the input size of the problem. Loosely stated, P is the class of problems that are tractable and can be solved efficiently (Goldreich, 2008). NP is defined as the class of problems for which a solution can be verified in a time polynomial in the input size. Clearly, $P \subseteq NP$ but whether or not $P = NP$ remains an unsolved problem (Cormen et al., 2009). As a further (informal) classification, problems are called NP-hard if they are at least as hard as the hardest problems in NP. This loose definition also means that NP-hard problems are not necessarily in NP. NP-hard problems are considered some of the computationally hardest problems to solve (Cormen et al., 2009).

The bilevel optimization problem is proven be NP-hard (Jeroslow, 1985), meaning that solving a bilevel optimization problem is a heavy computational burden. For this reason, developing fast solution methods to bilevel optimization problems has significant practical implications, although, to a theoretical limit, as per the NP-hard classification. In this thesis, Chapter 2 proposes a general method to solve linear bilevel optimization problems using standard solvers and proves effectiveness through extensive numerical studies. Chapter 4 presents a more specialized algorithm that allows for decomposition of certain bilevel optimization problems and demonstrates computational advantages over state-of-the-art methods.

1.1.5 MPEC Reformulation

A related problem to the bilevel optimization problem is a Mathematical Program with Equilibrium Constraints (MPEC). The MPEC is a two-level optimization problem with an upper-level optimization problem and, contrary to the bilevel optimization problem, a lower-level complementary problem. A complementarity problem, or equilibrium problem, is a system of equations and inequalities that relates to equilibrium or optimality conditions (Gabriel et al., 2013). The bilevel optimization problem is equivalent to an MPEC if the lower-level problem can be replaced by necessary and sufficient Karush-Kuhn-Tucker (KKT) optimality conditions (Gabriel et al., 2013).

The KKT conditions are a set of equations and inequalities that determine the optimal solutions of an optimization problem (Boyd and Vandenberghe, 2009). These conditions are necessary if the optimization problem satisfies certain constraint qualifications. One of the most common constraint qualifications is the linear independence constraint qualification, where the gradients of $h(x) = 0$ and the binding constraints of $g(x) \leq 0$ are supposed to be linear independent. Linear optimization problems qualify for this constraint qualification. The KKT conditions are sufficient if the Hessian of the Lagrangian is positive-definite (Gabriel et al., 2013). With necessary and sufficient KKT conditions, the bilevel optimization problem can be reformulated as an MPEC. In particular, (1.3) becomes

$$\min f(x,y) \tag{1.10a}$$
$$\text{s.t. } g_1(x,y) \leq 0 \tag{1.10b}$$
$$h_1(x,y) = 0 \tag{1.10c}$$
$$g_2(x,y) \leq 0 \tag{1.10d}$$
$$h_2(x,y) = 0 \tag{1.10e}$$
$$\lambda \geq 0 \tag{1.10f}$$
$$\nabla_y f_2(x,y) + \lambda \nabla_y g_2(x,y) + \mu \nabla_y h_2(x,y) = 0 \tag{1.10g}$$
$$\lambda g_2(x,y) = 0, \tag{1.10h}$$

where λ holds the dual variables to (1.3e) and μ holds the free dual variables to (1.3f). Note that we use the notation ∇_y for the gradient with respect only to the lower-level variables y. The constraints (1.10h) are the complementary slackness constraints, making (1.10) a non-linearly constrained problem, irrespective of the original constraints.

Many popular solution approaches to bilevel optimization problems are based on reformulating the problem to an MPEC. Chapter 2 provides a detailed overview of existing solution methods and proposes a method to efficiently solve linear bilevel optimization problems. Chapter 4 proposes a method that does not reformulate the bilevel optimization problem to an MPEC but rather solves the bilevel optimization problem using parametric programming.

1.2 Power Markets and Optimization

Electricity markets are similar to other markets in many ways. Consumers and producers submit buying and selling offers to the market, and then the market clearing settles the price and the volume e.g. by matching an aggregated supply curve with an aggregated demand curve. However, electricity is characterized by physical laws that further complicates the market process. Electricity cannot be efficiently stored and, furthermore, an imbalance between demand and supply leads to undesired blackouts or melt downs of e.g., cables. Thus, the supply and demand must match at all times, leading to a need for real-time optimization of the market. To handle uncertainty of production and/or demand under such constraints, many markets have implemented sequential market clearing. First, a day-ahead spot market is cleared the day ahead of production, to allow scheduling of slow units that need several hours to turn on/off or change production level. Then a subsequent intra-day or balancing market is run close to real-time operations in order to handle imbalances that have occurred in the meantime, often changes in demand or wind power production etc. This is the way the markets function in e.g. U.K., Germany and the Nordic Countries (Nord Pool AS, 2017).

Throughout this thesis, we assume inelastic demand of power. This is a common assumption in energy economics, meaning that consumers do not change consumption in response to price changes (Sorokin et al., 2012). In many power markets, risk connected to price changes is not passed on to the consumer, and as such, the consumer has no incentive to change consumption according to electricity prices.

The market-clearing optimization is handled by a system operator (SO) that receives the producers supply bid and then dispatches the production units to meet demand at the lowest cost. Because of inelastic demand, cost minimization is equal to maximizing social welfare or minimizing social costs (Gabriel et al., 2013). The market-clearing optimization problem is called the economic dispatch (ED) where the solution is to dispatch the generation units in the order of the lowest generation costs to the highest cost. To model the ED problem, we introduce a set of time periods $\mathcal{T} = \{1, \dots, T\}$ and a set of generation units $\mathcal{G} = \{1, \dots, G\}$. The simplest model has linear cost functions for the generation units, i.e. units has marginal cost c_g, and a demand for each time period of d_t. The decision variables are the production of each generation unit in each time period, given by p_{gt}, and limited by maximum capacity p_g^{max}. The formulation of the ED is (Sorokin et al., 2012)

$$\min \quad \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} c_g p_{gt} \tag{1.11a}$$

$$\text{s.t.} \quad \sum_{g \in \mathcal{G}} p_{gt} = d_t \quad \forall t \in \mathcal{T} \tag{1.11b}$$

$$0 \leq p_{gt} \leq p_g^{max} \quad \forall g \in \mathcal{G}, t \in \mathcal{T} \tag{1.11c}$$

The constraint (1.11b) is the balancing constraint for each time period, ensuring supply is equal to demand, and (1.11c) is the capacity constraint for each generation unit. Both the objective function and constraints are linear functions, and so (1.11) is a linear programming problem. In a linear programming problem, we have strong duality (Boyd and Vandenberghe, 2009). Hence for an optimal primal solution, we have an optimal dual solution with each variable corresponding to a constraint. The dual solution to the balancing constraint (1.11b) is the price of electricity in each time period.

The ED problem in (1.11) does not account for transmission of electricity to and from different generation and demand locations. Such transportation is facilitated by the power network which is subject to physical constraints on transmission cables and transformers. Electricity current running through a cable can both be alternating current (AC) or direct current (DC). From an optimization point of view, however, DC is the simplest to model and often AC networks are approximated by a DC formulation. When taking a power network into consideration, the ED becomes an Optimal Power Flow (OPF) problem. To model this, we introduce a set of nodes, $\mathcal{N} = \{1, \ldots, N\}$. At each node i, we have demand for each time period, d_{it}, and a set of units, $\mathcal{G}(i)$. In a DC power network, the flow on each line is determined by the voltage angles at the sink and source node and by the susceptance of the transmission line, given by parameter B_{ij} (where i is the source node and j is the sink node). The voltage angles are determined by decision variables θ_{it} and the corresponding flow is defined as variable f_{ijt} from node i to node j. $\mathcal{N}(i)$ is the set of all nodes connected to node i. The OPF model is formulated as (Sorokin et al., 2012)

$$\min \quad \sum_{t\in\mathcal{T}}\sum_{g\in\mathcal{G}} c_g p_{gt} \tag{1.12a}$$

$$\text{s.t.} \quad \sum_{g\in\mathcal{G}(i)} p_{gt} - \sum_{j\in\mathcal{N}(i)} f_{ijt} = d_{it} \quad \forall t \in \mathcal{T}, \quad \forall i \in \mathcal{N} \tag{1.12b}$$

$$f_{ijt} = B_{ij}(\theta_{it} - \theta_{jt}) \quad \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{N}(i) \tag{1.12c}$$

$$0 \leq p_{gt} \leq p_g^{max} \quad \forall t \in \mathcal{T}, \forall g \in \mathcal{G} \tag{1.12d}$$

$$-f_{ij}^{max} \leq f_{ijt} \leq f_{ij}^{max} \quad \forall t \in \mathcal{T}, \forall i \in \mathcal{N}, \forall j \in \mathcal{N}(i) \tag{1.12e}$$

$$-\pi \leq \theta_{it} \leq \pi \quad \forall t \in \mathcal{T}, \forall i \in \mathcal{N} \tag{1.12f}$$

$$\theta_{it} = 0 \quad \forall t \in \mathcal{T},\ i = \texttt{ref} \tag{1.12g}$$

The objective function is again to minimize production costs. Meeting the demand at each node is enforced by constraint (1.12b) where the flow to and from the nodes is included. The power flow from node i to node j is defined in constraint (1.12c). The power generation is limited by capacity for each generating unit in constraint (1.12d). The flow through each line is constrained by (1.12e). The voltage angles at each node is restricted by constraint (1.12f) and finally (1.12g) defines a reference node in the network to be zero.

Both (1.11) and (1.12) are simple models of the ED and OPF. We could add physical constraints on the generation units such as ramp limits on the production changes from a time period to the next. Due to the increase of non-dispatchable renewable energy sources, such as wind and photo-voltaic, uncertainty of the power production has an increasing effect on operation and the market clearing. In short-term operations, such details are often added to the model, resulting in a unit commitment model or a stochastic unit commitment model, using stochastic programming. In the unit commitment model, ramping limits as well as minimum up- and downtimes for the generating units are added to the model (Sorokin et al., 2012). In a stochastic unit commitment problem, the production from wind and or PV energy is first realized after a dispatch schedule is obtained, meaning some adjustments in dispatch is needed to balance the stochastic production (Pritchard et al., 2010).

In economics, market-clearing prices and quantities constitute an equilibrium determined by the intersection of the aggregated demand and supply curves. The demand curve here is a vertical line since we have assumed inelastic demand. The aggregated supply curve, however, is supplied by the producers' supply bids. Producers have market power if they can affect the market outcome by changing their supply bid in order to obtain a greater profit. In this thesis, however, we assume perfect competition, making all producers price-takers, i.e., unable (or unaware of their ability to) affect the market by changing their supply. In this case, the producers bid their marginal cost and the supply curve is equal to the cost function of producing. Perfect competition is a common assumption in economic models (Sorokin et al., 2012). With supply curves equal to production cost functions, it can be shown that finding an equilibrium has exactly the same outcome as if the market is solved using social welfare maximization (Gabriel et al., 2013).

1.2.1 Long-Term Planning

Long-term decisions, e.g., investments in generation and/or transmission, in power markets has to include a sufficient amount of data to represent operation and market clearing throughout the planning horizon. These data are usually historical or simulated data of demand, stochastic production, etc., with a typical time resolution of hours. Long-term planning models are categorized as static if they include decisions for a single future year and dynamic if they involve several decisions over a span of years. In both cases, including hourly data for the entire planning horizon can result in intractable models. Instead, representative hours or days are used, where each hour or day is weighted by the number of hours or days that it represents. Including more representative hours or days results in more precise models at the expense of tractability. This trade-off is examined in detail in Chapter 3 with focus on power markets with increased penetration of renewable non-dispatchable power sources and the challenges therein.

As an example of a long-term planning model, we extend the ED model from (1.11) to include a long term decision on capacity expansion. We assume the perspective of a central planer who wishes to build new capacity of one or more power generation units. We divide the generation units into subsets of existing units, $\mathcal{G}_E$, and candidate units, $\mathcal{G}_C$. With these subsets, we have $\mathcal{G} = \mathcal{G}_E \cup \mathcal{G}_C$. The objective is to minimize the total costs of the power market, including investment costs from new capacity. The investment costs are linear with marginal cost c_g^I and the decision variables x_g are the investment variables for $g \in \mathcal{G}_C$. The planning problem is then

$$\min \quad \sum_{g \in \mathcal{G}_C} c_g^I x_g + \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} c_g p_{gt} \tag{1.13a}$$

$$\text{s.t.} \quad \sum_{g \in \mathcal{G}} c_g p_{gt} = d_t \quad \forall t \in \mathcal{T} \tag{1.13b}$$

$$0 \leq p_{gt} \leq p_g^{max} \quad \forall g \in \mathcal{G}_E, \forall t \in \mathcal{T} \tag{1.13c}$$

$$0 \leq p_{gt} \leq x_g \quad \forall g \in \mathcal{G}_C, \forall t \in \mathcal{T}. \tag{1.13d}$$

This is a linear program and relatively simple and could be extended with more detailed constraints or include a power network as (1.12). If so, including hours to correctly represent the planning horizon could result in an intractable problem as seen in Chapter 2.

1.3 Bilevel Models in Power Markets

In the long-term planning model in (1.13), the perspective is of a central planner whose objective is aligned with the market clearing objective of minimizing costs. A long-term planning model with another perspective, e.g. a strategic decision from a market player, cannot be formulated as a similar single-level model. Instead, we adopt bilevel optimization and let a leader take a strategic decision with the market clearing acting as a follower. This is possible, since as we have seen in Section 1.2, market-clearing in power markets can be formulated as an optimization problem. The bilevel optimization problem has a market-clearing problem as the lower-level problem and a strategic decision maker in the upper-level problem anticipating the market outcome. Such strategic decisions could be supply bidding, investment decisions, policy adjustments etc. Strategic decisions are fixed parameters in the lower-level problem and the market feedback of any decision is captured in the structure of the bilevel model. For example, if an investment is made in a new generation unit in order to profit from selling power, this new capacity will potentially change the order of the market dispatch. Consequently, the price that determines sales profits also changes.

A common determinant for a strategic decision is the price of electricity in all time periods of the planning horizon. As noted above, the price can be found as the dual variable to the balancing constraint of a market-clearing problem. When the lower-level problem meets the conditions of strong duality, the optimal dual solution can be interpreted as the price of electricity and be included in the upper-level problem. However, including lower-level dual solutions in the upper-level problem introduces some complications, especially if dual solutions are included in a bilinear term. This is often the case when considering revenue from producing or transporting power, as revenue is given as production or power flow, lower-level primal variables, times price, lower-level dual variables. If a bilevel optimization problem with a bilinear term in the upper-level objective is reformulated as an MPEC, the bilinear term is again present in the objective of the MPEC. Such a non-linear objective is difficult to solve to global optimality. Chapter 4 further examines such problems in detail and presents a solution method to deal with these.

1.3.1 Generation Expansion Example

As an example of a bilevel optimization problem in a power market, we consider a generation expansion problem. A similar problem also serves as a case study in Chapter 4. The perspective is that of a power producer who wishes to invest in new capacity of one or more power generation units. We use (1.11) as a lower-level market clearing, but divide the generation units into subsets of existing competing units, $\mathcal{G}_E$, existing units that belong to the investor, $\mathcal{G}_I$, and candidate units that can be built by the investor, $\mathcal{G}_C$. With these subsets, we have $\mathcal{G} = \mathcal{G}_E \cup \mathcal{G}_I \cup \mathcal{G}_C$. The objective in the upper-level investment problem is to maximize profits from new and existing generation units. Revenues are offset by construction costs of the new generation units, given as linear costs with marginal cost c_g^I. The bilevel formulation is then

$$\min \quad \sum_{g\in\mathcal{G}_C} c_g^I x_g + \sum_{t\in\mathcal{T}} \sum_{g\in\mathcal{G}_I\cup\mathcal{G}_C} (c_g - \lambda_t^*) p_{gt}^* \tag{1.14a}$$

$$\text{s.t.} \quad x_g \geq 0 \quad \forall g \in \mathcal{G}_C \tag{1.14b}$$

$$p_{gt}^* \text{ are primal optimal solutions to (1.15)} \tag{1.14c}$$

$$\lambda_t^* \text{ are dual optimal solutions to (1.15)} \tag{1.14d}$$

where the lower level problem is formulated as

$$\min \quad \sum_{t \in \mathcal{T}} \sum_{g \in \mathcal{G}} c_g p_{gt} \tag{1.15a}$$

$$\text{s.t.} \quad \sum_{g \in \mathcal{G}} c_g p_{gt} = d_t \quad : \lambda_t \quad \forall t \in \mathcal{T} \tag{1.15b}$$

$$0 \leq p_{gt} \leq p_g^{max} \quad \forall g \in \mathcal{G}_E \cup \mathcal{G}_I, \forall t \in \mathcal{T} \tag{1.15c}$$

$$0 \leq p_{gt} \leq x_g \quad \forall g \in \mathcal{G}_C, \forall t \in \mathcal{T}. \tag{1.15d}$$

Note, that x_g acts as the maximum capacity for the newly installed units and is a parameter in (1.15). In the upper-level objective function, we have the bilinear revenue term, $\lambda_t^* p_{gt}^*$. Under some assumptions this term can be linearized using the KKT conditions and strong duality of the lower-level problem. These cases are specified in Chapter 4.

Often, the time resolution of the lower-level problem is hours, meaning that we solve the ED for all hours of the planning horizon of the investment problem. Simplifications can be made by clustering the hours into representative hours, as mentioned in Section 1.2.1. When clustering the hours, however, some inaccuracies can occur when choosing the representation period and which constraints to include in the ED. These trade-offs are discussed and analyzed through numerical studies in Chapter 2

To solve such a generation expansion problem, Chapter 4 introduces a method that allows for decomposition of the lower-level problem into subproblems for each hour. This method involves solving more but smaller problems using parametric programming and can have significant computational advantages.

Chapter 2

Efficiently solving Linear Bilevel Programming Problems using Off-the-shelf Optimization Software

S. PINEDA, H. BYLLING AND J.M. MORALES

Chapter Abstract

A lot of optimization models in engineering are formulated as bilevel problems. Bilevel optimization problems are mathematical programs where a subset of variables is constrained to be an optimal solution of another mathematical program. Due to the lack of optimization software that can directly handle and solve bilevel problems, most existing solution methods reformulate the bilevel problem as a mathematical program with complementarity conditions (MPCC) by replacing the lower-level problem with its necessary and sufficient optimality conditions. MPCCs are single-level non-convex optimization problems that do not satisfy the standard constraint qualifications and therefore, non-linear solvers may fail to provide even local optimal solutions. In this paper we propose a method that first solves iteratively a set of regularized MPCCs using an off-the-shelf non-linear solver to find a local optimal solution. Local optimal information is then used to reduce the computational burden of solving the Fortuny-Amat reformulation of the MPCC to global optimality using off-the-shelf mixed-integer solvers. This method is tested using a wide range of randomly generated examples. Obtained results show that our method outperforms existing general purpose methods in terms of computational burden and global optimality.

2.1 Introduction

Decentralized environments are characterized by multiple decisions makers with divergent objectives that interact with each other in a hierarchical organization. In the simplest case with only two decision makers, one player, called the *leader*, makes her decisions first and then the other player, called the *follower*, determines the optimal reaction to the leader's decisions. This non-cooperative sequential game is known as *Stackelberg game* and was first investigated in Stackelberg (1934). A Stackelberg game can be mathematically formulated as a bilevel problem (BLP) as follows Bard (1998); Dempe (2002):

$$\min_{x} \quad F(x,y) \tag{2.1a}$$

$$\text{s.t.} \quad G_i(x,y) \geq 0, \quad \forall i \tag{2.1b}$$

$$\min_{y} \quad f(x,y) \tag{2.1c}$$

$$\text{s.t.} \quad g_j(x,y) \geq 0, \quad \forall j \tag{2.1d}$$

where $F(x,y)$ and $f(x,y)$ are, respectively, the leader's and follower's objective functions, and $G_i(x,y)$ and $g_j(x,y)$ are the leader's and follower's constraint functions, respectively. Even if $F(x,y)$, $f(x,y)$, $G_i(x,y)$ and $g_j(x,y)$ are all linear functions, solving bilevel problem (2.1) is a very challenging task because its feasible region is non-convex in most of the cases. Furthermore, the BLP is proven to be NP-hard Jeroslow (1985); Bard (1991) and therefore the solution methods to solve BLP are computationally intensive. A review of the different solution approaches to solve the bilevel problem (2.1) can be found in Dempe (2003); Colson et al. (2005a, 2007).

From a practical point of view, methods to solve linear bilevel problems can be divided into two main categories. The first category includes those methods that make use of dedicated solution algorithms to solve bilevel problems Bialas and Karwan (1984); Shi et al. (2005a); Calvete et al. (2008); Li and Fang (2012); Sinha et al. (2013); Jiang et al. (2013); Bard and Falk (1982); Bard and Moore (1990); Hansen et al. (1992); Shi et al. (2006). While these methods are usually efficient and ensure global optimality, they involve substantial additional and ad-hoc coding work to be implemented in commercially available off-the-shelf optimization software such as CPLEX IBM Corp. (2015). The second category includes the methods that can be implemented in or in combination with general purpose optimization software without any further ado Fortuny-Amat and McCarl (1981); Ruiz and Conejo (2009); Gabriel and Leuthold (2010); Siddiqui and Gabriel (2013); Scholtes (2001); Ralph and Wright (2004); White and Anandalingam (1993); Hu and Ralph (2004); Lv et al. (2007); Fletcher and Leyffer (2004, 2002). Although these methods are sometimes preferred due to their straightforward implementation, they may involve a high computational burden or only guarantee local optimality. The method proposed in this paper belongs to this second group and is shown to outperform existing methods within its category in terms of computational efficiency and global optimality.

An important property of a linear bilevel problem (LBLP) with a bounded constraint region is that its solution set contains at least one extreme point of such a constraint region Bialas and Karwan (1984). Therefore, the first dedicated methods to solve LBLP were based on vertex enumeration. For instance, the Kth best method that computes global solutions of LBLP by enumerating the extreme points of the polyhedral constraint region is introduced in Bialas and Karwan (1984); Candler and Townsley (1982). Authors of Shi et al. (2005a) propose an extended Kth best approach when the upper-level constraint functions are of an arbitrary linear form. Although quite robust, the Kth best method is computationally costly, specially for large-size problems.

If the lower-level problem (2.1c)-(2.1d) is convex and satisfies some constraint qualification, problem (2.1) can be reformulated as a one-level optimization problem by replacing the lower-level problem with its KKT optimality conditions as follows Dempe and Zemkoho (2012); Dempe et al. (2015):

$$\min_{x,y,\lambda_j} \quad F(x,y) \tag{2.2a}$$

$$\text{s.t.}\quad G_i(x,y) \geq 0, \quad \forall i \tag{2.2b}$$

$$g_j(x,y) \geq 0, \quad \forall j \tag{2.2c}$$

$$\nabla_y f(x,y) - \sum_j \lambda_j \nabla_y g_j(x,y) = 0 \tag{2.2d}$$

$$\lambda_j \geq 0, \quad \forall j \tag{2.2e}$$

$$\lambda_j \cdot g_j(x,y) = 0, \quad \forall j \tag{2.2f}$$

where λ_j denotes the dual variable corresponding to each lower-level constraint (2.1d). Although (2.2) is the most commonly used approach, there exist alternative single-level reformulations of bilevel problems. Also under convexity assumptions, a bilevel problem (BLP) can be replaced by its primal KKT reformulation that does not need additional variables λ_j but requires determining the normal cone to the follower's feasible region for each value of x. Alternatively, problem (2.1) can be recast as a nonsmooth and nonconvex single-level optimization problem using an optimal value function of the lower-level problem. Further details about these two approaches can be found in Dempe et al. (2015).

Problem (2.2) is a mathematical program with complementarity conditions (MPCC) Outrata (2000). As proven in Dempe and Dutta (2010), if (x^*, y^*, λ_j^*) is a global optimal solution of problem (2.2), and the lower-level problem (2.1c)–(2.1d) is convex and satisfies some constraint qualification, then (x^*, y^*) is a global optimal solution of the original bilevel problem (2.1). Besides, if the lower-level problem is convex and Slater's condition holds, the local optimal solutions of problem (2.2) are also local optimal solutions of the bilevel problem (2.1) Dempe and Dutta (2010). Note that these conditions are always satisfied for the linear bilevel problems analyzed in this paper.

Note also that although constraint (2.2d) remains affine provided that f and g_j are linear or convex quadratic functions, problem (2.2) is non-convex due to the nonlinear complementarity conditions (2.2f). Besides, as shown in Scheel and Scholtes (2000), problem (2.2) violates the Mangasarian-Fromovitz constraint qualification at every feasible point of the problem, which makes both the formulation of (necessary and sufficient) optimality conditions and the computation of global optimal solutions difficult.

Taking the single-level optimization problem (2.2) as a starting point, we can also find methods within the two categories previously discussed. For example, some dedicated methods take advantage of the intrinsically combinatorial structure of problem (2.2) to handle the complementarity constraints using ad-hoc branch-and-bound algorithms as first proposed in Bard and Falk (1982) and further developed in Bard and Moore (1990); Hansen et al. (1992); Shi et al. (2006). In these methods, the root node solves the problem obtained by removing the complementarity conditions (2.2f). If at a given node one complementarity constraint j' is not satisfied, two new nodes are added to the tree, one with the additional constraint $\lambda_{j'} = 0$ and the other with the constraint $g_{j'}(x,y) = 0$. By repeating this process and solving the linear problems obtained after each branching, all possible combinations that satisfy the complementarity conditions are evaluated and therefore, obtaining the global optimal solution is guaranteed.

Alternatively, Fortuny-Amat and McCarl propose in Fortuny-Amat and McCarl (1981) a mixed-integer reformulation of problem (2.2) that can be directly implemented using off-the-shelf optimization software. This approach replaces the complementarity conditions (2.2f) with the following set of disjunctive constraints:

$$\lambda_j \leq z_j M, \quad \forall j \tag{2.3a}$$

$$g_j(x,y) \leq (1 - z_j) M, \quad \forall j \tag{2.3b}$$

where z_j is a binary variable and M a sufficiently large positive number. Note that for the linear case, problem (2.2) is reformulated as a mixed-integer linear programming problem that can be solved to optimality using conventional branch-and-bound or branch-and-cut techniques available in most mixed-integer optimization solvers. For this reason, this approach is the most commonly used to solve LBLP in practical applications. Notwithstanding this, the equivalence between problem (2.2) and its mixed-integer reformulation using (2.3) is only true provided that the value of M is large enough so that constraints (2.3a) and (2.3b) are only binding for $z_j = 0$ and $z_j = 1$, respectively. On the other hand, choosing a too large constant M may create numerical instabilities due to scalability issues. Hence, finding suitable values of M a priori is a delicate task. Although some ad-hoc methods have been proposed to solve this issue for particular applications of bilevel programming Ruiz and Conejo (2009); Gabriel and Leuthold (2010), tuning the large constants M for general LBLP requires a nontrivial trial-and-error process. In fact, in references Motto et al. (2005); Hasan et al. (2008); Garcés et al. (2009); Baringo and Conejo (2011); Wogrin et al. (2011); Pozo and Contreras (2011); Kazempour et al. (2011); Kazempour and Conejo (2012); Ruiz et al. (2012); Kazempour and Conejo (2012); Baringo and Conejo (2012b, 2013); Jenabi et al. (2013); Wogrin et al. (2013); Pozo et al. (2013); Zugno et al. (2013); Pisciella et al. (2014); Baringo and Conejo (2014); Lorenczik et al. (2014); Maurovich-Horvat et al. (2014); Morales et al. (2014); Ruiz and Conejo (2014); Valinejad and Barforoushi (2015); Moiseeva et al. (2015) the authors solve either MPEC or bilevel problems using the Fortuny-Amat reformulation approach, but without explaining how the large constants M are determined.

Another approach to solve (2.2) as a mixed-integer problem consists in reformulating the complementarity conditions using Special Order Sets (SOS) Siddiqui and Gabriel (2013). Special Order Sets of type 1 (SOS1) are a set of variables for which at most one member can be strictly positive. Therefore, constraint (2.2f) can be equivalently expressed as:

$$s_j(1) = \lambda_j, \quad \forall j \tag{2.4a}$$

$$s_j(2) = g_j(x, y), \quad \forall j \tag{2.4b}$$

where the pair $\{s_j(1), s_j(2)\}$ is defined as a SOS1 for each j. The main advantages of this approach are that no large constant is required and that it can be also directly solved using commercially available mixed-integer optimization solvers. On the other hand, this method can also be computationally very expensive, especially for large models, as proven in Section 2.5.

As previously mentioned, optimization problem (2.2) is not regular since it fails to comply with the standard Mangasarian-Fromovitz constraint qualification and therefore, off-the-shelf non-linear solvers may even fail to find a local optimal solution. For instance, if the non-linear solver is based on a sequential quadratic programming algorithm (SQP), the quadratic programming subproblems may be degenerated because the original problem (2.2) has no strictly feasible points Fletcher and Leyffer (2004). To overcome this issue, a regularization approach to solve mathematical programs with complementarity conditions (MPCC) was first introduced in Scholtes (2001) and further investigated in Ralph and Wright (2004). This method proposes to replace each complementarity constraint (2.2f) by:

$$\lambda_j \cdot g_j(x, y) \leq t, \quad \forall j \tag{2.5}$$

where t is a small non-negative scalar. In doing so, problem (2.2) becomes a parametrized nonlinear optimization problem that typically satisfies constraint qualifications and is thus easier to solve. Alternatively, all inequalities in (2.5) can be replaced by a single inequality as follows:

$$\sum_j \lambda_j \cdot g_j(x, y) \leq t \tag{2.6}$$

Using (2.6) instead of (2.5) may improve the numerical behaviour of nonlinear solvers since the number of inequality constraints is reduced. In either case, reference Scholtes (2001) provides the necessary conditions under which a local minimizer of the original problem (2.2) is a limit point of a curve of stationary points of the parametrized nonlinear problem as t tends to 0. Although this regularization method significantly reduces the computational burden of solving problem (2.2), using existing nonlinear optimization techniques such as SQP only guarantees local optimal solutions of problem (2.2), which are not necesarily local optimal solutions of the generic bilevel problem (2.1) Dempe and Dutta (2010). Another advantage of this method is that it can also be directly implemented using off-the-shelf non-linear optimization software since it just consists on iteratively solving a set of non-linear problems.

Some other works investigate the solution of linear bilevel problems using a penalty function. For example, the procedure proposed in White and Anandalingam (1993) disregards the complementarity conditions (2.2f) and adds a term to the upper-level objective function that penalizes the duality gap of the lower-level optimization problem. In the linear case, the authors of White and Anandalingam (1993) demonstrate that the proposed procedure guarantees global optimality. Further studies about penalty methods for solving LBLP can be found in Hu and Ralph (2004); Lv et al. (2007).

Finally, some heuristic methods have been suggested in the literature to solve linear bilevel problems. For example, the procedure proposed in Hejazi et al. (2002) applies genetic algorithms to solve the KKT reformulation of the LBLP. Similarly, authors of Calvete et al. (2008) present a solution algorithm that combines extreme point enumeration techniques with genetic search methods. References Li and Fang (2012); Sinha et al. (2013) introduce evolutionary algorithms to solve bilevel problems. The approach proposed in Jiang et al. (2013) applies particle swarm optimization to a smooth version of the KKT reformulation of the bilevel problem. Given the complexity of these approaches and the amount of extra code required to be implemented in standard optimization software, they fall into the category of dedicated methods.

In summary, dedicated methods such as Kth best method, ad-hoc branch-and-cut algorithms, or heuristic approaches, can be efficient to provide the global optimal solutions of linear bilevel problems. However, they cannot be directly coded using off-the-shelf optimization software. Among general purpose methods that can be directly implemented using optimization solvers, the mixed-integer reformulations (Fortuny-Amat or SOS1 approaches) determine global optimal solutions at the expense of drastically increasing the computational burden. On the other hand, regularization approaches to solve the KKT reformulation of the LBLP using off-the-shelf non-linear optimization software prove to be fast but cannot guarantee neither global nor local optimality of the original bilevel problem Dempe and Dutta (2010). In this paper we propose a new procedure that combines these two approaches to efficiently solve linear bilevel programming problems and that can be directly implemented using off-the-shelf optimization software. The contribution of this paper is thus twofold:

- We provide a computationally efficient method to solve linear bilevel programming problems using available optimization software. The proposed method uses first a regularization approach to efficiently determine a local optimal solution of the

KKT reformulation of the LBLP using a non-linear optimization solver. Then, this local optimal solution is used to significantly reduce the computational burden of solving the mixed-integer linear reformulation proposed in Fortuny-Amat and McCarl (1981) using a conventional mixed-integer optimization solver as follows. First, by setting appropriate values of the large constant M in (2.3) according to the order of magnitude of the primal and dual variables. Second, by providing initial values to the binary variables based on which term of the complementarity conditions is equal to 0 at the local optimal solution.

- We test the performance of the proposed method through a set of comprehensive computational studies based on a large family of randomly generated examples of different sizes. The proposed method is compared in terms of computational burden and global optimality against other general purpose methods to solve LBLP. The obtained results show that the proposed approach is an efficient generic algorithm to solve lineal bilevel problems in practice.

The remaining of this paper is organized as follows. Section 2.2 formally presents the generic formulation of the linear bilevel problem under study together with some important definitions and properties. Section 2.3 introduces the KKT reformulation of the LBLP and explains in detail how both existing and the proposed algorithm can be used to solve it. Section 2.4 elaborates on how test examples are randomly generated and sets the basis for comparing the results provided by the different methods. The main computational results are presented and discussed in Section 2.5. Finally, Section 2.6 concludes the paper.

2.2 Linear bilevel programming problem

Given the complexity of bilevel programming problems, in this paper we restrict ourselves to the simplest case in which functions $F(x, y)$, $f(x, y)$, $G_i(x, y)$ and $g_j(x, y)$ are all linear. Hence, a linear bilevel problem (LBLP) is generally formulated as follows Bard (1998); Zhang et al. (2015):

$$\min_{x} \quad F(x, y) = c_1 x + d_1 y \tag{2.7a}$$

$$\text{s.t.} \quad A_1 x + B_1 y \leq b_1 \tag{2.7b}$$

$$\min_{y} \quad f(x, y) = c_2 x + d_2 y \tag{2.7c}$$

$$\text{s.t.} \quad A_2 x + B_2 y \leq b_2 \tag{2.7d}$$

where $c_1, c_2, d_1, d_2, b_1, b_2, A_1, B_1, A_2, B_2$ are vectors and matrices of appropriate dimensions. The induced region (IR) of the LBLP is the set of feasible points of the leader and rational responses from the follower Bard (1998). With this notation, the LBLP can be equivalently recast as the following one-level optimization problem:

$$\min_{x,y} \quad F(x, y) \tag{2.8a}$$

$$\text{s.t.} \quad (x, y) \in IR \tag{2.8b}$$

If an explicit formulation of the IR as a polyhedron were possible and available, the solution to (2.7) could be obtained by solving problem (2.8) as a one-level linear programming problem using, for example, the simplex method. However, even for simple instances of LBLP, the IR cannot be formulated as a polyhedron, which makes (2.8) a very hard problem to solve Jeroslow (1985); Bard (1991); Ben-Ayed and Blair (1990). As proven in Bard (1998), if follower's rational reaction set is bounded and the constraint region is non-empty and bounded, then an optimal solution to the LBLP (2.8) exists. Therefore, unless otherwise specified, these assumptions apply to all problems presented in this paper.

One issue worth discussing is the existence of upper-level constraints that include both upper-level and lower-level variables. The validity of such joint upper-level constraints is beyond the choice of the leader and can only be validated after the follower's optimal choice is determined Dempe et al. (2015). Mathematically, joint upper-level constraints can lead to disconnected or empty IR Colson et al. (2005a), which may further complicates the solution of the linear bilevel problem as illustrated in Shi et al. (2005c). Extended approaches to apply existing solution algorithms to LBLP with upper-level constraints of arbitrary form can be found in Shi et al. (2006, 2005b); Mersha and Dempe (2006). However, for the sake of simplicity, this paper only considers LBLP with upper-level constraints that do not include lower-level variables, i.e., $B_1 = 0$ in (2.7) unless otherwise stated.

Another important aspect of LBLP is the existence of multiple optimal solutions to the lower-level problem. Under such circumstances, the leader's choice has to be determined without exactly knowing the reaction of the follower, who can choose among a set of decisions that lead to the same value of her objective function. To overcome this indeterminacy, there are two main possibilities, namely, the optimistic and the pessimistic solution Dempe (2002); Colson et al. (2005a, 2007). The leader can assume that the follower can be influenced to select the solution that involves a higher leader's objective function. This is known as the optimistic solution of a LBLP. Conversely, the pessimistic solution considers that the leader has no possibility to alter the behavior of the follower, who can choose the worst solution with respect to the leader's objective function. In this paper we focus on the optimistic formulation since it is simpler, is the usual approach and has been more deeply investigated in the technical literature Dempe et al. (2007); Strekalovsky et al. (2010b); Dempe and Franke (2014). For further details about the pessimistic formulation of a linear bilevel problem, the interested reader is referred to Dempe et al. (2014) and references therein.

2.3 Solution methods

The original linear bilevel problem (2.7a)-(2.7d) can be reformulated as the single-level optimization problem (2.9a)-(2.9f) by replacing its lower-level optimization problem with its KKT optimality conditions. Note that model (2.9a)-(2.9f) is a non-linear optimization problem because of the products $\lambda \cdot x$ and $\lambda \cdot y$ in equation (2.9f), where λ denotes a vector with the dual variables of the lower-level constraint (2.7d). All the methods presented in this Section aim at solving this single-level non-linear optimization model using different approaches. The following subsections provide the detailed steps of the solution algorithms compared in this paper.

$$\min_{x,y,\lambda} \quad F(x,y) = c_1 x + d_1 y \tag{2.9a}$$

$$\text{s.t.} \quad A_1 x + B_1 y \leq b_1 \tag{2.9b}$$

$$d_2 + \lambda B_2 = 0 \tag{2.9c}$$
$$b_2 - A_2x - B_2y \geq 0 \tag{2.9d}$$
$$\lambda \geq 0 \tag{2.9e}$$
$$\lambda\,(b_2 - A_2x - B_2y) = 0 \tag{2.9f}$$

2.3.1 Branch-and-bound approach

This method solves the single-level reformulation of the LBLP (2.9) using a binary tree. The method starts by solving the relaxed linear problem (2.9a)-(2.9e). If all complementarity conditions are satisfied, then this is the optimal solution to (2.9). Otherwise, the tree is branched in one of the violated complementarity constrains j' so that two nodes are added to the tree. A linear optimization problem is defined for each new node by adding the constraint $\lambda_{j'} = 0$ or $(A_2x + B_2y - b_2)_{j'} = 0$ to the problem corresponding to the predecessor node. This procedure continues until the subproblems corresponding to all ending nodes are infeasible or have an objective value larger than the current upper bound Bard and Moore (1990).

Note that this approach only involves the solution of linear programming problems and therefore, convergence to global optimality is guaranteed. For this reason, and despite the fact that this approach belongs to the category of dedicated solution methods, the solution provided by the branch-and-bound is used to check the performance of the other general purpose methods investigated in this paper. On the other hand, applying this algorithm to solve LBLP may easily become computationally expensive, even for low size problems.

2.3.2 Mixed-integer approach

Given the combinatorial nature of the complementarity constraints (2.9f), some solution methods propose to reformulate problem (2.9) as a mixed-integer programming problem and directly use off-the-shelf integer optimization software. The idea of Fortuny-Amat is to rewrite these complementarity conditions using disjunctive constraints that require the use of binary variables and large enough constants Fortuny-Amat and McCarl (1981). Problem (2.9) is thus reformulated as follows:

$$\min_{x,y,\lambda,u} \quad F(x,y) = c_1x + d_1y \tag{2.10a}$$
$$\text{s.t.} \quad A_1x + B_1y \leq b_1 \tag{2.10b}$$
$$d_2 + \lambda B_2 = 0 \tag{2.10c}$$
$$b_2 - A_2x - B_2y \geq 0 \tag{2.10d}$$
$$\lambda \geq 0 \tag{2.10e}$$
$$b_2 - A_2x - B_2y \leq (1-u)M_1 \tag{2.10f}$$
$$\lambda \leq uM_2 \tag{2.10g}$$
$$u \in \{0,1\} \tag{2.10h}$$

where u is a vector of binary variables of appropriate size and M_1, M_2 are large enough scalars. Note that formulation (2.10) is obtained from formulation (2.9) by simply replacing the non-linear constraint (2.9f) with constraints (2.10f), (2.10g) and (2.10h). Problem

(2.10) is a mixed-integer linear programming problem that can be solved using conventional branch-and-bound algorithms as the one used by CPLEX IBM Corp. (2015).

Alternatively, SOS1 variables can be used to impose the complementarity conditions by replacing equations (2.10f)-(2.10h) with Siddiqui and Gabriel (2013):

$$s_j(1) = (b_2 - A_2x - B_2y)_j \,, \quad \forall j \tag{2.11a}$$

$$s_j(2) = \lambda_j, \quad \forall j \tag{2.11b}$$

where the pair $\{s_j(1), s_j(2)\}$ is declared as SOS1 for each j. Problem (2.11) can also be solved using mixed-integer linear solution methods as those in commercially available optimization software.

If the values of M_1, M_2 are properly set, both (2.10) and (2.11) can be solved to global optimality using existing mixed-integer optimization solvers. However, similarly to the branch-and-bound approach, the computational burden of solving these models dramatically increases with the size of the bilevel problem.

2.3.3 Regularization approach

As shown in Scheel and Scholtes (2000), all feasible points of (2.9) are nonregular, which implies that most existing nonlinear optimization solvers may fail even to find a local optimal solution. If the regularization approach proposed in Scholtes (2001); Ralph and Wright (2004) is applied to problem (2.9), we obtain the following formulation:

$$\min_{x,y,\lambda} \quad F(x,y) = c_1x + d_1y \tag{2.12a}$$

$$\text{s.t.} \quad A_1x + B_1y \leq b_1 \tag{2.12b}$$

$$d_2 + \lambda B_2 = 0 \tag{2.12c}$$

$$b_2 - A_2x - B_2y \geq 0 \tag{2.12d}$$

$$\lambda \geq 0 \tag{2.12e}$$

$$\lambda\left(b_2 - A_2x - B_2y\right) \leq t \tag{2.12f}$$

where t is a non-negative small scalar. Formulation (2.12) is derived from formulation (2.9) by replacing the non-linear equality constraint (2.9f) with the non-linear inequality constraint (2.12f). Notice that both models are, therefore, equivalent for t tending to 0. This approach consists in iteratively solving a set of non-linear regular optimization problems. In each iteration, the value of t is reduced. The local optimal solution in one iteration is used as the initial starting point for the following iteration. While being relatively fast and presenting strong theoretical and empirical convergence properties Scholtes (2001), this regularization approach is only guaranteed to provide local optimal solutions of the MPCC, which are also local optimal solutions of the original LBLP Dempe and Dutta (2010).

2.3.4 Penalty approach

Another method to solve the nonregular problem (2.9) consists in penalizing the complementarity constraints in the objective function as follows White and Anandalingam (1993); Hu and Ralph (2004); Lv et al. (2007):

$$\min_{x,y,\lambda} \quad F(x,y) = c_1x + d_1y + \frac{1}{t}\sum_j \lambda_j\left(b_2 - A_2x - B_2y\right)_j \tag{2.13a}$$

$$\text{s.t.} \quad A_1x + B_1y \leq b_1 \tag{2.13b}$$
$$d_2 + \lambda B_2 = 0 \tag{2.13c}$$
$$b_2 - A_2x - B_2y \geq 0 \tag{2.13d}$$
$$\lambda \geq 0 \tag{2.13e}$$

where t is also a non-negative scalar that is iteratively decreased to make the complementarity conditions tend to 0. The initial value of t is set to a large value and is reduced by a factor of $\rho > 1$ in each iteration. As in the regularization method, a non-linear optimization problem has to be solved at each iteration.

2.3.5 Proposed approach

The purpose of the proposed solution method is to combine the mixed-integer and the regularization approaches presented above in order to obtain a global optimal solution while reducing the computational burden. The main issue with the regularization approach is that, albeit fast, it only ensures local optimal solutions for the MPCC reformulation. On the other hand, formulation (2.10) can be solved to global optimality. However, finding appropriate values of the large constants M_1, M_2 that allow solving (2.10) in a reasonable time is usually a difficult task. In fact, very low or very high values of M_1, M_2 may lead to infeasible, suboptimal and numerically unstable problems, respectively. The proposed approach uses the local optimal solution for the MPCC reformulation provided by the regularization method to soundly determine values of these large constants that allow us to find the optimal global solution of (2.10) at a low computational cost.

The proposed approach relies on non-linear optimization solvers whose performance is significantly improved if a feasible initial point is provided. This initial feasible point is calculated by sequentially solving two linear programming problems. The first linear optimization problem is obtained by removing the non-linear complementarity condition from model (2.9) to obtain a pair (x, y) that satisfies all upper- and lower-level constraints, but that is not optimal for the lower-level problem. We then fix the values of x and solve the lower-level optimization problem alone, which is also a linear programming problem, to find values of y that are also optimal for the lower-level problem. Therefore, by sequentially solving these two linear programming problems, we obtain a feasible point (x, y) that satisfies all the constraints (2.9b)-(2.9f).

The proposed approach requires the use of the following parameters:

k Iteration counter.

t Non-negative small scalar representing the slackness of the complementarity conditions.

ρ Non-negative scalar used to update the value t.

$\mathcal{M}$ Non-negative scaling parameter used to compute the large enough constants.

The steps of the proposed procedure are the following:

- Step 0 (Initialization) Select parameters $t > 0$, $\rho > 1$, $\mathcal{M} > 1$ and the number of iterations K. Set $k \leftarrow 0$ and go to Step 1.

- Step 1 (Feasible point) Solve the linear programming problem (2.9a)–(2.9e) and denote the obtained leader's variables as x_0. Solve the lower-level linear programming problem (2.7c)–(2.7d) in which upper-level variables are fixed at x_0. Denote the optimal values of the primal and dual variables as y_0 and λ_0, respectively. Go to Step 2.

- Step 2 (Iteration) Set $k \leftarrow k + 1$. Solve problem (2.12) taking $(x_{k-1}, y_{k-1}, \lambda_{k-1})$ as an initial point. Denote its solution as (x_k, y_k, λ_k). If $k < K$, then $t \leftarrow t/\rho$ and go to Step 2. Otherwise, go to Step 3.

- Step 3 (Tuning) Set $M_1 \leftarrow \mathcal{M} \max_j\{(b_2 - A_2x_k - B_2y_k)_j\}$ and $M_2 \leftarrow \mathcal{M} \max_j\{(\lambda_k)_j\}$. Go to Step 4.

- Step 4 (Warming) Set initial values of binary variables u as follows. If $(b_2 - A_2x_k - B_2y_k)_j > 0$, then $u_j = 0$. If $\lambda_j > 0$, then $u_j = 1$. Go to Step 5.

- Step 5 (Solution) Solve the mixed-integer linear problem (2.10) using the values of M_1, M_2 determined in Step 3 and the initial values of the binary variables computed in Step 4. Declare its solution (x^*, y^*, λ^*) as the optimal solution.

The core of the proposed approach relies on Steps 3 and 4, in which the local optimal solution provided by the regularization method is used to tune the large constants M_1 and M_2 and to compute initial values for the binary variables u, respectively. Let us explain first the reasoning behind Step 3. Note that the mixed-integer approach (2.10) is only valid provided that constraints (2.10f) and (2.10g) are binding if and only if $u = 1$ or $u = 0$, respectively. This is only true if the following two conditions hold: M_1 is larger than $b_2 - A_2x - B_2y$ for any feasible pair (x, y) and M_2 is larger than any feasible value of the dual variable λ. Even though the solution obtained in Step 2 using regularization is just locally optimal, we assume that the maximum value of $b_2 - A_2x - B_2y$ over all lower level constraints at the local optimal solution is a good proxy of M_1. Similarly, the maximum value of the lower-level dual variable λ_j over all constraints at the local optimal solution is also a good estimation of the large constant M_2. If large constants M_1 and M_2 are tuned based exclusively on the locally optimal solution computed in Step 2, two issues may arise. In some cases, the globally optimal solution to the original linear bilevel problem may be actually infeasible due to the bad adjustment of the large constants M_1 and M_2. For other cases, the optimal solution (2.10) may not be globally optimal for the original optimization problem due to the overly-constrained feasible region. To avoid these two issues, these values are multiplied by the scaling parameter $\mathcal{M} > 1$, which needs to be adjusted by trial and error bearing in mind the following trade-off: the larger the value of $\mathcal{M}$, the lower the risk that the global optimal solution becomes infeasible or suboptimal, but the higher the computational time required to solve the problem due to numerical instabilities. The intuition behind Step 4 is the following. Note that the values of variables u obtained in Step 2 provide information about which term of each complementarity condition (2.9f) is equal to 0 at the locally optimal solution. Assuming that the globally optimal solution is not "too different" from the locally optimal solution obtained by the regularization approach, the terms of the complementarity conditions equal to 0 are expected to coincide for most of these constraints.

Providing initial values for the binary variables u and tuning the large constants M_1, M_2 only seeks to improve the computational performance of the mixed-integer solver without jeopardizing the optimality of the solution that the solver eventually returns. How much the computational burden of solving (2.10) will be reduced by taking advantage of the locally optimal information provided by (2.12) cannot be exactly established a priori with full guarantees. To provide some guidance on this issue, however, we conduct and present an exhaustive numerical analysis in Sections 2.5.1, 2.5.2 and 2.5.3, in which a large set of linear bilevel problems of different size, sparsity and scale are solved.

Finally, note that the proposed solution algorithm can be directly implemented using off-the-shelf optimization software since it only involves solving:

- Two linear programming problems using a linear optimization solver to find a point in the induced region.
- A family of regularized non-linear optimization problems using a non-linear optimization solver to find a local optimal solution.
- A mixed-integer linear programming problem with appropriate large constants and initial values of the binary variables using a mixed-integer optimization solver to find the global optimal solution.

2.4 Test and comparison

In this section, we first describe how test bilevel problems are randomly generated and then explain how the results provided by the different solutions methods are compared.

As previously discussed, the test examples considered in this paper do not include any joint upper-level constraints and therefore, matrix B_1 is empty. In order to avoid unbounded test problems, it is also imposed that both the coefficients of the upper-level and lower-level objective functions (c_1, d_1, c_2, d_2) and the variables involved (x, y) must be non-negative. For the sake of generality, test bilevel problems include two sets of lower-level constraints: the first set of constraints involves upper- and lower-level variables, while the second only comprises lower-level variables. According to these assumptions, vectors and matrices of bilevel problem (2.7) are generated as follows:

$$c_1 = |\mathcal{N}(1,n)| \quad d_1 = |\mathcal{N}(1,m)| \quad A_1 = \begin{pmatrix} \mathcal{N}(p,n) \\ -I \end{pmatrix} \quad B_1 = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \quad b_1 = \begin{pmatrix} \mathcal{N}(p,1) \\ \mathbf{0} \end{pmatrix}$$

$$c_2 = |\mathcal{N}(1,n)| \; d_2 = |\mathcal{N}(1,m)| \; A_2 = \begin{pmatrix} \mathcal{N}(q,n) \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \; B_2 = \begin{pmatrix} \mathcal{N}(q,m) \\ \mathcal{N}(r,m) \\ -I \end{pmatrix} \; b_2 = \begin{pmatrix} \mathcal{N}(q,1) \\ \mathcal{N}(r,1) \\ \mathbf{0} \end{pmatrix}$$

where $\mathcal{N}(i,j)$ denotes a $i \times j$ matrix in which each element is randomly generated according to a standard normal distribution with mean and variance equal to 0 and 1, respectively. As follows from these definitions, n and m are the number of upper- and lower-level variables, respectively. Furthermore, each random problem includes p upper-level constraints, q lower-level joint constraints and r lower-level constraints not involving upper-level variables.

Given one random problem, let l be an index for the different solution approaches presented in this paper. The optimal solution, objective function value and solver status provided by solution approach l are denoted as (x_l^*, y_l^*), z_l^* and s_l, respectively, and are computed as follows:

- Step 1) The bilevel problem is solved using solution method l and the optimal upper-level variables are denoted as x_l^*. If no solution is provided, set s_l to 0 and stop. Otherwise, go to Step 2).

- Step 2) The upper-level variables are fixed to x_l^* and the lower-level problem is solved again using linear programming to obtain the lower-level optimal variables y_l^*. If the lower-level is infeasible, set s_l to 0 and stop. Otherwise, go to Step 3).

- Step 3) Set s_l to 1 and compute the value of the objective function z_l^* as $c_1 x_l^* + d_1 y_l^*$.

This procedure to compare the different methods is particularly relevant for those formulations that include products of binary variables and large numbers. Note that some mixed-integer solvers may round down this product and thus yield optimal values for the binary variables different from 0 and 1 due to numerical instabilities. If this happens, the objective function obtained by these methods may be lower than the optimal one since complementarity conditions do not hold. However, if we fix the upper-level variables and then solve the lower-level problem as described above, this issue is avoided and the values of the upper-level objective function provided by different solution methods can be fairly compared. For each random problem, the true optimal solution $\hat{z}$ is defined as:

$$\hat{z} = \min\{z_l^* : s_l = 1\}$$

In most examples, $\hat{z}$ will be equal to the solution provided by the branch-and-bound and SOS1 methods, since these approaches guarantee global optimality. If these methods do not provide a solution due to time restrictions, then $\hat{z}$ will be the minimum objective function among the methods that deliver a solution. The optimality gap for the solution given by method l is thus computed as:

$$\mathrm{g}_l = 100 \times \frac{z_l^* - \hat{z}}{\hat{z}}$$

which is only defined for those methods with $s_l = 1$.

In this paper we compare the following methods to solve linear bilevel problems:

- Branch and bound method (B&B).

- Mixed-integer solution method with SOS1 variables (SOS1).

- Mixed-integer solution method in which disjunctive constraints are modeled as proposed by Fortuny-Amat in Fortuny-Amat and McCarl (1981). The following 11 values for the large constants are used: 5, 10, 20, 50, 100, 200, 500, 1000, 5000, 10000, 100000. Each variant of this method is thus referred to as FA-5, FA-10, FA-20, etc.

- Regularization method proposed in Scholtes (2001) and Ralph and Wright (2004) (REG). The number of iterations (K) is set to 20, the initial value of t to 10^4, and ρ is equal to 10.

- Penalty approach proposed in White and Anandalingam (1993) (PEN). The number of iterations (K) is set to 20, the initial value of t to 1, and ρ is equal to 1.2.

- The proposed solution method, which is referred to as REG-FA. The regularized local optimization method is tuned as in REG. The following 3 values for the parameter $\mathcal{M}$ are used: 2, 5, 10. Each variant of this method is thus referred to as REG-FA-2, REG-FA-5 and REG-FA-10, respectively.

Table 2.1: Parameters of randomly generated problems

	n	m	p	q	r
Small size	50	50	25	25	25
Medium size	100	100	50	50	50
Large size	200	200	100	100	100

2.5 Computational results

This section compiles the main computational results of the methods presented in Section 2.3 to solve linear bilevel problems. First, the results of 300 test problems of different sizes are provided. Then, the impact of matrix sparsity on the performance of the different methods is investigated. Finally, we also analyze how bad scaling affects the obtained results.

All results here presented have been obtained using CPLEX 12.6.0.1 and CONOPT 3.16C optimization solvers under GAMS 24.3.3. The simulations have been run in a cluster with 288 nodes. Each node consists of Two Intel Xeon Processor E5649 (2.53 GHz, 6 cores) and 24GB of memory. The maximum time for each problem is set to 6 hours. The code and data are available in www.github.com/salvapineda/bilevel.

2.5.1 Impact of size

The solution methods presented in this paper are tested on 100 randomly generated problems of small size, 100 randomly generated of medium size, and 100 randomly generated problems of large size. The matrices of these problems are generated according to the parameters provided in Table 2.1. Note that the number of upper- and lower-level variables is the same in all cases. Furthermore, the number of each type of constraint is equal to half of the number of variables since a much higher or a much lower number of constraints may lead to infeasible or trivial problems, respectively. It is also worth mentioning that other works providing similar computational results consider randomly test cases with a maximum size of 150 upper- and lower-level variables Strekalovsky et al. (2010b,a).

Table 2.2 provides the results for the 18 methods compared in this study for the three problem sizes. For each problem size and solution approach four numerical results are provided, namely:

- The number of randomly generated problems solved to global optimality, that is, with zero optimality gap ($g_l = 0$). This is denoted as #opt.
- The number of randomly generated problems that are infeasible, that is, with $s_l = 0$. This is denoted as #inf.
- The average computational time (in seconds) for those randomly generated problems with valid solutions, that is, with $s_l = 1$.
- The average optimality gap (in percentage) for those randomly generated problems with valid solutions, that is, with $s_l = 1$.

Therefore, 100 - #opt - #inf is the number of non-optimal valid solutions.

Let us first analyze the results provided by the SOS1 method. Note that for small size examples, this method achieves the optimal solution in 98 of the 100 cases in around 1

Table 2.2: Results: Impact of size

	Small (n=50)				Medium (n=100)				Large (n=200)			
	#opt	#inf	time(s)	gap(%)	#opt	#inf	time(s)	gap(%)	#opt	#inf	time(s)	gap(%)
B&B	92	2	2761	0.66	44	0	13556	2.89	8	0	20211	2.54
SOS1	98	2	1	0.00	90	0	4656	0.27	27	0	17652	2.05
FA-5	11	24	12	7.65	8	7	5385	7.15	0	20	21415	4.57
FA-10	72	4	3	0.91	68	3	2798	0.18	44	0	16976	0.27
FA-20	95	4	3	0.01	92	1	3914	0.05	51	0	16981	0.32
FA-50	98	2	5	0.00	94	2	5495	0.04	53	1	16873	0.31
FA-100	97	2	9	0.00	93	0	6678	0.02	45	1	18453	0.60
FA-200	93	2	15	0.09	81	0	7716	0.19	37	0	18634	0.68
FA-500	85	2	36	0.14	78	2	9229	0.15	43	0	19251	0.38
FA-1000	79	2	55	0.29	66	1	10847	0.33	28	0	19241	0.51
FA-5000	49	2	80	2.28	26	4	7578	3.54	2	0	15517	3.63
FA-10000	37	2	36	3.95	13	0	4361	7.71	2	0	6782	10.34
FA-100000	27	2	0	6.44	11	0	0	10.19	2	0	1	9.86
REG	61	2	0	0.55	41	0	1	0.52	30	1	14	0.29
PEN	34	2	0	0.85	10	0	17	1.03	12	0	30	0.78
REG-FA-2	94	2	2	0.02	96	0	660	0.01	82	2	10657	0.07
REG-FA-5	98	2	2	0.00	99	1	767	0.00	71	11	10770	0.08
REG-FA-10	98	2	5	0.00	99	0	2353	0.00	72	0	13258	0.10

second, the remaining 2 cases being infeasible. For the medium size, 90 of the instances are solved to optimality while the average solution time is increased up to 1.3 hours. The average gap of 0.27% is due to the fact that some problems were not solved to optimality after six hours. Finally, for large-sized problems, the SOS1 method only achieved the optimal solution in 27 cases and the average solution time amounts to 4.9 hours. The increase in the computational time required by this method with the size of the problem is thus apparent. As the SOS1 method, the branch-and-bound method guarantees global optimality. Note, however, that the number of optimal solutions, the average computational time and the average optimality gap are worse for the branch-and-bound method for all problem sizes. Therefore, the SOS1 method is considered in this analysis as a benchmark.

Regarding the Fortuny-Amat method, the following general observations are in order. Both for very low and very large values of the large constant M, the number of examples solved to optimality is very low although for different reasons. While small values of M lead to a high number of infeasible problems, high values of M create numerical instabilities in the solution algorithm. Note also that the value of M that results in the largest number of test problems solved to optimality is equal to 50 for the three sets of examples, being the average time equal to 5 seconds, 90 minutes and 4.5 hours for small, medium and large problems, respectively. Observe that for the large size case, the maximum number of optimal solutions achieved by the best Fortuny-Amat method is only 53.

Despite being very fast, the regularization method only provides the global optimal solution in a low number of cases, which decreases as the dimension of the problems increases. Note that for large problems, in only 30 examples the local optimal solution found by this method is also global optimal. Observe as well that the results provided by the penalty method are even worse than those of the regularization method in terms of global optimality, computational time and optimality gap.

For the three problem sizes, the proposed approach provides very similar results for the three values of $\mathcal{M}$ in terms of number of optimal cases, computational time, and optimality gap. This shows that selecting an appropriate value of $\mathcal{M}$ for the proposed approach is substantially less critical than choosing a high enough value of M for the Fortuny-Amat approach. Let us then focus on the results for $\mathcal{M} = 10$, for example. For small size problems, REG-FA-10 also results in 98 instances solved to optimality, but with an average time higher than that of the SOS1 method. Given the low number of binary variables, optimization solvers such as CPLEX are quite efficient in solving problems of this size and that implies that the pre-calculations of the proposed method significantly increase the computational time in comparative terms. On the other hand, for medium-sized problems, REG-FA-10 is able to find the optimal solution in 99 cases in an average time of 40 minutes, thus outperforming the SOS1 method (90 optimal cases, 1.3 hours) and the best Fortuny-Amat method (94 optimal cases, 1.5 hours). These results demonstrate, therefore, the computational efficiency of the solution method proposed in this paper. For large problems, REG-FA-10 obtains 72 optimal cases in 3.6 hours, versus the 27 optimal cases and 4.9 hours of the SOS1 method, and the 53 optimal cases and 4.6 hours of the best Fortuny-Amat method. Notice also that the average gap corresponding to the non-optimal cases is equal to 0.10%, 2.05% and 0.31% for REG-FA-10, SOS1 and FA-50, respectively.

It should be noted that the discussion above is based on comparing the proposed approach with the Fortuny-Amat method providing the best results. However, the value of M that performs best is not known in advance and can only be determined after a trial-error process similar to the extensive testing done in this paper, which makes our method even more advantageous than what this analysis already reveals.

2.5.2 Impact of sparsity

All the randomly generated matrices for the analysis of the previous subsection are full matrices. In order to investigate the performance of the proposed solution algorithm for more sparse bilevel problems, three additional sets of 100 randomly generated problems are solved using the different methods in this section. For this study, half of the elements of each vector and matrix are randomly set to 0. The rest of parameters to generate the random problems are equal to those provided in Table 2.1. Table 2.3 contains the results corresponding to the bilevel problems with 50% sparsity.

As in Table 2.2, we can observe that although the SOS1 method outperforms the B&B method for all problem sizes, this method provides a number of optimal cases and an average computational time that drastically worsen as the problem dimension increases. It is also shown that the results of the Fortuny-Amat method highly depend on the value of M, being the best value around 50. Again, the results provided by the proposed method are not very sensitive to the value of $\mathcal{M}$ and hence, we focus on those of REG-FA-10 to make the following comparison analysis. For small problems, the results of the proposed method are similar to those of the SOS1 and the best Fortuny-Amat. For medium-sized problems, the proposed method achieves 97 optimal cases in 27 minutes, versus the 86 optimal cases in 72 minutes of the SOS1 method and the 92 optimal cases in 71 minutes of the best Fortuny-Amat. Finally, for large problems, our method provides 61 optimal cases in 3.5 hours, versus the 29 optimal cases in 4.5 hours of the SOS1 approach and the 48 optimal cases and 5 hours of the best-tuned Fortuny-Amat method. Note also that our

Table 2.3: Results: Impact of sparcity

	Small (n=50)				Medium (n=100)				Large (n=200)			
	#opt	#inf	time(s)	gap(%)	#opt	#inf	time(s)	gap(%)	#opt	#inf	time(s)	gap(%)
B&B	81	3	5100	1.81	42	0	14019	3.19	9	0	19964	2.53
SOS1	97	3	2	0.00	86	0	4293	0.48	29	0	16061	2.19
FA-5	13	24	21	7.14	7	12	4370	8.75	0	11	21366	5.50
FA-10	67	11	5	1.12	67	5	2123	0.62	34	9	15709	0.35
FA-20	90	6	8	0.01	88	3	3109	0.12	44	10	15803	0.21
FA-50	97	3	31	0.00	92	2	4283	0.02	48	0	16049	0.34
FA-100	97	3	40	0.00	87	6	4918	0.08	45	7	16364	0.38
FA-200	94	3	94	0.12	81	6	5498	0.09	37	5	17198	0.45
FA-500	89	3	119	0.24	72	6	6671	0.31	47	3	17801	0.31
FA-1000	83	3	189	0.39	61	13	6641	0.33	26	13	17672	0.42
FA-5000	54	3	121	2.26	30	11	6968	2.77	8	9	15808	1.67
FA-10000	34	3	134	4.04	15	3	5255	6.42	1	3	13094	6.54
FA-100000	20	3	0	8.91	10	0	0	10.58	1	0	16	9.77
REG	61	3	0	0.52	45	0	1	0.67	22	0	11	0.30
PEN	28	3	0	1.41	11	0	11	1.27	5	0	18	0.90
REG-FA-2	91	3	2	0.08	95	2	453	0.00	76	11	10409	0.07
REG-FA-5	96	3	3	0.01	97	3	536	0.00	73	14	10795	0.07
REG-FA-10	97	3	3	0.00	97	1	1644	0.01	61	15	12512	0.08

method attains the lowest average gap (0.07-0.08%) for the non-optimal cases.

2.5.3 Impact of scaling

Real-life optimization problems often have parameters with different orders of magnitude. For example, some parameters may have values around 10^3, while other parameters may take on values around 1. Such problems are badly scaled and are difficult to solve by optimization solvers. In order to investigate the impact of bad-scaling on the proposed solution method, the elements of matrices and vectors $c_1, d_1, A_1, B_1, b_1, c_2, d_2, A_2, B_2, b_2$ are multiplied by 10^z, where z follows a discrete uniform distribution with values $0, 1, 2, 3$ and probability 0.25 each. In doing so, one fourth of the elements is multiplied by 1, one fourth by 10, one fourth by 100, and one fourth by 1000. Table 2.4 contains the results of the randomly generated bad-scaled examples for the three sizes considered.

The first observation is that, although B&B and SOS1 still perform reasonably well for small- and medium-sized problems, none of the large-sized problems are solved to optimality and the average gap amounts to 57.79% and 57.06%, respectively. Note also that, for values of M below 1000, the Fortuny-Amat approach was infeasible for all cases of the three problem sizes. Besides, for larger values of M, the number of optimal cases is always below 10. The regulation and penalty methods also exhibit a very small number of optimal cases. On the other hand, the proposed method for $\mathcal{M} = 5$ achieves the lowest objective function in 91, 80 and 51 cases for small, medium and large problems, respectively. Furthermore, the average solution time for these sizes is 6 seconds, 1.5 hours and 6 hours, in that order. This means that none of the random problems with $n = 200$ was finished before 6 hours. For this reason, the results for large problems should be interpreted with caution, since few methods are able to provide solutions in most cases.

Table 2.4: Results: impact of scaling

	Small (n=50)				Medium (n=100)				Large (n=200)			
	#opt	#inf	time(s)	gap(%)	#opt	#inf	time(s)	gap(%)	#opt	#inf	time(s)	gap(%)
B&B	55	3	11535	54.58	12	0	19170	47.84	0	0	21605	57.79
SOS1	97	3	12	0.00	56	0	18419	7.02	0	0	21601	57.06
FA-5	0	100	-	-	0	100	-	-	0	100	-	-
FA-10	0	100	-	-	0	100	-	-	0	100	-	-
FA-20	0	100	-	-	0	100	-	-	0	100	-	-
FA-50	0	100	-	-	0	100	-	-	0	100	-	-
FA-100	0	100	-	-	0	100	-	-	0	100	-	-
FA-200	0	100	-	-	0	100	-	-	0	100	-	-
FA-500	0	100	-	-	0	100	-	-	0	100	-	-
FA-1000	0	95	4408	251.92	0	100	-	-	0	100	-	-
FA-5000	3	22	340	165.59	1	15	14822	57.00	3	4	21603	37.09
FA-10000	9	10	257	177.58	4	11	7878	121.04	0	3	20801	126.56
FA-100000	10	3	0	251.70	3	0	1	202.40	0	0	27	201.24
REG	12	37	1	9.25	4	41	4	6.68	4	85	36	2.51
PEN	3	77	1	38.02	0	89	9	30.60	0	98	16	2.80
REG-FA-2	76	3	4	1.36	68	3	3474	0.92	29	38	20764	1.90
REG-FA-5	90	3	6	0.75	80	11	5637	0.25	51	10	21397	1.61
REG-FA-10	91	3	15	0.54	82	6	10702	0.16	31	0	21788	3.51

Therefore, the average gap of 57.06% linked to the SOS1 method should be understood as the gap between the best solution provided by this method and the solution given by the proposed method after six hours of running time. The results in Table 2.4 clearly prove that the proposed solution approach is superior to the existing ones for bad-scaled problems.

2.6 Conclusions

Linear bilevel problems are non-convex and NP-hard and therefore, finding their optimal solution is computationally costly. In this paper we focus on methods that allow to directly solve LBLP using off-the-shelf optimization software. Among these methods, mixed-integer reformulations provide global optimal solutions at the expense of drastically increasing the computational time, which implies that they can only be applied to small problems. On the other hand, regularization approaches based on iteratively solving non-linear optimization problems can efficiently solve large-sized bilevel problems, but only guarantee local optimality of the MPCC reformulation.

In this paper we propose a new solution method that combines the advantages of the two aforementioned approaches. First, the regularization approach is used to efficiently find a local optimal point of the MPCC reformulation. Local optimal information is then used in the mixed-integer reformulation of the problem to i) provide initial values for the binary variables and ii) tune the large-enough constants. The results provided by this method have been compared with those obtained by other general purpose methods when solving a set of 900 randomly generated linear bilevel problems with different size, sparsity and scaling. These results show that the proposed method substantially outperforms the others in terms of number of cases solved to global optimality, average computational time and average optimality gap. For the largest examples, the proposed method achieved the optimal solution in 50% more cases than all the other methods, with an average time 30-95% lower, and an average optimality gap lower than 3.5% in all cases. Finally, it is worth highlighting that the proposed method does not require the adjustment of any large enough constant, and that setting the scaling parameter $\mathcal{M}$ to 5 or 10 is good enough to solve a wide set of different problems.

As future research, it must be investigated how to adapt the proposed methodology so that it can be applied to linear bilevel problems with upper-level constraints that involve both upper- and lower-level variables. Likewise, how to solve bilevel problems with an upper-level objective function that includes dual variables of the lower-level problem requires further research. Besides, the fact that the coefficients of the upper- and lower-level objective functions are all positive implies that the angle between the objective function vectors is statically small, which, in turn, may reduce the computational burden of solving the LBLP. Therefore, further investigation is required to analyze how the proposed method perform for arbitrary objective function parameters. The results presented in this paper could also be complemented by comparing the computational performance of different commercial solvers, such as GUROBI. Finally, testing the performance of the proposed solution approach in specific real applications is also left as ground for future research.

Chapter 3

The Impact of Short-term Variability and Uncertainty on Long-Term Power Planning

H. BYLLING, S. PINEDA AND T. BOOMSMA

Chapter Abstract

Traditionally, long-term investment planning models have been the apparent tool to analyse future developments in the energy sector. With the increasing penetration of renewable energy sources, however, the modelling of short-term operational issues becomes increasingly important in two respects: first, in relation to variability and second, with respect to uncertainty. A model that includes both may easily become intractable, while the negligence of variability and uncertainty may result in sub-optimal and/or unrealistic decision-making. This paper investigates methods for aggregating data and reducing model size to obtain tractable yet close-to-optimal investment planning decisions. The aim is to investigate whether short-term variability or uncertainty is more important and under which circumstances. In particular, we consider a generation expansion problem and compare various representations of short-term variability and uncertainty of demand and renewable supply. The main results are derived from a case study on the Danish power system. Our analysis shows that the inclusion of representative days is crucial for the feasibility and quality of long-term power planning decisions. In fact, we observe that short-term uncertainty can be ignored if a sufficient number of representative days is included.

3.1 Introduction

Long-term planning problems related to the electricity market, system and/or network arise in multiple contexts: generation expansion (Baringo and Conejo, 2012a; Jin et al., 2011; Pineda and Morales, 2016), transmission expansion (Orfanos et al., 2013; Hemmati et al., 2014; Pozo et al., 2013), storage investment (Babrowski et al., 2015; Jabr et al., 2015; Ghofrani et al., 2013) etc. Fundamental to all of these problems is the modelling of short-term system operation, ideally accounting for both dynamics and uncertainty. With the penetration of renewable energy sources in many power systems, not only demand but also part of supply vary over time and stochastically. For instance, wind and solar power production is driven by weather conditions and thereby varies from hour to hour and from day to day and is difficult to accurately predict. To maintain the balance between demand and supply at all times, the system should be sufficiently flexible. The increasing penetration of renewables implies a greater need for flexibility in conventional generation and accentuates the effects of inter-temporal constraints and balancing costs (Poncelet et al., 2016). In particular, if demand is higher or lower than renewable production, conventional generation sources must be able to increase or decrease production accordingly. To handle variations over time, production must be able to adjust from one hour to the next. This type of flexibility is restricted by the technical specifications of the operating units, usually modelled by so-called ramping constraints. Stochasticity is often handled through the division of the market into a day-ahead market for commitment of predicted demand and supply and a real-time balancing market that allows for adjustments at additional costs. In theory, these short-term characteristics could be explicitly modelled in the long-term planning. In practise, however, the computational effort to solve the planning problem becomes excessive (Poncelet et al., 2016). In fact, a complete representation of ramping abilities requires an hourly discretisation of a multi-decade time horizon, whereas the modelling of balancing decisions involves discretising the continuous distribution of demand and renewable supply. As a result, the model size increases with the number of time periods and the number of scenarios describing uncertainty.

The negligence of variability and uncertainty may result in sub-optimal and/or unrealistic decision-making. Indeed, failure to account for ramping restrictions and balancing costs may significantly overestimate flexibility and suggest investments in renewable energy sources beyond what is the physically and/or economically feasible. A compromise between computational effort and accuracy of the model results is provided by aggregated representations of time and uncertainty. The present paper investigates and compares methods for aggregating data to obtain tractable yet close-to-optimal investment planning decisions. We consider the following types of data aggregation:

- Representative hours: Hours are divided into a number of groups, each representing a given number of hours. The division is based on clustering of data and carried out for each hour independently.

- Representative days: Days are divided into a number of groups, each representing a given number of days. The division is likewise based on clustering of data but carried out for a day at the time, respecting the chronology of the hours.

- With short-term uncertainty: The distributions of unknown parameters are discretised, using a limited number of scenarios.

- Without short-term uncertainty: The distributions of unknown parameters are replaced by their expected values.

By disentangling short-term variability and uncertainty, we investigate which is more important, under which circumstances and how to obtain suitable data representations. To the best of our knowledge and as evidenced by the following review, the comparison of such modelling aspects cannot be found in the existing literature.

In evaluating the impact of short-term dynamics and uncertainty, we use a family of generation expansion models. All models take the perspective of a central planner, minimising total costs of meeting demand and a minimum requirement for renewable supply by investing and operating accordingly. We consider a planning horizon of a single year and with an hourly discretisation. Investments are one-time installations, whereas production decisions are made for every time period. Operation is subject to a number of technical constraints, including ramping restrictions, and the structure of the market, including a day-ahead market for commitment of predicted demand and supply and a real-time balancing market. We consider energy-only markets and the implications of short-term uncertainty and dynamics in these. Other related markets, e.g. capacity markets, are not considered.

The performance of these models is compared with respect to both the quality of the expansion plan and the computation time. The model results are illustrated for a case study on the Danish power system.

3.2 Literature review

Various methods have been suggested to discretise the time horizon of long-term planning problems in a way that enables computational tractability (Haydt et al., 2011). Most of these aim at aggregating hours, days and years to achieve an acceptable model size.

With an hourly representation (often referred to as 'time slices'), time periods are represented by the values of their state variables (demand, wind power production, etc.) and grouped according to these. A traditional example is the load duration curve for which time periods are sorted with respect to demand level and grouped into blocks of a given duration (Stoft, 2002). This approach is used in the generation expansion planning of Pineda et al. (2014); Chaton and Doucet (2003) and Jin et al. (2011). Bertsch and Fichtner (2015) likewise use the PERSEUS-NET model with a load duration curve in multi-criteria analysis of power generation and transmission planning. As an alternative to sorting the hours throughout the year, demand can be clustered according to additional information such as seasonal variation (Pozo et al., 2013). Baringo and Conejo (2013) include both wind production and demand profiles in the clustering, and the correlation between the two variables is taken into account. With the same purpose, Wogrin et al. (2014) introduce another method based on a duration curve and chronological transitions between states. The method estimates transitions from one state to another and incorporates inter-connecting constraints on the significant transitions.

Representative days consists of choosing a number of days, or connected time periods in general, to represent the planning horizon. In this way, inter-temporal dynamics can be preserved within the time periods. An example is given by Babrowski et al. (2015) who investigate long-term storage planning. Another example is by Ghofrani et al. (2013) who use a representative scheduling period of 24 hours to optimize storage placement. A fully dynamic model including all hours of the entire planning horizon have been proposed by Jabr et al. (2015). The model relates to storage investment and relies on robust optimisation. As the fully dynamic setup may very well be intractable for larger problems, Pina et al. (2013) use 12 representative days, 3 of each season in a year, for generation expansion in electricity systems with high penetrations of renewable energy. In a similar fashion, Ma et al. (2013) select five whole weeks to represent demand variations throughout the year in a unit construction and commitment problem and use this to analyse power system flexibility. Representative weeks are also used by Sisternes et al. (2013) to approximate the net load in a generation expansion problem. The paper demonstrates how the quality of investments depends on the choice of representative weeks. In contrast, however, Nahmmacher et al. (2016) present a clustering technique that determines the representative days and show that using representative weeks instead of days increases the required number of hours to obtain a sufficiently good representation of the data.

The main difference between hourly and daily aggregation is the ability to include short-term operational flexibility. A daily representation may account for short-term flexibility by including inter-temporal constraints. Such constraints cannot be incorporated with an hourly representation. With an increasing penetration of non-dispatchable renewable energy sources, however, the representation of short-term dynamics in long-term models becomes increasingly important (Pfenninger et al., 2014). This is supported by Poncelet et al. (2016) who study and compare the effect of using the hourly and daily representations. More specifically, the results confirm that the need for inter-temporal techno-economic constraints increases with the penetration of renewable energy. Slednev et al. (2017) consider a combination of representative days and hours. These are determined in a k-means clustering method, using an error measure that measures grid-impact. The time resolution for both the hourly and the daily aggregation is usually hours. An example of a finer time resolution (such as 15 or 30 minutes) is provided by Schwarz et al. (2018) who analyse residential heat storage with photo-voltaic power generation.

In addition to variability over time, long-term planning naturally involves uncertainty of key parameters. Long-term uncertainty relates to the future development of demand and costs parameters. Nevertheless, uncertainty also arises in the short-term. Traditionally, the main concern has been the stochastic variability of demand. At present, however, demand can be rather accurately predicted 24 hours in advance (Aneiros et al., 2016). On the other hand, a high penetration of renewable electricity sources in modern power systems introduces a significant source of short-term uncertainty. Some authors ignore this short-term uncertainty by assuming perfect information of future power production and model system operation as deterministic. This approach can be found in the generation expansion problem of Jin et al. (2011), who assume long-term demand and price levels to be uncertain but solve the short-term scheduling problem with perfect information. The same approach is taken by Pozo et al. (2013) and Ludig et al. (2011).

In contrast, Baringo and Conejo (2013, 2011) and Ma et al. (2013) model the system operation problem as a two-stage stochastic program with a day-ahead market as the first stage and a balancing market as the second stage. In the day-ahead market, production decisions are made according to expected demand and wind production. Uncertainty in the real-time balancing market is modelled by scenarios for wind production. In each scenario, day-ahead commitments can be adjusted to realised production by making balancing decisions (with potentially additional costs). Further details are provided by Pineda and Morales (2016).

The remainder of this paper is organised in the following way. The generation expansion problem is presented in Section 3.3.1 and the different approaches to including short-term characteristics are discussed in Section 3.3.2. Section 3.4 provides a small example which serves as a basis for analysing the effects of short-term variability and uncertainty on the solutions to the generation expansion problem. A larger case study further elaborates on this in Section 3.5. Section 3.6 concludes the paper.

3.3 Investment optimisation and aggregation of data

The purpose of this paper is to compare different approaches to represent short-term dynamics and uncertainty in long-term planning problems. In particular, we consider four

Table 3.1: Nomenclature

Sets	
$\mathcal{G}$	set of production units
$\mathcal{G}^w$	set of wind production units
T	set of time periods
T_d	set of time periods, except the last, within an aggregation period (e.g. $\{1, \ldots, 23\}$ for a day)
S	set of scenarios for short-term uncertainty
Parameters	
c_g^I	linear investment cost of unit g (€/MW)
c_g	linear production cost of unit g (€/MWh)
c_g^+	additional cost of upward balancing of unit g (€/MWh)
c_g^-	opportunity cost of downward balancing of unit g (€/MWh)
r_g^D	ramp down rate of unit g (p.u.)
r_g^U	ramp up rate of unit g (p.u.)
ρ_{gt}	predicted production factor of unit g at time t (p.u.)
$\tilde{\rho}_{gts}$	realised production factor of unit $g \in \mathcal{G}^w$ at time t in scenario s (p.u.)
κ	minimum wind penetration (%)
v^L	cost of load shedding (€/MWh)
v^S	cost of wind curtailment (€/MWh)
ν_t	load factor at time t (p.u.)
$\bar{d}$	maximum load (MWh)
τ_t	duration of time period t
π_s	probability of short-term scenario s
Variables	
$\bar{p}_g$	investment capacity of unit g
p_{gt}	scheduled production of unit g at time t
k_t	scheduled load shedding at time t
l_t	scheduled wind curtailment at time t
p_{gts}^+	real-time upward balancing of unit $g \in \mathcal{G} \setminus \mathcal{G}^w$ at time t in scenario s
p_{gts}^-	real-time downward balancing of unit $g \in \mathcal{G} \setminus \mathcal{G}^w$ at time t in scenario s
$\tilde{p}_{gts}$	real-time production of unit $g \in \mathcal{G} \setminus \mathcal{G}^w$ at time t in scenario s
Δk_{ts}	real-time load shedding at time t in scenario s
Δl_{ts}	real-time regulating wind curtailment at time t in scenario s

different approaches for aggregation of data in a generation expansion problem. All aggregation approaches are used in combination with the same optimisation model. The model is presented in Section 3.3.1 whereas the data aggregation approaches are defined in Section 3.3.2.

3.3.1 Model

The model takes the perspective of a central planner, with the objective of minimising total costs of meeting a fixed demand by investing and operating accordingly. The decisions obtained by this model coincide with those of a generation expansion equilibrium with perfectly competitive and risk-neutral power producers (Ehrenmann and Smeers, 2011). Furthermore, since demand is assumed to be inelastic, minimising costs is equivalent to maximising social welfare (Gabriel et al., 2013). The model includes a minimum wind penetration constraint that represents a political target for the penetration of renewables, like those imposed as part of the political agenda in the European Union (European Commision, 2014). Note that the minimum wind penetration is given as an exogenous parameter, whereas the decision to invest in wind capacity is endogenous to the model. For simplicity, we focus on short-term uncertainty of wind power production, although the model could easily be extended to include demand uncertainty, stochastic capacity availability etc.

The modelling of generation expansion is divided into two: investment and market clearing.

Investment

Generally, generation expansion models are classified as either static (single-year) or dynamic (multiple-year) models (Akbari et al., 2012). For simplicity and as is common practice in the literature (Baringo and Conejo, 2012a; Wang et al., 2009; Murphy and Smeers, 2005), our investment model is static with a single-year planning horizon. Thus, investment variables relate to a one-time installation of generation capacity while system operation involves production decisions for every time period (e.g. hour) throughout the year (in the following referred to as the target year). We assume that at the beginning of the year, there is no existing capacity in the market, i.e. we take a greenfield approach. We also assume that new generation capacity is available once installed, meaning that construction time is zero.

Market clearing

Our market model consists of a day-ahead market and a real-time balancing market (Pineda et al., 2014). For each time period, the day-ahead market is modelled as an economic dispatch problem in which the generating units are dispatched to meet demand at minimal costs given the forecasted wind power production. In the balancing market, stochastic wind power production is realised, and the demand must be met given this realised wind power production. This may require re-dispatch of power generation and incurs an additional (positive) balancing cost. Such costs may be justified by increased stress on the generation units. Balancing costs are further discussed in Section 3.5.1.

Our techno-economical constraints include ramping constraints of the generation units, but for simplicity, we do not consider a unit commitment problem (Poncelet et al., 2016). This simplification is likewise discussed in Section 3.3.2.

To ensure that the expansion problem is feasible irrespective of the investment plan, we include the possibility of load shedding and wind curtailment during economic dispatch. If installed capacity is insufficient to meet demand, load shedding occurs. Likewise, if the realised wind power production exceeds demand, wind curtailment occurs. Load shedding and wind curtailment are present in both the day-ahead and the balancing market. The realised load shedding or wind curtailment is given as the sum of the scheduled load shedding or wind curtailment and the adjustments to these. Only the realised load shedding or wind curtailment is penalised in the objective function. As an estimate for load shedding and wind curtailment costs, we use the maximum and minimum price caps for the market in question (Stoft, 2002). These costs serve as compensation to the consumer and the wind power producer, respectively, should load shedding or wind curtailment occur.

The static generation expansion problem is formulated as follows:

$$\min \sum_{g\in\mathcal{G}} c_g^I \bar{p}_g + \sum_{t\in T} \tau_t \left(\sum_{g\in\mathcal{G}} c_g p_{gt} + \sum_{g\in\mathcal{G}\setminus\mathcal{G}^w} \sum_{s\in S} \pi_s \left((c_g + c_g^+) p_{gts}^+ - (c_g - c_g^-) p_{gts}^- \right) \right) + \sum_{t\in T} \tau_t \sum_{s\in S} \pi_s \left(v^L (k_t + \Delta k_{ts}) + v^S (l_t + \Delta l_{ts}) \right) \tag{3.1a}$$

$$\text{s.t.} \sum_{g\in\mathcal{G}\setminus\mathcal{G}^w} p_{gt} + k_t - l_t = \nu_t \bar{d} - \sum_{g\in\mathcal{G}^w} \rho_{gt} \bar{p}_g,\ \forall t \in T \tag{3.1b}$$

$$0 \le p_{gt} \le \bar{p}_g,\ \forall g \in \mathcal{G} \setminus \mathcal{G}^w, \forall t \in T \tag{3.1c}$$

$$-r_g^D \bar{p}_g \le p_{gt} - p_{g(t-1)} \le r_g^U \bar{p}_g,\ \forall g \in \mathcal{G} \setminus \mathcal{G}^w, t \in T_d \tag{3.1d}$$

$$0 \le k_t \le v_t \bar{d},\ \forall t \in T \tag{3.1e}$$

$$0 \le l_t \le \sum_{g\in\mathcal{G}^w} \rho_{gt} \bar{p}_g,\ \forall t \in T \tag{3.1f}$$

$$\sum_{g\in\mathcal{G}\setminus\mathcal{G}^w} \tilde{p}_{gts} + \Delta k_{ts} - \Delta l_{ts} = \nu_t \bar{d} - \sum_{g\in\mathcal{G}^w} \tilde{\rho}_{gts} \bar{p}_g,\ \forall t \in T, s \in S \tag{3.1g}$$

$$0 \le \tilde{p}_{gts} \le \bar{p}_g,\ \forall g \in \mathcal{G} \setminus \mathcal{G}^w, t \in T, s \in S \tag{3.1h}$$

$$-r_g^D \bar{p}_g \le \tilde{p}_{gts} - \tilde{p}_{g(t-1)s} \le r_g^U \bar{p}_g,\ \forall g \in \mathcal{G} \setminus \mathcal{G}^w, s \in S, t \in T_d \tag{3.1i}$$

$$0 \le k_t + \Delta k_{ts} \le v_t \bar{d},\ \forall t \in T, s \in S \tag{3.1j}$$

$$0 \le l_t + \Delta l_{ts} \le \sum_{g\in\mathcal{G}^w} \tilde{\rho}_{gts} \bar{p}_g,\ \forall t \in T, s \in S \tag{3.1k}$$

$$\tilde{p}_{gts} = p_{gt} + p_{gts}^+ - p_{gts}^-,\ \forall g \in \mathcal{G} \setminus \mathcal{G}^w, t \in T, s \in S \tag{3.1l}$$

$$\sum_{t\in T} \tau_t \sum_{s\in S} \pi_s \Big(\sum_{g\in\mathcal{G}^w} \tilde{\rho}_{gts} \bar{p}_g - (l_t + \Delta l_{ts}) \Big) \ge \kappa \sum_{t\in T} \sum_{s\in S} \pi_s \tau_t \big(\nu_t \bar{d} - (k_{ts} + \Delta k_{ts}) \big) \tag{3.1m}$$

$$p_{gts}^+, p_{gts}^- \ge 0,\ \forall g \in \mathcal{G} \setminus \mathcal{G}^w, t \in T, s \in S \tag{3.1n}$$

$$\bar{p}_g \ge 0,\ \forall g \in \mathcal{G} \tag{3.1o}$$

The objective function in (3.1a) accumulates investment costs, day-ahead planning costs and expected real-time balancing costs, including load shedding and wind curtailment costs. Day-ahead planning costs of time period t are weighted by the parameter τ_t due to the aggregation of data (see Section 3.3.2) and balancing costs of scenario s are weighted by the probability π_s. The day-ahead operational constraints (3.1b), (3.1c), (3.1d), (3.1e) and (3.1f) cover demand satisfaction, capacity limits, ramp rate restrictions and bounds on load shedding and wind curtailment, respectively. Note that there are no ramping constraints between aggregation periods (e.g. days), as indicated by the set T_d, see also Section 3.3.2. Similar operating constraints apply to the balancing market in (3.1g), (3.1h), (3.1i), (3.1k) and (3.1j). Moreover, the realised dispatch of generation is defined in (3.1l). Finally, (3.1m) requires a percentage of κ of the annual demand to be covered by wind power production, while (3.1n),(3.1o) ensure non-negativity of the relevant variables.

3.3.2 Variability and uncertainty

We consider different representations of data with respect to two major short-term aspects: the aggregation over time and the representation of uncertainty.

Aggregation over time

We consider two approaches to aggregation of data over time: Representative hours and representative days.

- **Representative hours:** Aggregation by hours means that hours are evaluated separately with respect to their state values, e.g. demand and wind production. Hours are clustered into a number of groups, each representing a number of ”similar” hours in a year. The index of a time period t therefore refers to a group. The duration of a group is given by τ_t, indicating the number of hours represented. Due to the loss of chronology, ramping constraints cannot be considered, and hence, the constraints (3.1d) and (3.1i) are omitted from the model.

- **Representative days:** Aggregation by days means that hours are evaluated while respecting the order of their state values throughout a day. Days are likewise clustered into a given number of groups, each representing a number of ”similar” days in a year. A representative day has an associated weight, referring to the number of days represented by the group and given by τ_t. This weight applies to time periods and is the same for all time periods t of the same representative day. The index of the hourly time periods t runs from 1 to $24 * N$, where N is the number of representative days. Moreover, the set T_d contains all hours except the last of each day, i.e. T_d includes indices that are not multiples of 24. With preservation of chronology within a day, ramping constraints (3.1d) and (3.1i) are included in the model, although not between days.

Alternative choices of aggregation over time, such as a combination of representative hours and representative days or time periods of more or less than an hour, could be considered, see for example Slednev et al. (2017). We briefly discuss the combination of representative hours and representative days in Section 3.3.2.

Representation of uncertainty

To evaluate the importance of including short-term uncertainty, two different approaches are considered: one with short-term uncertainty and another without short-term uncertainty.

- **With short-term uncertainty:** Section 3.3.1 describes the two-stage day-ahead and balancing market clearing in the presence of short-term uncertainty. In the following, we refer to the model with stochastic market clearing. The distribution of the state values, e.g. wind power production, is discretised, using a limited number of scenarios ($|S| > 1$) with corresponding probabilities. Depending on the representation of data over time, scenarios either consist of hourly values of production or of daily production schedules.

- **Without short-term uncertainty:** In the absence of short-term uncertainty, the balancing market serves no purpose and the day-ahead market clearing is sufficient. We refer to this as conventional market clearing. The distribution is replaced by the expected hourly wind power production ($|S| = 1$).

Overview

We consider four combinations of data aggregation over time and representation of uncertainty and the resulting four models for the generation expansion problem: daily representation and conventional market clearing (DC), daily representation and stochastic market clearing (DS), hourly representation and conventional market clearing (HC), and hourly representation and stochastic market clearing (HS). These four models are found in Table 3.2, where an acronym indicates the model and the number of days or hours included.

Table 3.2: Overview of models for generation expansion planning, categorised by data aggregation over time and representation of uncertainty.

S-T uncert.	Data agg.	Acronym	Optimisation Problem		
Without	Rep. days	DC-(# of days)	min s.t.	(3.1a) (3.1b)-(3.1o)	$|S| = 1$
With	Rep. days	DS-(# of days)	min s.t.	(3.1a) (3.1b)-(3.1o)	$|S| > 1$
Without	Rep. hours	HC-(# of hours)	min s.t.	(3.1a) (3.1b),(3.1c),(3.1g)-(3.1h),(3.1j)-(3.1o)	$|S| = 1$
With	Rep. hours	HS-(# of hours)	min s.t.	(3.1a) (3.1b),(3.1c),(3.1g)-(3.1h),(3.1j)-(3.1o)	$|S| > 1$

Limitations of the methodology

In our analysis, we use the most simple model that includes both short-term variability and uncertainty. Focus is on whether variability or uncertainty is more important and in which situations. Our simplifications, however, do introduce limitations to the scope of the paper.

A main simplification is to represent flexibility using ramp rates only and not include the typical features of a unit commitment problem such as start-up costs, minimum up- and down-time constraints etc. However, we expect that the effects of short-term variability and uncertainty will be more pronounced with less flexibility in the power system, and thus, in the presence of unit commitment constraints.

Furthermore, we confine ourselves to the temporal dimension and do not consider the spacial dimension of a power network. The representation of the network could provide both flexibility and restrictions to the optimisation model, and thus, both reduce and amplify the effects of variability and uncertainty. When clustering days or hours, for example, the effect of the peak flow on the network should ideally be considered (Schwarz et al., 2018).

For simplicity, we use either representative days or representative hours and not a combination of days and hours. A hybrid approach is proposed by Slednev et al. (2017) who report promising computational results. The number and selection of representative days and hours, however, are critical to the performance.

Finally, we consider a greenfield system to highlight the differences in the expansion plans resulting from the four models of variability and uncertainty. Such differences would be diluted if existing capacities were considered. In other words, if considering a brownfield system, the differences between the four models would be much less.

3.4 Illustrative example

We start by illustrating the effects of short-term uncertainty and variability on a stylised example.

Demand and wind power data is from the pricing zone DK1 (Nord Pool AS, 2017). We assume deterministic demand, using a single representative day, and stochastic wind production, characterised by two scenarios with the same probability. These two scenarios correspond to the wind capacity factor in DK1 for two representative days of 2017 and were determined by scenario reduction techniques, see Section 3.5.1. Demand and wind production profiles are shown in Figure 3.1. We consider the models DS-1, including ramping and stochastic market clearing, DC-1 with ramping only, HS-24 with stochastic market clearing only, and HC-24 excluding both ramping and stochastic market clearing. For the notation, see Table 3.2.

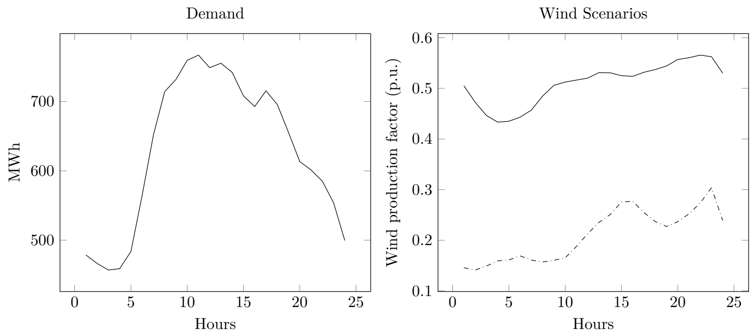


Figure 3.1: Demand and wind production profiles of a representative day and two wind power scenarios, respectively.

We consider the following generation units named according to the technology with most similar characteristics: wind turbines, nuclear, gas and coal. To illustrate the differences resulting from choice of modelling, we divide the flexible gas units into two different types: a gas unit that is flexible in the day-ahead market (with ramping ability, but high balancing costs) and a gas unit that is flexible in the balancing market (with low balancing costs, but no ramping ability). In reality, as in the case study of Section 3.5, however, most gas units are flexible in both markets. The nuclear unit is assumed inflexible in both the day-ahead and in the balancing market. The data for these units is shown in Table 3.3. Furthermore, load shedding and wind curtailment costs are set to $v^L = v^S = 500$ €/MWh.

To evaluate and compare investment decisions across the four models, the following procedure is used:

1. Solve each of the problems DC-1, DS-1, HC-24 and HS-24, see Table 3.2. for their definition.

Table 3.3: Generation unit data for an illustrative example. Source: (Ea Energianalyse, 2014) and (Schröder et al., 2013)

g	wind	coal	gasDA	gasBal	nuclear
c_g^I (T€/MW)	124	106	51	51	150
c_g (€/MWh)	0	31.4	63.1	63.1	15.4
c_g^+, c_g^- (€/MWh)	0	5.23	500	4.51	500
r_g^D, r_g^U (p.u.)	1	0.3	0.7	0	0

2. For each of the optimal solutions to DC-1, HC-24 and HS-24, fix the investment decision and solve the generation expansion problem DS-1 (without minimum wind penetration constraints (3.1m)). Record the objective function value.

Since the DS-1 model includes both short-term variability and uncertainty, we use this as the baseline for evaluation and comparison. Thus, by definition this model provides the optimal investment decisions and the minimal costs. We evaluate the objective function value of using the (feasible, but not necessarily optimal) investment capacities from DC-1, HC-24 and HS-24, which are at least as high as those of DS-1. The difference in objective function values can be interpreted as the costs of disregarding variability and/or uncertainty in the optimisation. Figure 3.2 show the objective function values of the DS-1 model plotted as functions of the measured wind penetration. Note that whereas the required wind penetration is exogenous, the measured wind penetration is a function of capacity, and hence, endogenous. For $\kappa = 0.2$ and $\kappa = 0.6$, Table 3.4 shows optimal investment capacities for each of the problems DS-1, DC-1, HS-24 and HC-24.

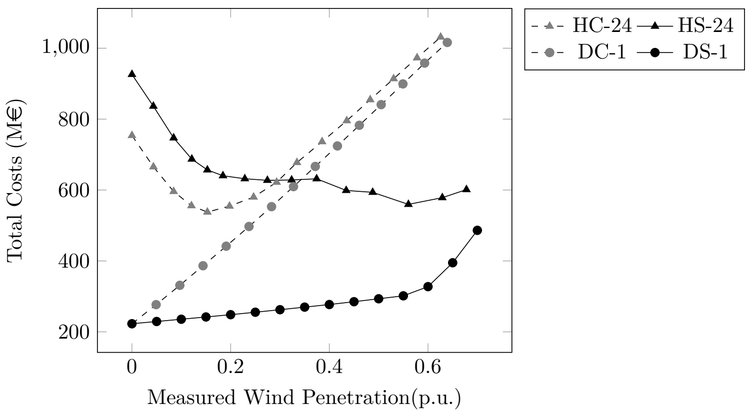


Figure 3.2: Total costs from fixing the optimal investment capacities from the problems DC-1, DS-1, HC-24 and HS-24 in DS-1. As functions of the measured wind penetration.

The results show that since the HC-24 model disregards both variability and uncertainty, investments are mainly in the inflexible base nuclear generation, meaning that load shedding costs are high, cf. Table 3.4 and Figure 3.2. The minor investment in day-ahead flexible gas serves to cover peak load hours and is almost the same for all wind penetrations. With higher wind penetration, the major change in investments is substitution of wind for nuclear. Total costs decrease with wind penetrations up to $\kappa = 0.2$ since wind power provides some flexibility through curtailment. For wind penetrations from $\kappa = 0.2$ and up, total costs increase, as the cost savings of wind power are out-weighted by the costs insufficient balancing capacity and the resulting load shedding.

Table 3.4: Investment decisions in MW for $\kappa = 0.2$ and $\kappa = 0.6$. LS is load shedding as a ratio of demand and WC is wind curtailment as a ratio of realised wind production.

model	κ	wind	nuclear	gasDA	gasBal	coal	LS	WC
DC-1	0.2	352	355	288	0	0	0.08	0.05
DS-1	0.2	352	300	279	103	24	0	0
HC-24	0.2	352	577	66	0	0	0.07	0.24
HS-24	0.2	352	528	0	173	0	0.12	0.24
DC-1	0.6	1060	134	266	0	0	0.2	0.09
DS-1	0.6	1076	0	208	288	86	0	0.02
HC-24	0.6	1056	326	76	0	0	0.19	0.12
HS-24	0.6	1071	162	48	372	0	0.06	0.08

As with the HC-24 model, the HS-24 model invests in the nuclear unit to cover base load. Moreover, when accounting for uncertainty, the model also invests in the gas unit flexible in the balancing market to provide peak load capacity in some hours. The choice of gas unit, however, means that the total costs of the HS-24 model are higher than those of the HC-24 model for low wind penetrations. As wind penetration increases, total costs of the HS-24 model first decreases and then stabilises. The reason for decreasing costs is that wind power provides cost savings through the flexibility to curtail, but also that the installed balancing capacity handles the uncertainty with minimal load shedding. As with the HC-24 model, minor installations in day-ahead flexible gas serve the peak load.

The DC-1 model includes variability and thus invests in day-ahead flexible gas in addition to nuclear. The deterministic model, however, disregards balancing and therefore investment capacities neither include the gas units flexible in this market nor coal units. This leads to expensive wind curtailment and/or load shedding as wind penetration increases, and thus, increasing total costs.

By definition, the DS-1 model provides the lowest costs for all wind penetration levels. By accounting for both variability and uncertainty, this model avoids significant wind curtailment and load shedding costs. The higher wind penetrations, the higher the total costs. The reason is that higher requirements of wind penetration leads to higher investment costs of wind investments and, for very high wind penetration, wind curtailment costs in some hours and scenarios. For low penetrations, the DS-1 model produces a combination of all generation technologies to serve flexibility needs both in terms of ramping and balancing. For wind penetrations of $\kappa = 0.6$ and up, however, nuclear is substituted by the other technologies.

We conclude this example by noting that our model clearly captures the impact of the two short-term effects: uncertainty and variability. The results in Figure 3.2 show that representative days are very valuable for incorporating the short-term variability, although cost savings are less for high wind penetration levels. In contrast, for wind penetrations above a certain level, the inclusion of the stochastic market clearing provides significant cost savings.

3.5 Case study

We continue by applying the modelling framework introduced in Section 3.3 to data from the Nordpool bidding region DK1 covering the Western part of Denmark (Nord Pool AS, 2017). The wind penetration target is set to 30%, as is the Danish 2020 renewable energy target (The Danish Government, 2013).

3.5.1 Data

We use historical market data from 2014. The data includes aggregated demand, wind power forecasts and realised wind power production for the entire region. The data is available on an hourly basis and is normalised by total capacity.

With data available for both forecasted and realised wind power production, we model the hourly forecast error:

$$\tilde{\rho}_t = \rho_t + e_t, \tag{3.2}$$

where $\tilde{\rho}_t$ is the realised wind production, ρ_t is the wind production forecast and e_t is the forecast error, all given as capacity factors. Recall that only the forecast is known when the day-ahead market clears, whereas the forecast error is realised at the time of clearing the balancing market.

For simplicity, we fit the wind forecast errors to an ARMA time series model, assuming that the process is stationary with decaying autocorrelations. More detailed approaches to modelling wind forecasting errors are given by Bludszuweit et al. (2008) and Box et al. (1994). By inspection of the autocorrelation functions, we choose an AR(2) model on the following form:

$$e_t = \phi_1 e_{t-1} + \phi_2 e_{t-2} + \epsilon_t, \quad \epsilon_t \sim N(0, \sigma^2). \tag{3.3}$$

Here, the error terms ϵ_t capture variations in the historical data that are not explained by previous observations, and are assumed independent and identically normally distributed around zero. Fitting this model to the forecast errors from DK1 in 2014, we obtain the estimates $\phi_1 = 1.186$ and $\phi_2 = -0.294$ and the z-test statistics indicate that the coefficients are statistically significant (Pr$(> |z|) < 2.2 \cdot 10^{-16}$ for both coefficients). The assumption of normally distributed residuals is confirmed to a satisfying extent by histograms and QQplots.

The time series model is used to generate scenarios for realised wind power production for each day of the target year. The model takes as input the observed forecast errors of the last two hours from the previous day. For the following 24 hours, we sample the error term and recursively compute the forecast errors. We generate 1000 scenarios of wind forecast errors and reduce these to 10 by the scenario reduction technique of Dupacova et al. (2003). The aim is to accurately represent uncertainty while the model remains computationally tractable (Morales et al., 2009). Using (3.2), these scenarios are translated into realised production. The result is 10 24-dimensional scenario vectors, $(\tilde{\rho}_{1s}, \dots, \tilde{\rho}_{24s}), s = 1, \dots, 10$ of realised wind power production for each of the 365 days of the year. In Figure 3.3 we plot the scenarios and the observed historical wind power production of three selected days.

The data for conventional generation taken from Ea Energianalyse (2014) and Schröder et al. (2013), and chosen to represent a diverse selection of production units. All costs taken from Ea Energianalyse (2014) are 2020 predictions and investment costs are annualised with expected lifetime of the technology and using a discount rate of 4%. The expected lifetime is defined as the minimum of the technical and the economical lifetime of a unit, where the technical lifetime is taken from Danish Energy Agency (2012) and the economical lifetime captures the number of years operation is profitable, taking future discounted fuel and CO2 prices into account. The four production units are: wind power

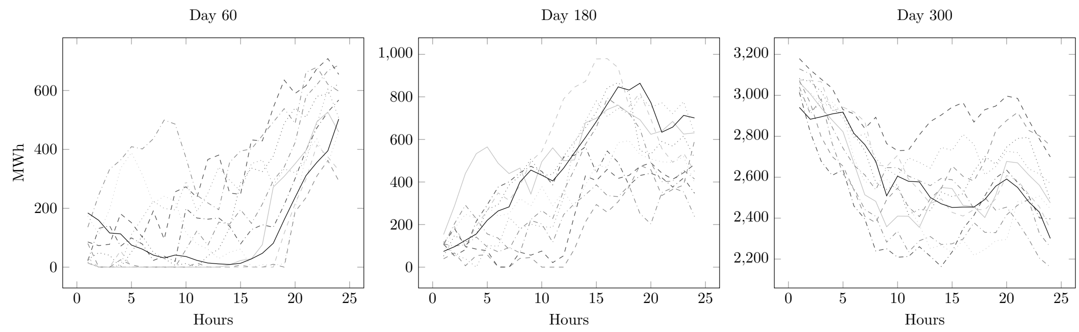


Figure 3.3: Scenarios of wind production (dashed lines) and historical wind production (solid line) for 3 selected days.

(wind), coal-fired pulverised fuel power plant (coal), combined-cycle gas turbine (gas) and nuclear. Table 3.5 provides the data.

Table 3.5: Generation unit data for the case study. Source: Ea Energianalyse (2014) and Schröder et al. (2013)

g	wind	coal	gas	nuclear
c_g^I (T€/MW)	124	106	51	150
c_g (€/MWh)	0	31.4	63.1	15.4
c_g^+, c_g^- (€/MWh)	0	5.23	4.51	25.67
r_g^D, r_g^U (p.u.)	1	0.3	0.7	0.03

Ramp rates of the *gas* and *coal* units are not publicly available. We, therefore, derive the ramp rates from the aggregated hourly output for each technology, collected from Entsoe (2016) and with outliers removed. More specifically, we use the maximum hourly change in aggregate output to approximate the aggregate ramp rate. The ramp rate is finally normalised by the maximum hourly output. For simplicity and supported by the data, upward and downward ramp rates are the same.

Balancing costs are modelled as follows. We assume that the balancing costs are increasing in production costs and decreasing in ramp rates and consider the following relation for c_g^+ and c_g^-:

$$c_g^+ = M \cdot \frac{c_g}{r_g^u} \qquad \text{and} \qquad c_g^- = M \cdot \frac{c_g}{r_g^d}, \tag{3.4}$$

where M is an adjustment factor. This M is estimated from historical balancing prices, cf. Nord Pool AS (2017). The average balancing price in DK1 for 2014 is 6.30 €/MWh, and thus, we set $M = 0.05$ to achieve the balancing costs in Table 3.5. We further consider the case of a zero balancing cost for all units and present both cases in the results.

The load shedding costs are estimated by the maximum price of electricity. From Nord Pool AS (2017) we note that the maximum price in DK1 is 3000 €/MWh, and thus, we set $v^L = 3000$ €/MWh. Similarly, we estimate the wind curtailment costs by the minimum price. The minimum price of electricity in DK1 is -500 €/MWh, and the wind curtailment costs are therefore set to $v^S = 500$ €/MWh.

Data Aggregation

The technical literature includes several methods to select representative days or hours. In Hastie et al. (2009) and ElNozahy et al. (2013), representative days or weeks are chosen using classical clustering techniques such as K-means or hierarchical clustering. New methods to select representative days have recently been proposed. For instance, Poncelet et al. (2017) provide a novel optimisation-based approach to select representative periods. Similarly, Liu et al. (2017) propose a modified hierarchical clustering procedure to choose a reduced set of representative days that retains important statistical features of the input data such as correlation.

Our data aggregation is carried out using the GAMS/SCENRED tool (Römisch, 2002). Although this tool is intended for scenario reduction, the clustering algorithm may likewise apply for the reduction of hours or days to a smaller subset, with each day or hour of a year being equally probable. The GAMS/SCENRED tool is an out-of-the-box tool and the reduction selects a specific hour or day as representative. When clustering by hour, we consider all 8760 hours of historical wind production and demand data and reduce to the required number of representative hours, as indicated by the suffix of the model name, e.g. HC-24. When clustering by day, we likewise use all 365 days of historical data and reduce to the required number of days, likewise revealed by the suffix of the model name, e.g. DC-1.

3.5.2 Results

We consider the four combinations of data aggregation over time and representation of uncertainty and the resulting models for the generation expansion problem, cf. Section 3.3.2. The results are divided into two sections: First, we analyse these models using the full data set (we refer to the models HC-8760, HS-8760, DC-365 and DS-365 as full models). Secondly, we include only a subset of the data obtained via aggregation and benchmark against the full DS-365 model, using the procedure of Section 3.4. The full results are included in Appendix A.1.

Technical Details

Our model is implemented using GAMS 24.7.4 and solved using CPLEX 12.6.3.0 on a HP ProLiant server with 4 AMD 2.50 Ghz CPUs, with a total of 64 cores and 256 GB RAM. The reported runtimes are as measured by GAMS (Rosenthal, 2014).

Results from the full models

The optimal investment decisions and resulting costs of the four models are provided in Table 3.6 with respectively non-zero and zero balancing costs.

Regarding the investment decisions, we note that all models in Table 3.6 install approximately the same wind capacity (around 2560 MW) due to the minimum wind penetration constraint. The small differences in wind investments is due to load shedding and wind curtailment.

When comparing representative days and hours in Table 3.6 the main difference is in nuclear investment capacities. Representative hours results in approximately 35% larger nuclear capacities; 1929 MW versus 1417 MW in the deterministic models (HC-8760 and DC-365) and 1841 MW versus 1354 MW in the stochastic models (HS-8760 and DS-365)

Table 3.6: Optimal investment decisions and model runtimes for the different full models. Total costs (TC), investment costs (IC), operating costs (OC), load shedding costs (LSC) and wind curtailment costs (WCC), all in M€, from evaluating the investment decisions in DS-365.

(a) $c_g^+ = 0.05\frac{c_g}{r_g^u}$ and $c_g^- = 0.05\frac{c_g}{r_g^d}$.

Model	wind	coal	gas	nuclear	Runtime (s)	TC	IC	OC	LSC	WCC
DS-365	2562	983	796	1354	30859	990	665	316	3	7
HS-8760	2559	339	939	1841	1829	1020	676	322	4	18
DC-365	2561	935	747	1417	217	990	666	312	4	8
HC-8760	2559	241	902	1929	32	1035	677	326	7	25

(b) $c_g^+ = c_g^- = 0$

Model	wind	coal	gas	nuclear	Runtime (s)	TC	IC	OC	LSC	WCC
DS-365	2561	950	777	1406	24512	981	668	304	3	6
HS-8760	2559	244	937	1937	1162	1019	680	315	3	20
DC-365	2561	935	747	1417	88	981	666	303	4	7
HC-8760	2559	241	902	1929	32	1022	677	316	7	23

including balancing costs. The reason is that ramping needs are ignored and nuclear is inexpensive baseload. For representative days accounting for ramping, nuclear is replaced by coal, the capacity of which is 2-3 times larger than for representative hours. Somewhat surprisingly gas investment capacity is around 20% less for representative days than for representative hours; 747 MW versus 902 MW in the deterministic models and 796 versus 939 MW in the stochastic models. This can be explained by the larger installation of coal that to some extend covers the need for flexibility.

Note that the optimal investment decisions from the deterministic models are the same with non-zero or zero balancing costs while they differ for the stochastic models, e.g. coal investment in HS-8760 is 339 MW with balancing costs and 244 MW without balancing costs. The reason is that in the deterministic models it is never optimal to use the balancing market. Nevertheless, balancing costs do influence the costs of the investment decisions when evaluated in the DS-365 model.

The differences between the deterministic and stochastic models are less than 5%, except when comparing the coal investments for representative hours (HC-8760 and HS-8760) with non-zero balancing costs. These small differences in the case study are in contrast to the example in Section 3.4, for which we observed significant differences between the investments from the stochastic and the deterministic models. Since the example in Section 3.4 is a stylised, illustrative example, this is not surprising. The units of the example are either flexible with respect to ramping and balancing, whereas this is rarely the case in reality. In contrast, some units of the case study such as coal and gas are flexible with respect to both ramping and balancing, with relatively high ramp rates and low balancing costs. Thus, the flexibility needs in a stochastic market clearing are already partly covered by the flexible units installed to cope with variability of demand and wind power production. In fact, the only significant difference between the deterministic and stochastic models is for the representative hours in Table 3.6(a). Here, the HS-8760 model results in 40% higher investment capacities in the flexible coal than the HC-8760 model, that is, 339 MW versus 241 MW. The same does not apply for the results in Table 3.6(b) since the assumption of zero balancing costs produces the less realistic conclusion that nuclear is the best option for balancing. To summarise, the inclusion of stochastic market clearing improves the results for representative hours when balancing costs are non-zero but not much for representative days.

The same tendency is observed for the costs, for which the main differences are between representative days and representative hours and not between the deterministic and stochastic models. The higher nuclear capacities lead to higher investment costs for representative hours than representative days. The higher nuclear capacities, however, also generate higher realised operating costs as gas satisfy the flexibility needs for representative hours whereas coal serves this purpose for representative days. Moreover, wind curtailment costs are very different for representative days and hours because the large inflexible nuclear capacities act as baseload and peaks in wind production must be curtailed. Finally, since the objective functions are based on expected costs, security of supply is only accounted for through expected load shedding penalties. Lower penalties reveal that the stochastic models have slightly higher reliability rates.

Table 3.7: The number of variables and constraints and the runtimes for the four different models with non-zero balancing costs.

Model	# of variables	# of constraints	runtime (s)
DS-365	$13 \cdot 10^5$	$12 \cdot 10^5$	30859
HS-8760	$13 \cdot 10^5$	$7 \cdot 10^5$	1829
DC-365	$2 \cdot 10^5$	$2 \cdot 10^5$	217
HC-8760	$2 \cdot 10^5$	$1 \cdot 10^5$	32

The number of variables and constraints in the different models is specified in Table 3.7. When comparing the deterministic models to the stochastic models with 10 scenarios, we observe that the runtimes increase with a factor between 60 and 200. The number of variables and constraints increase with a factor slightly less than 10, indicating that the computational burden increases more than linearly in the number of variables and constraints. When comparing the representative days and hours, the increase in runtimes is with a factor between 3 and 11 and is due to the additional constraints for representative days.

We conclude this section by comparing the optimal investment decisions, the total costs and the model runtimes of the four full models. The trade-off between runtime and total costs clearly points at DC-365 as the preferred model. Representative hours do not perform as well as representative days and DS-365 does not perform significantly better than DC-365, even with a runtime significantly larger. Hence, when faced with the choice between modelling variability or uncertainty, the inclusion of dynamics is preferable.

Results from the models with aggregated data

We evaluate the performance of the different models for an increasing number of days and hours. For example, we solve each of the problems DC-10, DS-10, HC-240 and HS-240. For each of the optimal solutions to DC-10, DS-10, HC-240 and HS-240, we fix the investment decision and solve the full DS-365. We do the same for higher numbers of days and hours. Figure 3.4 shows the differences between the resulting objective function values and the minimal costs of DS-365 in percentage.

In both Figure 3.4(a) and Figure 3.4(b), we observe that with representative days the cost differences approach zero when the number of days increases. As expected, the in-

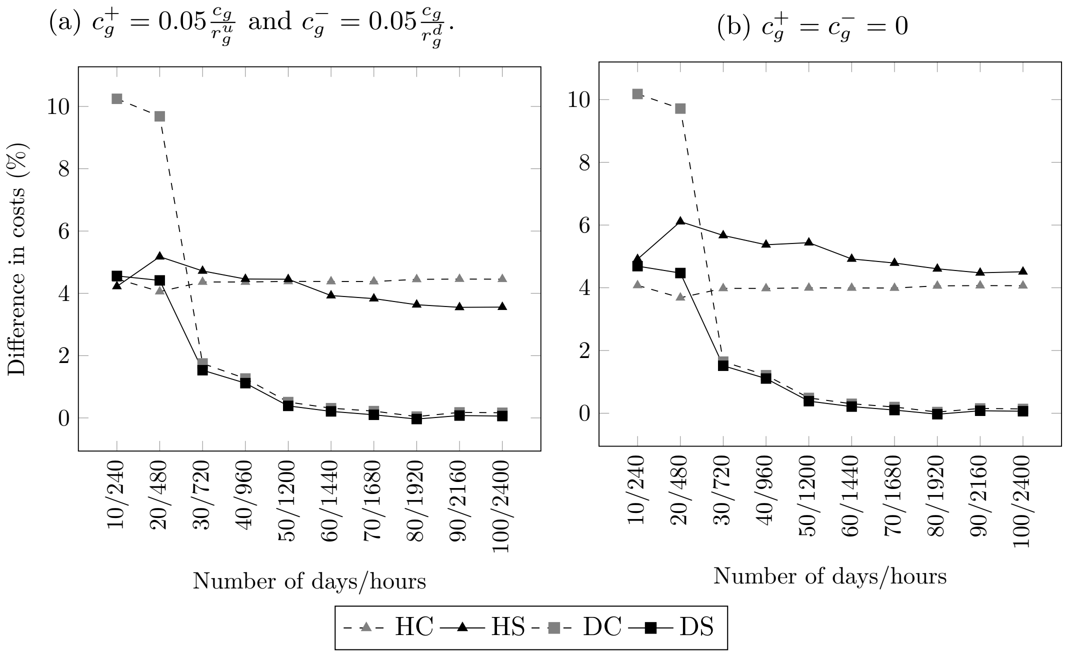


Figure 3.4: Total costs differences between the models with aggregated data and the full model. The x-axis refers to the number of representative days and hours.

clusion of more days results in investment decisions closer to those of the full model. The same is not observed with representative hours. The cost differences do not improve for an increasing number of hours, and thus, even for the highest number of hours included, 2400, the level of detail in representative hours is insufficient. When comparing the representative days and hours, the former outperform the latter when including 30 days or more. In fact, we confirm that the effect of taking short-term variability into account is crucial, even for a limited number of days.

When comparing the deterministic and stochastic models, we note that for 30 days or more, the models produce very similar cost differences from the full model. When including 30 representative days, the costs difference is already less than 2%, indicating that 30 representative days offset the effects of uncertainty in this specific case study. We, therefore, stipulate that you can ignore uncertainty by adding a sufficient number of representative days, which is computationally much less expensive than doing stochastic optimisation.

Analysing the differences between Figure 3.4(a) and Figure 3.4(b), the main difference is that the stochastic model with representative hours performs better in Figure 3.4(a) than in Figure 3.4(b). This is because the non-zero balancing costs in Figure 3.4(a) incentivise investment in more flexible units which in turn then reduce the difference in costs from the full model. Observe also that the the stochastic model with representative hours performs slightly worse than the deterministic counterpart in Figure 3.4(b) which seems counter-intuitive. In the stochastic model with zero balancing costs, the inflexible nuclear unit can be used as a balancing unit at zero costs because there are no ramping constraints. Thus, the stochastic model invests more in nuclear than the deterministic, which is more costly when evaluated in the full model.

Figure 3.5 shows the runtimes of each model plotted against the number of hours or days. Note that the y-axis is logarithmic. The results are very similar with zero and non-zero balancing cost. In both cases, the stochastic models are by far the most com-

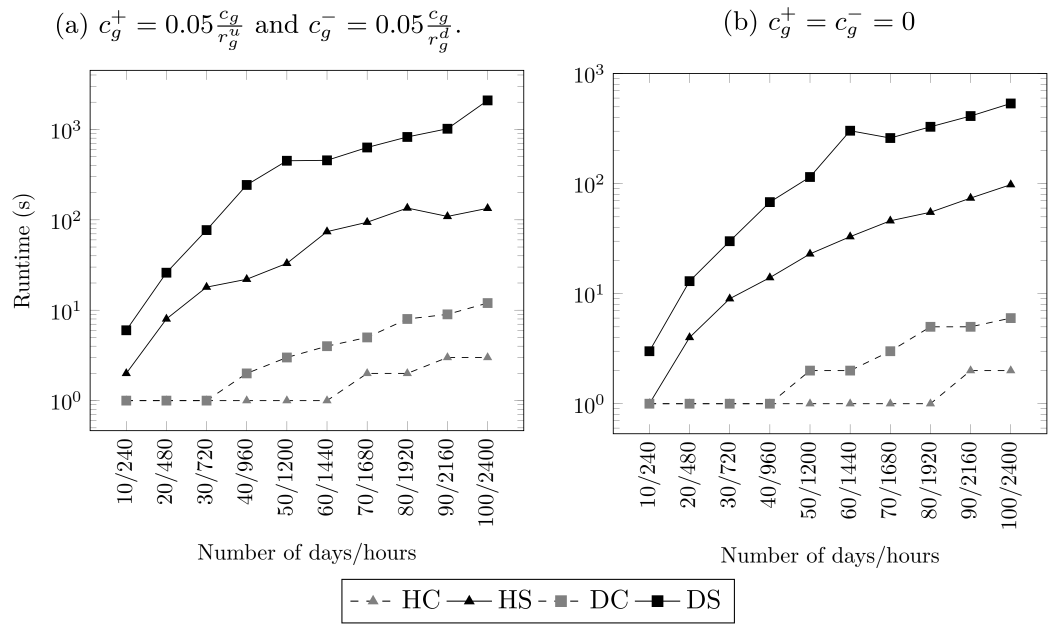


Figure 3.5: Runtimes as a function of number of days/hours for all models. Note the y-axis is logarithmic and that runtimes have a lower bound of 1 second.

putationally heavy. The reason is that the stochastic models are larger by an order of magnitude of 50-150 with representative days and 25-50 with representative hours.

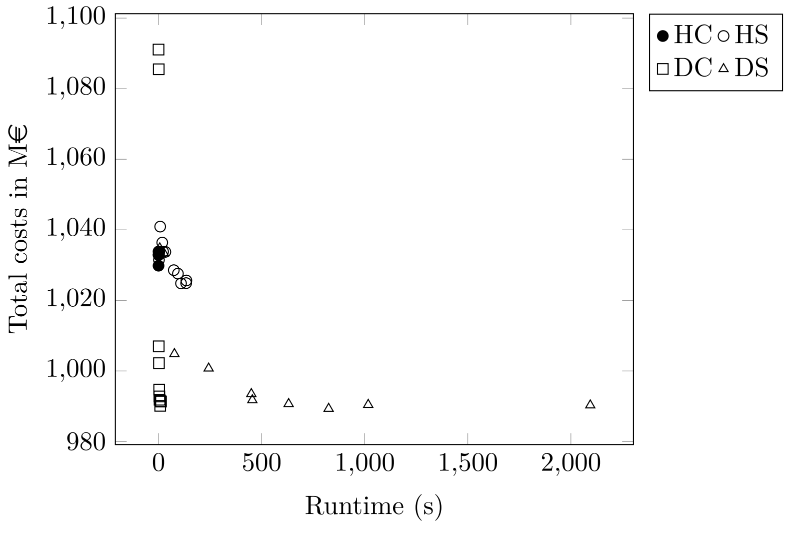


Figure 3.6: Runtime versus total costs for all models (except DS-365).

To illustrate the trade-off between the quality of the investment decisions and the computational effort, Figure 3.6 plots the runtimes against the total costs for all models and all days/hours. With hourly representation, the points are all close, with small relative differences in both runtime and total costs. The stochastic models, however, always have lower total costs and higher runtime than the deterministic. With daily representation, all models have relative low runtime, whereas the best deterministic models also have relatively low total costs. The stochastic models have the lowest total costs but only for models with a very high runtime.

To summarise the findings of the case study, the DC-30 model yields investment decisions with less than a 2% difference in total costs to the full DS-365 model. Furthermore, the computational burden of the DC-30 model is far less than DS-365, with runtimes under 1 second for the DC-30 model and over 30,000 seconds for the DS-365 model when considering non-zero balancing costs.

3.6 Conclusion

With higher shares of renewable energy sources in many power systems, it is increasingly important to account for short-term variability and uncertainty in long-term power planning. Nevertheless, this often requires a level of techno-economical detail in modelling that significantly affects computational tractability. In this paper, we compare different approaches to represent variability and uncertainty in a model, while reducing runtime. We use an example to illustrate the effects of variability and uncertainty, whereas a Danish case study provides more realistic results.

Our example shows that accounting for short-term variability through ramping constraints and/or uncertainty via balancing costs has significant impact on the quality of investment decisions. In our more realistic case study, however, the inclusion of representative days and ramping constraints has the most significant effect, both regarding the quality of the solution and the computational burden of solving the model. In particular, we observe that the impact of short-term uncertainty is less important as the number of representative days increase.

Our model can be extended in various directions. For computational reasons, we capture inter-temporal restrictions through ramping constraints only. Our results may therefore underestimate the importance of including short-term techno-economical details in a long-term power planning problem. At the expense of longer runtimes, however, the model can be extended to account for unit commitment. Our model can likewise be extended to include network and transmission expansion. Network expansion may provide further system flexibility, whereas transmission constraints may impose restrictions on flexibility in generation. This trade-off may be subject of future research. Moreover, the market structure with perfect competition could be further investigated, from the perspective of both investors and policy makers. Allowing for market power, the model may become a mathematical programming problem with equilibrium constraints, for which computational tractability is of an even higher concern.

Chapter 4

A Parametric Programming Approach to Bilevel Optimization with lower-level Variables in the upper Level

H. BYLLING, S. GABRIEL AND T. BOOMSMA

Chapter Abstract

This paper examines linearly constrained bilevel programming problems in which the upper-level objective function depends on both the lower-level primal and dual optimal solutions. We parametrize the lower-level solutions and thereby the upper-level objective function by the upper-level variables and argue that it may be non-convex and even discontinuous. However, when the upper-level objective is affine in the lower-level primal optimal solution, the parametric function is piece-wise affine. We show how this property facilitates the application of parametric programming and demonstrate how the approach allows for decomposition of a separable lower-level problem. When the upper-level objective is bilinear in the lower-level primal and dual optimal solutions, we also provide an exact linearization method that reduces to a single-level mixed-integer linear programming (MILP) formulation of the bilevel problem. We assess the performance of the parametric programming approach on two case studies of strategic investment in electricity markets and benchmark against state-of-the-art MILP and non-linear solution methods for bilevel optimization problems. Preliminary results indicate substantial computational advantages over several standard solvers, especially when the lower-level problem separates into a large number of subproblems. Furthermore, we show that the parametric programming approach succeeds in solving problems to global optimality for which standard methods can fail.

4.1 Introduction

Despite its complexity, bilevel programming has become a well-studied subject in optimization due to its many application areas, including economics and engineering. In economics, for example, bilevel programming is used for modeling a Stackelberg game

(Stackelberg, 1934), in which a leader makes optimal decisions while anticipating the reactions of one or more followers. For reviews of applications, see Dempe et al. (2015), Saharidis et al. (2013), Colson et al. (2007), Colson et al. (2005b) and Bard (1998).

In this paper, we specifically consider linearly constrained bilevel programming problems with lower-level primal and dual optimal solutions in the upper-level objective function (abbreviated BPP-Ds). The structure of BPP-Ds arises in many applications. Important examples are economic decision problems in which a strategic agent maximizes profit at an upper level, while anticipating market-clearing at a lower level. The upper-level objective function includes a bilinear product of lower-level primal and dual optimal solutions such as a revenue. For example, the prices are the market-clearing constraint dual prices and the quantity the primal solutions. Often, the lower-level problem represents a number of separate market-clearing conditions, e.g., for representative hours of a day or days of a year. Besides leader-follower games in economics, some other well-known problems for bilevel optimization or the related mathematical program with equilibrium constraints (MPEC) include: production and marketing models, robotics, continuous transportation design and facility location and production, see Luo et al. (1996), Bard (1998) and Dempe (2018). More specific to energy economics, examples of upper-level strategic decisions include generation capacities, as by Conejo et al. (2016), Kazempour et al. (2011), Baringo and Conejo (2012a) and Koschker and Möst (2016), transmission capacitites, cf. Garcés et al. (2009), or price-offering decisions as by Ruiz and Conejo (2009). Furthermore, bilinear products of primal and dual variables occur in many complementarity models and mathematical problems with equilibrium constraints (MPECs), cf. Ehrenmann and Smeers (2011), Gabriel et al. (2006), Chen et al. (2006), Ruiz and Conejo (2009) and Gabriel et al. (2010).

In this paper, we provide a formal analysis of the BPP-D with a view towards parametric programming. In particular, we parametrize the lower-level optimal solutions and thereby the upper-level objective function by the upper-level variables only and discuss why it may be non-convex and even discontinuous. When the upper-level objective is affine in the lower-level primal optimal solution, and hence, also when the upper-level objective is bilinear in the primal and dual optimal solutions, we show that the parametric function is likewise affine on its critical regions, i.e., the regions of upper-level solutions for which the same lower-level basis is optimal.

As our main contribution, we demonstrate how the upper-level objective function is piece-wise affine which facilitates the application of parametric programming. In the spirit of Gal (1995) and Faísca et al. (2007), we iteratively determine neighboring critical regions and thereby completely specify the parametric function. As a result, we reduce the bilevel problem to solving a number of single-level linear programming (LP) problems over each region. In contrast to state-of-the-art non-linear solution methods, this approach guarantees global optimality. Moreover, we extend the parametric programming approach to allow for decomposition when the lower level separates into a number of subproblems for a given upper-level solution.

When the upper-level objective is bilinear in the lower-level primal and dual optimal solutions, we also provide sufficient conditions for exact linearization, using the optimality conditions and strong duality of the lower-level problem. This produces a single-level mathematical program with equilibrium constraints and linear objective function, which again warrants a mixed-integer linear programming (MILP) reformulation under certain assumptions. The MPEC and MILP can be solved by standard software for non-linear or mixed-integer programs, respectively. To the best of our knowledge, the existing literature does not cover such conditions for linearization of the general BPP-D.

We illustrate the characteristics of the parametric function on a stylized problem of investment in generation capacity. For two more detailed case studies of investment in generation and transmission capacity, respectively, we assess the performance of the parametric programming approach and benchmark against the MILP (when reformulation is possible) and standard non-linear solution solvers to bilevel optimization problems.

4.2 Literature review

Many reformulations and algorithms have been suggested for the highly challenging class of bilevel optimization problems (and the closely related class of mathematical programming problems with equilibrium constraints (MPECs)) that are generally NP-hard, cf. Ben-Ayed and Blair (1990). When the lower-level problem is convex, one reformulation is obtained by replacing this by its Karush-Kuhn-Tucker (KKT) optimality conditions, see Dempe (2018) and Mirrlees (1999). As a result, the bilevel problem becomes an MPEC, which is single-level but non-convex. Such MPECs can be solved using an equivalent mixed-integer linear program (MILP), descent algorithms or penalty function methods, see Colson et al. (2005b). Another reformulation is obtained via the optimal value function, cf. Outrata (1988). For linearly constrained bilevel optimization problems, some algorithms that are particularly relevant for the present paper. These include the vertex enumeration and descent method by Han et al. (2000) and enumeration of the basis matrices of the lower-level problem, which is of polynomial time according to Liu and Spencer (1995) and Dempe (2018). For reviews of solution methods to bilevel optimization we refer to Vicente and Calamai (1994) and Dempe (2018) and for approaches using descent methods or penalty functions, see also Colson et al. (2005b). Due to the non-linearity of the upper-level objective function, however, these methods may not immediately apply to the general BPP-D and/or do not guarantee global optimality.

Closest to our work is the global optimization approach of Faísca et al. (2007) for linearly and quadratically constrained bilevel optimization problems. As the present paper, the method relies on parametric programming methods from e.g. Gal (1995) or Dua et al. (2002). Whereas our approach can handle bilinear terms of lower-level primal and dual variables in the upper-level objective, however, Faísca et al. (2007) only consider lower-level primal variables in a linear or quadratic upper-level objective function. Furthermore, no decomposition of the BPP-D is offered.

In the current literature, a bilinear objective term which is used in the current paper has been dealt with in various ways. Certain problem structures allow for linearization using optimality conditions and duality theory. This strategy has been used in many applications, for examples in energy markets, see Gabriel et al. (2013) and Conejo et al. (2016). For the specific problem of Hobbs et al. (2000), the complementarity constraints of the lower-level problem allow the authors to replace the upper-level objective function by a concave quadratic function. Other approaches use integer variables and/or logical constraints to approximate the bilinear product, cf. Koschker and Möst (2016) and Gabriel et al. (2010). Alternatively, bilinear products can be approximated using Schur's decomposition and special ordered sets of variables, see Gabriel et al. (2006). Since exact linearization of the bilinear term has been applied only to specific problems, the present paper suggests sufficient conditions that allow exact linearization in more general instances.

For many applications of bilievel optimization, the lower-level problem separates into a number of subproblems for a given upper-level solution. In such cases, decomposition could provide computational advantages. However, the parametric functions involved, be it an upper-level objective term or a lower-level value function, are non-convex. Nevertheless, Kazempour and Conejo (2012) present a Benders' decomposition approach, for which the lower-level value function is argued to be *sufficiently* convex. The single-level MPEC and MILP reformulations are generally not separable and decomposition is likewise difficult, as the subproblems are also non-convex. In contrast, the parametric programming approach is easily extended to allow for decomposition.

The remainder of this paper is organized as follows: In Section 4.3, we define and analyze the BPP-D. Section 4.4 presents the parametric programming approach, whereas 4.5 contains reformulation and linearization of the BPP-D. Examples and numerical results are provided in Section 4.6 and the conclusion is given in Section 4.7.

4.3 Bilevel programming with lower-level primal and dual information in the upper level

We consider a linearly constrained bilevel programming problem with lower-level primal and dual optimal solutions in the upper-level objective function (BPP-D). The problem is defined as follows:

$$\min c^T x + f(y^*, \lambda^*) \tag{4.1a}$$
$$\text{s.t. } Ax = b \tag{4.1b}$$
$$x \geq 0 \tag{4.1c}$$
$$y^* \in \text{argmin}\{p^T y \tag{4.1d}$$
$$\text{s.t.} \quad Cy = Dx + e \tag{4.1e}$$
$$y \geq 0\} \tag{4.1f}$$
$$\lambda^* \in \text{argmax}\{\lambda^T (Dx + e) \tag{4.1g}$$
$$\text{s.t.} \quad C^T \lambda \leq p\} \tag{4.1h}$$

The upper-level objective function (4.1a) involves a linear function of the upper-level variables, $x \in \mathbb{R}^n$, and a function $f : \mathbb{R}^l \times \mathbb{R}^k \to \mathbb{R}$ of lower-level primal and dual optimal solutions, $y^* \in \mathbb{R}^l$ and $\lambda^* \in \mathbb{R}^k$, respectively. For simplicity, we assume that the upper-level constraints (4.1b) depend only on x and not on y^* and λ^*. This assumption is common in the bilevel programming literature, as such dependence could result in a disjoint or empty feasible region of the BPP-D, cf. Colson et al. (2005b) and Shi et al. (2005b).

The linear programming (LP) problem (4.1d)-(4.1f) is the lower-level primal problem and (4.1g)-(4.1h) is the corresponding dual problem. The upper-level variables x are fixed parameters in the right-hand side of the lower-level primal problem and the objective function of the dual problem. Hence, y^* and λ^* are parameterized by x and so is the function $f(y^*, \lambda^*)$.

The upper-level cost vector and the vector of right-hand sides have dimensions $c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$, and the lower-level cost vector and right-hand side have dimensions $p \in \mathbb{R}^l$ and $e \in \mathbb{R}^k$. All constraints are linear. Accordingly, the constraint matrices have the dimensions $A \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^{k \times l}$ and $D \in \mathbb{R}^{k \times n}$.

We define the feasibility sets

$$S = \{x \in \mathbb{R}^n \,|\, Ax = b, x \geq 0\},$$

and for a fixed solution $x \in \mathbb{R}^n$,

$$P(x) = \{y \in \mathbb{R}^l \,|\, Cy = Dx + e,\ y \geq 0\},$$

of the upper-level and lower-level problems, respectively. Moreover, we let

$$S^* = S \cap \{x \in \mathbb{R}^n \,|\, \exists y \in P(x)\},$$

i.e., the set of solutions that are feasible in the upper-level problem and produces a feasible lower-level problem. Clearly, P and S are (closed and convex) polyhedra. The set S^* is likewise a polyhedron, see Gal (1995).

The following assumptions ensure feasibility and boundedness of the BPP-D.

Assumption 4.3.1. *Assume that $\{\lambda \in \mathbb{R}^l \,|\, C^T\lambda \leq p\}$ is non-empty and S^* is non-empty and bounded.*

Moreover, for fixed $x \in S^*$, the assumption guarantees primal and dual feasibility and thereby also optimality of the lower-level problem.

With these assumptions, we can define $y^*(x)$ and $\lambda^*(x)$ to be lower-level primal and dual optimal solutions for fixed $x \in S^*$. Hence, the parameterized solutions $y^* := y^*(x)$ and $\lambda^* := \lambda^*(x)$ in (4.1a)-(4.1h) are well-defined. For non-unique primal and dual optimal solutions to (4.1d)-(4.1f) and (4.1g)-(4.1h), $y^*(x)$ and $\lambda^*(x)$ may be chosen as so-called $optimistic$ or $pessimistic$ solutions to a bilevel programming problem, as defined by Colson et al. (2005b). The optimistic and pessimistic solutions are the best and worst lower-level solutions, respectively, with respect to the upper-level objective function. If these are also non-unique, one of them may simply be chosen.

If choosing only one optimal solution to the lower-level problem, we can likewise define the function $F : \mathbb{R}^n \to \mathbb{R}$ such that

$$F(x) = f(y^*(x), \lambda^*(x)),\ x \in S^*.$$

With this parametrization, the BPP-D becomes

$$\min c^T x + F(x) \tag{4.2a}$$
$$\text{s.t. } x \in S^* \tag{4.2b}$$

This is a single-level linearly constrained optimization problem with regard to x, albeit $F(x)$ is generally not known in closed form. However, we aim to characterize the function F in certain cases that allow for computational tractability of the BPP-D.

We start by analyzing the lower-level LP in more detail. Let therefore $x \in S^*$ be given. Assume that the rank of the matrix C is k. Let $B \in \mathbb{R}^{k\times k}$ be a basis for $P(x)$, i.e., a non-singular $k \times k$ submatrix of C. The primal basic solution is $y(x) = (y_B(x)^T, y_N(x)^T)$, where $y_B(x) = B^{-1}(Dx + e)$ and $y_N(x) = 0$ are the subvectors with k basic and $l - k$ non-basic variables, respectively. The complementary dual basic solution is $\lambda(x) = (B^{-1})^T p_B$, where p_B is the subvector of p with entries corresponding to the k basic variables.

For the lower-level LP, a basic solution is either degenerate or non-degenerate. For convenience, we include the definition of degeneracy below.

Definition 4.3.2. (Bertsimas and Tsitsiklis, 1997) Let $B \in \mathbb{R}^{k\times k}$ be a basis for $P(x)$. A basic solution $y(x) = (B^{-1}(Dx+e), 0)^T$ is *degenerate* if more than $l-k$ of the l variables are zero.

Consequently, a basic solution is non-degenerate if k variables are non-zero and $l-k$ variables are zero. For a non-degenerate solution, B is uniquely defined. For a degenerate primal basic solution, however, B is no longer uniquely defined and the complementary dual basic solution is non-unique.

The following defines critical regions.

Definition 4.3.3. (Gal, 1995) Let $B \in \mathbb{R}^{k\times k}$ be a basis for $P(x)$. The critical region corresponding to B is $\Lambda_B = \{x \in S^* \mid B^{-1}(Dx + e) \geq 0\}$.

The critical region corresponding to B is the set of $x \in S^*$ for which (4.1d)-(4.1f) has a basic solution $y(x) = (B^{-1}(Dx + e), 0)^T$ that satisfies non-negativity $y(x) \geq 0$. Since the constraints $Cy(x) = Dx + e$ are satisfied by construction, this implies that $y(x)$ is primal feasible. The complementary dual basic solution $\lambda(x)$ likewise satisfies $C^T\lambda(x) \leq p$ by construction and is therefore dual feasible. As a result, the basis B is optimal for all $x \in \Lambda_B$. Note that like S^*, Λ_B is a polyhedron. Note also that on the boundaries of the critical regions, the basis may not uniquely defined, or equivalently, the lower-level primal problem may be degnerate and the dual problem may have multiple optimal solutions.

Now, let $\mathcal{B}$ be the set of all optimal bases to (4.1d)-(4.1f) for $x \in S^*$. Evidently,

$$S^* = \bigcup_{B\in\mathcal{B}} \Lambda_B,$$

i.e., S^* is the union of all critical regions.

We proceed to determine the gradient of F on the interior of the critical regions and address the special case in which it is constant. For a given basis B, we denote the interior of Λ_B by Λ_B° .

Proposition 4.3.4. *Let $\Lambda_B \subseteq S^*$ be the critical region corresponding to the basis B. Then, $F(x)$ is differentiable on Λ_B° with gradient*

$$\nabla F(x) = (B^{-1}D)^T \frac{\partial f}{\partial y_B}(y^*(x), \lambda^*(x)),\ x \in \Lambda_B^{\circ}.$$

where $\partial f/\partial y_B$ is the vector of the k derivatives of f with respect to the basic variables y_B. If $f(\cdot, \lambda)$ is an affine function for any given λ, then $F(\cdot)$ is an affine function on Λ_B°.

Proof. Let $x \in \Lambda_B^{\circ}$. The primal optimal basic solution is given as $y^*(x) = (y_B^*(x)^T, y_N^*(x)^T) = (B^{-1}(Dx + e), 0)^T$, and hence, $y_N^*(x)$ does not depend on x at optimality. The complementary dual basic solution is $\lambda^*(x) = (B^{-1})^T p_B$, and so, $\lambda^*(x)$ likewise does not depend on x at optimality.

Using the chain rule to find the gradient of F, we obtain

$$\begin{aligned}\frac{\partial F}{\partial x_i}(x) &= \sum_{j=1}^{k} \frac{\partial f}{\partial y_j^*}(y^*, \lambda^*)\frac{\partial y_j^*}{\partial x_i}(x) + \sum_{j=1}^{k} \frac{\partial f}{\partial \lambda_j^*}(y^*, \lambda^*)\frac{\partial \lambda_j^*}{\partial x_i}(x)\\ &= \sum_{j\in B} \frac{\partial f}{\partial y_j^*}(y^*, \lambda^*)(B^{-1}D)_{ji},\ i = 1, \dots, n,\end{aligned}$$

where $y_i^*, i \in B$ refer to the basic variables (with a slight abuse of notation).

If f is affine, then $\partial f / \partial y_j^*, j = 1, \ldots, k$ are constant. As a result, $\partial F / \partial x_i, i = 1, \ldots, n$ are constants, and thus, F is affine. □

As a result of Proposition 4.3.4, F is a piece-wise affine (but not necessarily continuous) on S^*. To see that F is not necessarily continuous, recall that on the boundaries of the critical regions, the lower-level dual problem may have multiple optimal solutions. Moreover, within a critical region, the dual solution is constant. If the chosen dual solutions (e.g., optimistic or pessimistic) to two neighboring critical regions are different, a discontinuous jump occurs in the parametric function F.

The piece-wise affinity facilitates the application of parametric programming. In particular, F is completely specified by its gradient on the interiors of the critical regions and thereby easily on its entire domain. For the remainder of this paper, we therefore make the following assumption, unless otherwise specified.

Assumption 4.3.5. *Assume that for any given λ, $f(\cdot, \lambda)$ is an affine function.*

An important special case of a BPP-D is when $f(\cdot, \cdot)$ is a bilinear function on the general form

$$f(y, \lambda) = d^T y + h^T \lambda + \lambda^T H y,$$

with $d \in \mathbb{R}^l, h \in \mathbb{R}^k$ and $H \in \mathbb{R}^{k \times l}$. For a given basis B and $x \in \Lambda_B^o$, by Proposition 4.3.4, the gradient of F is

$$\nabla F(x) = (B^{-1} D)^T (d_B + H_B^T p_B^T B^{-1}),$$

where d_B is the subvector of d with entries corresponding to the basis B and H_B is the submatrix of H with columns corresponding to the basis B.

4.4 Parametric programming

To solve the BPP-D, we utilize (4.2) and propose a parametric programming approach. The approach takes advantage of the critical regions and Proposition 4.3.4, characterizing the parametric function F on these.

The first step to solve the BPP-D is to determine the critical regions. To do this, we follow Gal (1995). We introduce the so-called neighboring critical regions of the lower-level primal problem (4.1d)-(4.1f).

Definition 4.4.1. (Gal, 1995) Two critical regions, Λ_1, Λ_2, are *neighbors* if the following holds for their corresponding bases B_1, B_2:

1. There exists an $x \in S^*$ for which B_1 and B_2 both are optimal bases to (4.1d)-(4.1f).
2. It is possible to pass from B_1 to B_2 in one iteration of the dual simplex method.

Note that a neighboring basis is defined in terms of an iteration of the dual simplex method, but *not* the primal simplex method.

Instead of determining all critical regions corresponding to optimal bases of (4.1d)-(4.1f), by (Gal, 1995) (Theorem IV-6 and IV-7), it is sufficient to iterate through neighbors. More precisely, let $\mathcal{B}^*$ be the set of optimal bases, obtained by starting from some initial basis and iteratively finding all neighbors to the current basis. Then,

$$S^* = \bigcup_{B \in \mathcal{B}^*} \Lambda_B.$$

Furthermore, when a neighboring basis is defined in terms of an iteration of the dual simplex method, $\Lambda^o_{B_1} \cap \Lambda^o_{B_2} = \emptyset$ for $B_1, B_2 \in \mathcal{B}^*$, i.e., neighbors overlap only at their boundaries, cf. Gal (1995) (Theorem IV-5). Although it could sometimes be possible to pass from one basis to another by an iteration of the primal simplex method, the interiors of the critical regions overlap in this case (Gal, 1995).

We now describe how to find a neighbor. Consider a basis B and the corresponding critical region Λ_B. Let $x \in \Lambda_B$ and consider the primal basic solution $y := y(x) = (B^{-1}(Dx+e), 0)^T$. To determine if a neighbor exists, examine the potential variables to leave the basis. By dual simplex, y_i can only leave the basis if

$$(B^{-1}C)_{ij} < 0 \text{ for at least one non-basic variable } y_j, \tag{4.3}$$

i.e., the value of the basic variable y_i decreases as a y_j enters the basis and increases. Moreover, y_i can only leave the basis if there exist an $x^* \in S^*$ such that

$$(B^{-1}(Dx^* + e))_i = 0, \tag{4.4}$$

i.e., the value of the objective function does not decrease, and thus, the current and new bases are both optimal. To find such an x^*, we solve the LP problem

$$\min\, (B^{-1}(Dx+e))_i \tag{4.5a}$$
$$\text{s.t. } B^{-1}(Dx+e) \geq 0 \tag{4.5b}$$
$$Ax = b \tag{4.5c}$$
$$x \geq 0 \tag{4.5d}$$

If the optimal solution, x^*, to (4.5) has an optimal value of zero, then Λ_B has a neighbor. The neighbor is determined by replacing x by x^* in (4.1d)-(4.1f) and carrying out an iteration of the dual simplex method, letting the leaving variable be y_i and the entering variable y_j be determined by the minimum ratio test. The new optimal basic solution to the lower-level problem is clearly degenerate with two different bases corresponding to the neighboring critical regions.

On the basis of the above, we state the procedure for iteratively finding all neighbors.

Algorithm 4.4.2. Parametric Programming

Step 0 (initialization) Set $h := 0$. Given an initial solution x_0, let $x := x_0$ in (4.1d)-(4.1f) and solve the problem. Store an optimal basis B_0 and determine $F(x_0), \nabla F_{B_0}(x_0)$, Λ_{B_0}. Set $\mathcal{B} := \{B_0\}$.

Step 1 (iteration h) If $\mathcal{B} = \emptyset$, then stop. Otherwise, set $h := h+1$, select $B_h \in \mathcal{B}$ and set $\mathcal{B} := \mathcal{B} \setminus \{B_h\}$.

Step 2 (determine leaving variable) Let $B := B_h$. Select a basic i that satisfies (4.3), solve the problem (4.5), let x_{hi} be an optimal solution and determine if a neighbor exists. If not, select another basic i. If all basic variables have been considered, return to Step 1.

Step 3 (determine entering variable) Let $x := x_{hi}$ in (4.1d)-(4.1f) and carry out an iteration of the dual simplex method. Store a neighboring basis B_j and determine $F(x_{hi}), \nabla F_{B_j}(x_{hi}), \Lambda_{B_j}$. Set $\mathcal{B} := \mathcal{B} \cup \{B_j\}$. Return to Step 2.

In Algorithm 4.4.2, $\mathcal{B}$ is a pool of bases to be investigated. In Step 0, the algorithm starts with some initial upper-level feasible solution, stores an optimal lower-level basis and determines the critical region and characteristics of the parametric function F. Since the function is affine, it is sufficient to evaluate it and its gradient at the initial solution. If in Step 1 there are no more bases in the pool $\mathcal{B}$, the algorithm terminates. Otherwise, a basis B_h is selected. In Steps 2 and 3, all neighbors to this critical region are found by iterating through potential variables to leave the basis and for each such variable, determining the variables to enter by the minimum ratio test. If a neighbor is determined by dual simplex, the algorithm stores an optimal basis, determines the critical region and evaluates the parametric function. Only bases that have not yet been investigated in previous iterations need to be considered. These new bases are added to the pool. Algorithm 4.4.2 terminates when all neighbouring critical regions of S^* have been found. This finite termination is proven by Manas and Nedoma (1968).

To start the procedure, we assume an initial upper-level feasible solution is available. If no apparent initial feasible solution exists, then one can be found by solving (4.1) while omitting (4.1d) and (4.1g), i.e., assuming that all variables are determined by a single-level optimization problem. By further omitting any non-linear terms of f, the approximate problem becomes an LP. With the upper-level variables fixed to their optimal values, the lower-level problem is likewise an LP. The combination of the solutions to the upper-level approximation and the lower-level problems is feasible in (4.1).

Having determined all neighboring critical regions of S^*, the next step is to solve the restrictions of (4.2) to each region. For critical region Λ_B and given $x_B \in \Lambda_B$, the restricted problem is

$$\min c^T x + F_B(x_B) + \nabla F_B(x_B)^T(x - x_B) \tag{4.6a}$$

$$\text{s.t. } x \in \Lambda_B \tag{4.6b}$$

This is an LP, as $\Lambda_B = \{x \in S^* | B^{-1}(Dx + e) \geq 0)\}$ is a polyhedron. Thus, there exists an optimal basic solution, i.e., in a vertex of the critical region. Recall, however, that in a vertex, the lower-level primal problem may be degenerate and the dual problem may have multiple optimal solutions. To obtain a global optimal solution to the BPP-D, we solve the LP problem (4.6) for all neigboring critical regions and compare their optimal solutions.

The parametric programming approach relies on finding critical regions of S^*. Finding all regions can be computationally demanding if the number of regions is high. An upper bound is the number of distinct bases in the lower-level problem, given by the binomial coefficient

$$\binom{l}{k} = \frac{l!}{k!(l-k)!} \,.$$

Fortunately, it is sufficient to determine critical regions by iterating through all neighbors, the interiors of which do not overlap. Thus, the number of bases to investigate may be much lower.

The solution approach is particularly useful in two respects: Since the parametric programming method determines $F(x)$ for every $x \in S^*$, near optimal solutions are automatically provided ex post. For other solution methods to non-convex optimization problems such as the BPP-D, this postoptimal information is difficult to obtain if at all available. More importantly, the method allows for decomposition of a BPP-D for which the lower-level separates into a number of subproblems for a given upper-level solution.

Decomposition is described in Section 4.4.1. We propose two variants of the decomposition approach: An exact solution method that finds the global optimum and a heuristic with some computational advantages.

4.4.1 Decomposition

Consider a BPP-D of the form

$$\min c^T x + \sum_{t=1}^{s} f_t(y_t^*, \lambda_t^*) \tag{4.7a}$$

$$\text{s.t. } Ax = b \tag{4.7b}$$

$$x \geq 0 \tag{4.7c}$$

$$y_t^* \in \text{argmin}\{p_t^T y_t \tag{4.7d}$$

$$\text{s.t.} \quad C_t y_t = D_t x + e_t \tag{4.7e}$$

$$y_t \geq 0\} \quad \forall t \tag{4.7f}$$

$$\lambda_t^* \in \text{argmax}\{\lambda_t^T (D_t x + e_t) \tag{4.7g}$$

$$\text{s.t.} \quad C_t^T \lambda_t \leq p_t\} \quad \forall t \tag{4.7h}$$

Note that for fixed $x \in S^*$, the lower level separates into a number of LP problems indexed by $t = 1, \ldots, s$. Moreover, the upper-level objective function involves a sum of functions $f_t : \mathbb{R}^{l_t} \times \mathbb{R}^{k_t} \to \mathbb{R}$ of lower-level primal and dual optimal solutions $y_t^* \in \mathbb{R}^{l_t}$ and $\lambda_t^* \in \mathbb{R}^{k_t}$. The lower-level cost vector and right-hand side have dimensions $p_t \in \mathbb{R}^{l_t}$ and $e_t \in \mathbb{R}^{k_t}$ and the constraint matrices $C_t \in \mathbb{R}^{k_t \times l_t}$ and $D_t \in \mathbb{R}^{k_t \times n}$. Thus, the vectors and matrices of the BPP-D are partitioned into blocks such that $p = (p_1^T, p_2^T, \ldots, p_s^T)$, $e = (e_1^T, e_2^T, \ldots, e_s^T)$ and

$$C = \begin{pmatrix} C_1 & 0 & \ldots & 0 \\ 0 & C_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \\ 0 & 0 & & C_s \end{pmatrix}, \; D = \begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ D_s \end{pmatrix}.$$

For fixed $x \in \mathbb{R}^n$, we define the feasibility sets

$$P_t(x) = \{y \in \mathbb{R}^{l_t} \mid Cy = D_t x + e_t, \; y \geq 0\},$$

and let

$$S_t^* = S \cap \{x \in \mathbb{R}^n \mid \exists y \in P_t(x)\}.$$

We further define the parametric function as the sum

$$F(x) = \sum_{t=1}^{s} F_t(x),$$

where the function $F_t : \mathbb{R}^n \to \mathbb{R}$ is such that

$$F_t(x) = f_t(y_t^*(x), \lambda_t^*(x)), \; x \in S^*.$$

Finally, we let $x \in S^*$ be given and let $B_t \in \mathbb{R}^{k_t \times k_t}$ be a basis for $P_t(x)$. The critical region corresponding to B_t is then defined as $\Lambda_{B_t} = \{x \in S^* \mid B_t^{-1}(D_t x + e_t) \geq 0)\}$.

Let $\mathcal{B}_t^*$ be the set of all optimal bases to the lower-level subproblem indexed by t, obtained by starting from some initial basis and iteratively finding new neighbors to the current basis by Algorithm 4.4.2. By Gal (1995),

$$S_t^* = \bigcup_{B_t \in \mathcal{B}_t^*} \Lambda_{B_t}.$$

In solving the BPP-D of the form (4.7), the lower-level subproblems can be handled separately. In particular, the lower-level subproblem indexed by t is processed by Algorithm 4.4.2, with the result being a set of critical regions. By such decomposition of the problem, the algorithm processes a higher number of smaller problems. To speed up computations, these smaller problems can be processed in parallel.

Proposition 4.4.3 shows that it is indeed sufficient to determine critical regions by iterating through neighbors, for one lower-level subproblem at a time.

Proposition 4.4.3. *For a BPP-D of the form* (4.7)*, the following holds*

$$S^* = \bigcap_{t=1}^{s} S_t^* = \bigcap_{t=1}^{s} \Big(\bigcup_{B_t \in \mathcal{B}_t^*} \Lambda_{B_t} \Big) = \bigcup_{(B_1, B_2, \dots) \in \mathcal{B}_1^* \times \mathcal{B}_2^* \times \cdots \times \mathcal{B}_s^*} \Big(\bigcap_{t=1}^{s} \Lambda_{B_t} \Big).$$

By Proposition 4.4.3, the BPP-D can be handled by solving restrictions of (4.7). For critical regions $\Lambda_{B_1}, \Lambda_{B_2}, \dots, \Lambda_{B_s}$, the restricted problem is

$$\begin{aligned} &\min c^T x + \sum_{t=1}^{s} F_t(x) \\ &\text{s.t. } x \in \Lambda_{B_1} \cap \Lambda_{B_2} \cap \cdots \cap \Lambda_{B_s} \end{aligned}$$

where F_t is affine on Λ_{B_t} and therefore F is affine on $\Lambda_{B_1} \cap \Lambda_{B_2} \cap \cdots \cap \Lambda_{B_s}$. Thus, the restricted problem is an LP, and hence, there exists an optimal solution in a vertex of its feasible set.

As opposed to carrying out the optimization of a restricted problem, we may find all vertices of $\Lambda_{B_1} \cap \Lambda_{B_2} \cap \cdots \cap \Lambda_{B_s}$ and evaluate these. These are contained in the vertices of the sets $\Lambda_{B_1}, \Lambda_{B_2}, \dots, \Lambda_{B_s}$. For this reason, we find all vertices of all critical regions in $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_s$ and avoid the exponential number of combinations of critical regions to examine. To determine all vertices of a critical region, we use the vertex enumeration of Avis (2000), see also Avis and Fukuda (1996).

Since vertex enumeration is computationally expensive, we also propose a heuristic that replaces vertex enumeration by solving the following LP over each critical region

$$\begin{aligned} &\min c^T x + F_{t,B_t}(x_{B_t}) + \nabla F_{t,B_t}(x_{B_t})^T (x - x_{B_t}) \\ &\text{s.t. } x \in \Lambda_{B_t} \end{aligned}$$

A global optimal solution is not guaranteed by the heuristic. We do, however, maintain the advantages of decomposition.

4.5 Reformulation and linearization

As an alternative to the parametric programming approach, we discuss the possibilities of reformulation and linearization. If the lower level is an LP it can be replaced by its

Karush-Kuhn-Tucker (KKT) optimality conditions (including primal and dual feasibility and complementarity), and the bilevel programming problem can be formulated as a (single-level) mathematical programming problem with equilibrium constraints (MPEC). By introducing binary variables to handle the complementarity constraints, the MPEC can, in turn, be formulated as a mixed-integer programming (MIP) problem. Such reformulation, however, may not facilitate a solution to the general BPP-D when the resulting MIP is neither linear nor convex, and thus, is challenging to many solvers.

Linearization is possible for special cases of a BPP-D. Our linearization is based on exploiting the complementarity constraints of the lower-level LP, which are bilinear. Throughout this section, we therefore assume that $f(\cdot,\cdot)$ is a bilinear function.

Assumption 4.5.1. *Assume that $f(\cdot,\cdot)$ is a bilinear function on the form*

$$f(y,\lambda) = d^T y + h^T \lambda + \lambda^T H y.$$

In the following, we establish conditions under which the bilinear term can be linearized and the bilevel problem can be reformulated to an MPEC or a MILP with linear objective function and that can be solved using standard solvers.

As a prerequisite, we make a number of definitions. Define the sets

$$J_0 \triangleq \{j \in \{1,2,\dots,l\} : H_{ij} = 0 \; \forall i\},$$
$$I_0 \triangleq \{i \in \{1,2,\dots,k\} : D_{ij} = 0 \; \forall j\}.$$

Hence, J_0 holds the indices of the columns of H that contain only zeros and I_0 holds the indices of the rows of D that contain only zeros. Moreover, we let $\bar{J}_0 \triangleq \{1,2,\dots,l\} \backslash J_0$, $\bar{I}_0 \triangleq \{1,2,\dots,k\} \backslash I_0$. Finally, define

$$K_1 \triangleq \{j \in J_0 : C_{ij} = 0 \, \forall i \in \bar{I}^0\},$$
$$K_2 \triangleq J_0 \backslash K_1.$$

With the definition of J_0, I_0, K_1 and K_2, we define the following partition of the vectors and matrices of the BPP-D into blocks (by permutations of columns and rows, without loss of generality). The matrix of the bilinear term is:

$$H = \begin{array}{c} \\ \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \quad \bar{J}_0 \quad\; J_0 \\ \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix} \end{array} \triangleq \begin{array}{c} \\ \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \quad \bar{J}_0 \quad J_0 \\ \begin{pmatrix} H_{11} & 0 \\ H_{21} & 0 \end{pmatrix} \end{array}$$

The lower-level constraint matrices are divided into the following blocks:

$$C = \begin{array}{c} \\ \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \quad \bar{J}_0 \quad K_1 \quad K_2 \\ \begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \end{pmatrix} \end{array} \triangleq \begin{array}{c} \\ \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \quad \bar{J}_0 \quad K_1 \quad K_2 \\ \begin{pmatrix} C_{11} & 0 & C_{13} \\ C_{21} & C_{22} & C_{23} \end{pmatrix} \end{array}$$

$$D = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{pmatrix} D_1 \\ D_2 \end{pmatrix} \triangleq \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{pmatrix} D_1 \\ 0 \end{pmatrix}$$

The lower-level vectors of variables and parameters are divided as follows:

$$\lambda = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} \qquad e = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}$$

$$y = \begin{array}{c} \bar{J}_0 \\ K_1 \\ K_2 \end{array} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \qquad p = \begin{array}{c} \bar{J}_0 \\ K_1 \\ K_2 \end{array} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix}$$

4.5.1 Complementarity constraints and linearization

The following assumptions allow for linearization.

Assumption 4.5.2. *Assume that:*

1. $H_{11} = 0$
2. $C_{21} = \alpha H_{21}$ *for some* $\alpha \in \mathbb{R}\backslash\{0\}$
3. $C_{23} = 0$

By replacing the lower level by its KKT-conditions and linearizing the bilinear term, we obtain an MPEC that is a linear program with linear complementarity constraints. We state and prove this result below. Note that we refer to two equivalent formulations when their optimal solutions are the same.

Proposition 4.5.3. *Under Assumptions 4.5.1 and 4.5.2, an equivalent formulation of the BPP-D is the following linear program with linear complementarity constraints:*

$$\min c^T x + d^T y + h^T \lambda + (\lambda_2^T e_2 - p_2^T y_2)/\alpha \tag{4.10a}$$

$$\text{s.t. } Ax = b \tag{4.10b}$$

$$x \geq 0 \tag{4.10c}$$

$$Cy = Dx + e \tag{4.10d}$$

$$0 \leq y \perp p - C^T \lambda \geq 0 \tag{4.10e}$$

Proof. Since the lower-level is an LP problem, the KKT-conditions are necessary and sufficient for optimality. These conditions are

$$Cy = Dx + e \tag{4.11a}$$

$$y \geq 0 \tag{4.11b}$$

$$C^T \lambda \leq p \tag{4.11c}$$

$$y^T \mu = 0 \tag{4.11d}$$

$$p - C^T \lambda - \mu = 0 \tag{4.11e}$$

including primal feasibility (4.11a)-(4.11b), dual feasibility (4.11c), stationarity (4.11d) and complementarity (4.11e), and are equivalent to (4.10d)-(4.10e).

To linearize the upper-level objective function, we utilize the strong duality property for the lower-level linear program (second equality), the lower-level constraints (4.11a) (fourth equality) and the definitions of I_0 and K_1:

$$\begin{aligned} p_1^T y_1 + p_2^T y_2 + p_3^T y_3 \quad &= p^T y \\ &= \lambda^T (Dx + e) \\ &= \lambda_1^T (D_1 x + e_1) + \lambda_2^T e_2 \\ &= \lambda_1^T (C_{11} y_1 + C_{13} y_3) + \lambda_2^T e_2. \end{aligned}$$

By rearranging terms and noting that $y_1^T C_{11}^T \lambda_1 = \lambda_1^T C_{11} y_1$, we obtain

$$y_1^T (p_1 - C_{11}^T \lambda_1) \quad = -p_2^T y_2 - p_3^T y_3 + \lambda_1^T C_{13} y_3 + \lambda_2^T e_2. \tag{4.12}$$

From the complementarity constraints (4.11d)-(4.11e) and Assumption 4.2 we have that

$$y_1^T(p_1 - C_{11}^T\lambda_1) = \alpha y_1^T H_{21}^T \lambda_2. \tag{4.13}$$

The complementarity constraints (4.11d)-(4.11e) and Assumption 4.3 likewise give us

$$y_3^T p_3 = y_3^T C_{13}^T \lambda_1. \tag{4.14}$$

Substituting (4.13) and (4.14) into (4.12) we arrive at

$$\alpha y_1^T H_{21}^T \lambda_2 = -p_2^T y_2 + \lambda_2^T e_2. \tag{4.15}$$

Using the definitions of J_0 and Assumption 4.1, (4.15) becomes

$$\lambda^T H y = (\lambda_2^T e_2 - p_2^T y_2)/\alpha. \tag{4.16}$$

□

The MPEC (4.10) can be solved using standard non-linear programming (NLP) software or designated MPEC solvers. We refer to these solution approaches as NLP and MPEC, respectively. Note that since the MPEC (4.10) is a non-convex optimization problem, NLP and MPEC solvers can only guarantee local optimality.

4.5.2 Integer programming formulations

The equivalent MPEC (4.10) can be solved to global optimality by introducing binaries to handle the disjunctive complementarity constraints, see Fortuny-Amat and McCarl (1981). The resulting problem is the following MILP:

$$\begin{aligned}
\min\, & c^T x + d^T y + h^T \lambda + (\lambda_2^T e_2 - p_2^T y_2)/\alpha && (4.17a)\\
\text{s.t. } & Ax = b && (4.17b)\\
& Cy = Dx + e && (4.17c)\\
& 0 \le y_j \le M\delta_j \quad \forall j && (4.17d)\\
& 0 \le p_j - \sum_{i=1}^{k} C_{ij}\lambda_i \le M(1-\delta_j) \quad \forall j && (4.17e)\\
& x \ge 0, y \ge 0, \delta \in \{0,1\}^l && (4.17f)
\end{aligned}$$

Here, M is a sufficiently large constant and δ is a vector of binary variables. The choice of M can be difficult, since a small constant may cut off feasible solutions while a large constant may produce a weak LP-relaxation. For a general method to find suitable M's, see also Pineda et al. (2017).

A related formulation is based on special ordered sets (SOS). More specifically, we introduce SOS1 variables, for which only one element of a set is allowed to be non-zero. This likewise guarantees global optimality, cf. Siddiqui and Gabriel (2013). The problem is:

$$\begin{aligned}
\min\, & c^T x + h^T y + k^T \lambda + (\lambda_2^T e_2 - p_2^T y_2)/\alpha && (4.18a)\\
\text{s.t. } & Ax = b && (4.18b)\\
& Cy = Dx + e && (4.18c)
\end{aligned}$$

$$s_j(1) = y_j \quad \forall j \tag{4.18d}$$

$$s_j(2) = p_j - \sum_{i=1}^{k} C_{ij}\lambda_i \quad \forall j \tag{4.18e}$$

$$y \geq 0, x \geq 0, \tag{4.18f}$$

where the sets $\{s_j(1), s_j(2)\}$ are declared as SOS1 for each j.

We refer to these reformulations as MILP and SOS, respectively.

4.6 Strategic investment

The structure of a bilevel programming problem with lower-level primal and dual information in the upper-level objective function may arise in many applications. Important examples are strategic investment problems, in which an investor maximizes the profit of investing (upper level), while anticipating the clearing of the market (lower level). The revenue of is often given by the bilinear product of a lower-level primal solution (production) times a lower-level dual solution (market price).

We illustrate the parametric programming method on three examples of strategic investment in electricity markets. The first example is a stylized problem of investment in generation capacity that serve to illustrate the characteristics of the parametric function. The second and third examples are more detailed numerical case studies of investment in generation and transmission capacity, respectively.

4.6.1 Investment in production capacity

We consider a strategic investor, participating in a perfectly competitive market for dispatch of production and with inelastic demand. We adopt the following problem formulation by Conejo et al. (2016):

$$\min \; 40000x + 8760(10y_1^* - \lambda_1^* y_1^*) \tag{4.19a}$$

$$\text{s.t.} \; 0 \leq x \leq 250 \tag{4.19b}$$

$$y_1^* \in \text{argmin}\{10y_1 + 12y_2 + 15y_3 \tag{4.19c}$$

$$\text{s.t.} \quad y_1 + y_2 + y_3 = 200,\; 0 \leq y_1 \leq x, \tag{4.19d}$$

$$0 \leq y_2 \leq 150,\; 0 \leq y_3 \leq 100\} \tag{4.19e}$$

$$\lambda_1^* \in \text{argmax}\{200\lambda_1 + x\lambda_2 + 150\lambda_3 + 100\lambda_4 \tag{4.19f}$$

$$\text{s.t.} \quad \lambda_1 + \lambda_2 \leq 10,\; \lambda_1 + \lambda_3 \leq 12, \tag{4.19g}$$

$$\lambda_1 + \lambda_4 \leq 15,\; \lambda_2, \lambda_3, \lambda_4 \geq 0\} \tag{4.19h}$$

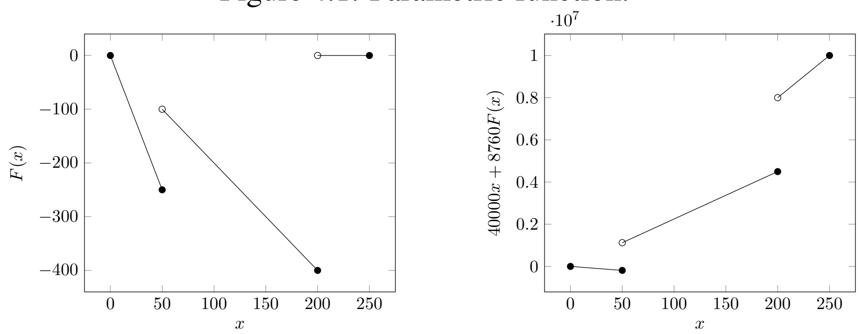
The upper-level variable, x, is generation capacity. The lower-level variables represent dispatch of production from the investor, y_1, and its rivals in the market, y_2, y_3. Moreover, λ_1 is the market (shadow) price of production, and $\lambda_2, \lambda_3, \lambda_4$ are the shadow prices of capacity. The upper-level objective function (4.19a) includes linear investment costs (40000 €/MW), and the number of hours in a year (8760) times the hourly operational (negative) profit. Profit is given by linear production costs (10 €/MWh) less the revenue, which is bi-linear. The upper-level constraints (4.19b) are upper and lower bounds on investment capacity. The lower-level primal problem consists of minimizing total production costs, cf. (4.19c), such that supply meets demand and is bounded by the generation capacities in (4.19d)-(4.19e). The lower-level dual problem, (4.19f)-(4.19h), determines the market price of production. Note that the upper-level variable, x, is a lower-level parameter of (4.19c)-(4.19d) and (4.19f)-(4.19h).

The parametric function is

$$F(x) = 8760(10y_1^*(x) - \lambda_1^*(x)y_1^*(x)),$$

which includes a bilinear product of the lower-level primal and dual variables. The critical regions are the intervals $[0, 50]$, $[50, 200]$ and $[200, 250]$. In Figure 4.1 (a) we plot the parametric function $F(x)$ and confirm that it is affine on the critical regions but discontinuous at $x \in \{50, 200\}$. Clearly, the upper-level objective function plotted in Figure 4.1 (b) is likewise piecewise affine but discontinuous.

Figure 4.1: Parametric function.



(a) Operational (negative) profit. (b) Upper-level objective function.

To explain the behavior of $F(x)$ note that for $0 < x < 50$, all producers are dispatched to meet demand. The market price equals the production costs of the marginal producer, i.e., 15, and the marginal operational profit is $15-10 = 5$. For $50 < x < 200$, producers 1 and 2 are dispatched, the market price is 12 and the marginal operational profit is $12-10 = 2$. For $x > 200$, only producer 1 is dispatched, the market price is 10 and the marginal operational profit is $10 - 10 = 0$. Evidently, on the interiors of the intervals $[0, 50]$, $[50, 200]$ and $[200, 250]$, the basis of the market-clearing problem is unique and the market price is constant. On their boundaries $\{50, 200\}$, however, the basis is no longer unique and there are multiple dual optimal solutions. This demonstrates the ambiguous-price fallacy of power markets noted by Stoft (2002).

It should be remarked that the number of distinct bases of the lower-level problem is at most

$$\binom{6}{4} = 15, \tag{4.20}$$

but the number of critical regions to investigate is only 3.

With the parametrization of F, problem (4.19) becomes

$$\begin{aligned} \min \quad & 40000x + 8760F(x) && (4.21a)\\ \text{s.t.} \quad & 0 \leq x \leq 250. && (4.21b) \end{aligned}$$

The optimal solution is $x^* = 50$ with the objective function value being -190,000.

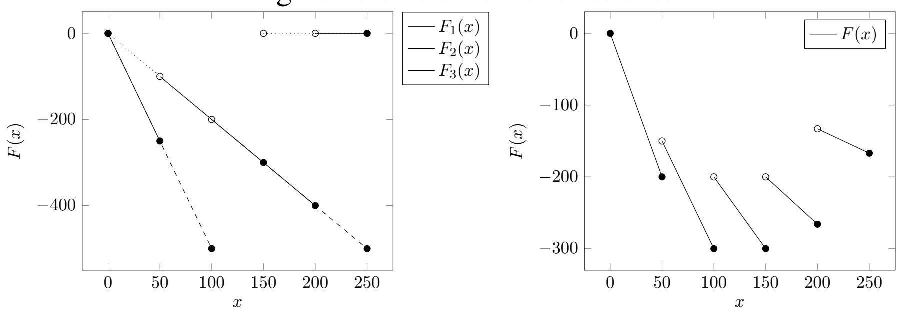
As an alternative to the parametric programming approach, it is possible to use reformulation and linearization. To see that Assumption 4.5.2 holds, note that

$$H = \left(\begin{array}{c|cccc|c} 0 & 0 & 0 & 0 & 0 & 0 \\ \hline -8760 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right), \quad C = \left(\begin{array}{c|cccc|c} 1 & 0 & 0 & 0 & 0 & 1 \\ \hline 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{array}\right),$$

where slack-variables have been introduced in the inequality constraints, the columns are ordered according to $(\bar{J}_0, K_1, K_2)$ and the rows are ordered according to $(\bar{I}_0, I_0)$. Clearly, $H_{11} = C_{23} = 0$ and $C_{21} = -8760 H_{21}$.

We extend the example to three lower-level market-clearing problems with different demands (150MWh, 200MWh and 250MWh, respectively). The critical regions are then the intervals $[0, 50]$, $[50, 200]$, $[200, 250]$ (corresponding to the first lower-level problem), $[0, 150]$, $[150, 250]$ (second lower-level problem) and $[0, 100]$, $[100, 250]$ (third lower-level problem). In Figure 4.2 we confirm that the parametric functions F_1, F_2, F_3 are affine on the corresponding critical regions but discontinuous at their boundaries, and hence, $F(x)$ is affine on the intervals $[0, 50]$, $[50, 100]$, $[100, 150]$, $[150, 200]$ and $[200, 250]$ but discontinuous at $x \in \{50, 100, 150, 200\}$.

Figure 4.2: Parametric functions.



(a) $F_1(x), F_2(x), F_3(x)$ (b) $F(x)$

4.6.2 Implementation

We proceed with the two case studies. To obtain numerical results, the parametric programming approach of Section 4.4 is implemented in R using the interfaces by Berkelaar (2015) to solve LPs and Robere (2015) for vertex enumeration. The MILP formulation is implemented in the General Algebraic Modeling System, GAMS (GAMS, 2017) and solved using CPLEX 12.6.3.0. The non-linear methods are also implemented in GAMS and solved using the CONOPT solver for NLP and the NLPEC solver for MPEC, see GAMS (2017) for details. The problems are solved on an HP ProLiant server with 4 AMD 2.50 GHz CPUs, with a total of 64 cores and 256 GB RAM. Unless otherwise specified, a time limit of 2 hours is imposed on all runs.

4.6.3 Case study I: Investment in production capacity

The first case study is an extension of the problem in Section 4.6.1, i.e., strategic investment in generation capacity. The extension includes a representation of the transmission network by regions and transmission lines connecting the regions. The strategic investor decides how much capacity to install in each region.

Data

In the first case study, we use data from the Danish power market. This market is divided into two price regions connected with a DC cable of 600 MW capacity, cf. Nord Pool AS (2017). We represent each of these price regions by a node and divide the existing power plant capacity into central power plants and de-central power plants as in Danish District Heating Association and Danish Energy Association (2016), which also provide production cost estimates.

Hourly demand data is available at Nord Pool AS (2017). We cluster this data using k-means clustering such as to obtain a set of so-called representative hours. A representative hour replaces a number of hours with similar operating conditions and is weighted by the number of such hours. For methods to reduce a data set in this fashion, see Baringo and Conejo (2013). In the following, we vary the number of representative hours to investigate problems of different sizes.

Model

Our model is a simplified version of a that by Conejo et al. (2016). We assume a perfectly competitive power market, such that the offer price of each producer equals their production cost. Market-clearing accounts for network flow, which is modeled using a DC representation to. The strategic investor has the opportunity to invest in capacity at each node of the network and there are two rivals at each node. The time horizon is a year, and thus, we consider a static investment problem.

The bilevel problem faced by the strategic producer is the following:

$$\min \sum_{i=1}^{n} \left(C_i x_i + \sum_{t=1}^{s} \rho_t \left(c_i^1 y_{it}^1 - \lambda_{it} y_{it}^1 \right) \right) \tag{4.22a}$$

$$\text{s.t. } 0 \leq \sum_{i=1}^{n} x_i \leq x^{max} \tag{4.22b}$$

$$y_{it}^1, \lambda_{it} \text{ are optimal solutions to (4.23) } \forall i, t \tag{4.22c}$$

The lower-level problem involves market-clearing for each time period t. Subproblem t is

$$\min \sum_{i=1}^{n} \sum_{g \in \mathcal{G}} c_i^g y_{it}^g \tag{4.23a}$$

$$\text{s.t. } \sum_{g \in \mathcal{G}} y_{it}^g - \sum_{j \in \mathcal{J}_i} B_{ij}(\theta_{it} - \theta_{jt}) = d_{it} \ : \lambda_{it} \quad \forall i \tag{4.23b}$$

$$0 \leq y_{it}^1 \leq x_i \ : \mu_{it}^1 \quad \forall i \tag{4.23c}$$

$$0 \leq y_{it}^g \leq y_i^{max,g} \ : \mu_{it}^g \quad \forall i, g \neq 1 \tag{4.23d}$$

$$B_{ij}(\theta_{it} - \theta_{jt}) \leq F_{ij}^{max} \ : \eta_{ijt} \quad \forall j \in \mathcal{J}_i \tag{4.23e}$$

$$-\pi \leq \theta_{it} \leq \pi \ : \alpha_{it}, \beta_{it} \quad \forall i \tag{4.23f}$$

$$\theta_{it} = 0 \; : \gamma_t \quad i = \text{ref} \tag{4.23g}$$

The upper-level variable, x_i, is generation capacity in node i. The set $\mathcal{G}$ defines all market participants. The lower-level variables represent dispatch of production in node i and time period t, from the investor, y^1_{it}, and its rivals in the market, $y^g_{it}, g \neq 1$. Moreover, λ_{it} is the market's nodal price of production, and $\mu^g_{it}, \eta_{ijt}, \alpha_{it}, \beta_{it}, \gamma_t$ are the remaining dual variables. The upper-level objective (4.22a) is to maximize profits from investment and operation, operational profits being the sum of profits for each time period t weighted by the number of time periods represented by t, ρ_t. The upper-level constraint (4.22b) bounds total investment capacity. The lower-level constraints include a power balance (4.23b) for each node and the dual variable λ_{it} that determines the nodal price. The set $\mathcal{J}_i$ defines all nodes connected to node i, B_{ij} denotes the subsceptance of the transmission line between nodes i and j and θ_{it} is the voltage angle in node i and time period t, determining the flow. Thus, the sum on the left-hand-side of the equality is the net outflow from node i. Further constraints are generation capacity constraints (4.23c) and (4.23d), transmission capacity constraints (4.23e) and bounds on the voltage angles (4.23f). Finally, (4.23g) fixes the voltage angle of some reference node to zero. A complete nomenclature is provided in the Appendix B.1. For more details regarding the model, we refer to Conejo et al. (2016).

The equivalent MILP is formulated in Appendix B.2.

Results

We solve 59 instances of the investment problem, varying the number of lower-level subproblems from 10 to 100 with a step size of 10 and from 200 to 5000 with a step size of 100. For each of the solutions methods of Sections 4.4 and 4.5, we report the number of instances for which the method terminates, the time limit is reached and the method returns infeasibility, respectively, see Table 4.1. As seen from the table, the NLP and

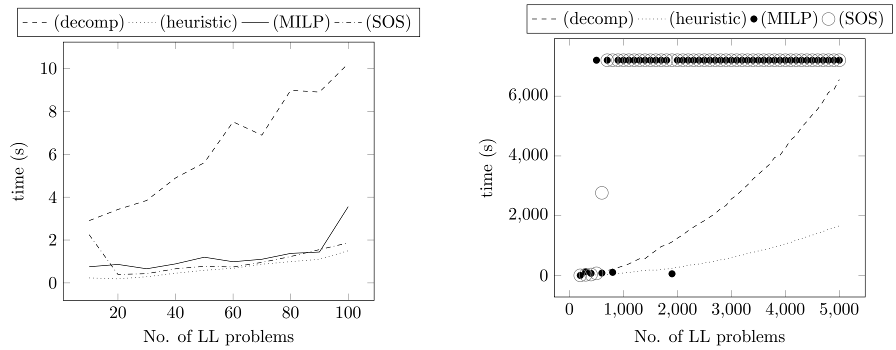
Table 4.1: Number of instances for which a solution method terminates, the time limit is reached and the method returns infeasibility, respectively.

	Solved	Time Limit	Reported Local Infeasible
decomp	59	0	0
heuristic	59	0	0
full	9	50	0
MILP	16	43	0
SOS	15	44	0
NLP	0	0	59
MPEC	0	0	59

MPEC methods return local infeasibility for all instances, even when we provide an initial feasible solution as in the parametric programming approaches. As shown by Scheel and Scholtes (2000), all feasible points of the MPEC (4.10) are non-regular (i.e., the gradients of the binding constraints are linearly dependent), which is the reason that most non-linear optimization solvers fail. The MILP and SOS solve 16 and 15 instances, respectively, while the full parametric programming method solves only 9 instances within the time limit. In contrast, however, both decomposition by parametric programming and the parametric programming heuristic solve all instances to optimality.

We further investigate the MILP and parametric programming approaches. Figure 4.3 shows the solution times for MILP, the decomposition by parametric programming and the parametric programming heuristic, plotted as functions of the number of lower-level subproblems. Figure 4.3 (a) and (b) cover the number of lower-level problems from 10 to 100 and from 200 to 5000, respectively. Note that running times for the MILP are truncated to the time limit of 2 hours (43 out of 49 runs are above the time limit). Note also that the figure does not include parametric programming without decomposition, since solution times are substantially larger and the method does not terminate within the time limit for more than 100 lower-level subproblems. A complete list of results can be found in Table B.3 of Appendix B.3.

Figure 4.3: Solution times (seconds) as a function of the number of lower-level (LL) subproblems.



(a) No. of LL problems from 10 to 100. (b) No. of LL problems from 200 to 5000

From the results, we notice that for less than 100 subproblems, the MILP has the lower solution times, whereas for more than 100 problems, the heuristic is faster. For more than 800 subproblems, the MILP does not solve within the time limit (except with 1900 subproblems), and thus, although parametric programming cannot compete with the heuristic, this method also solves faster than the MILP.

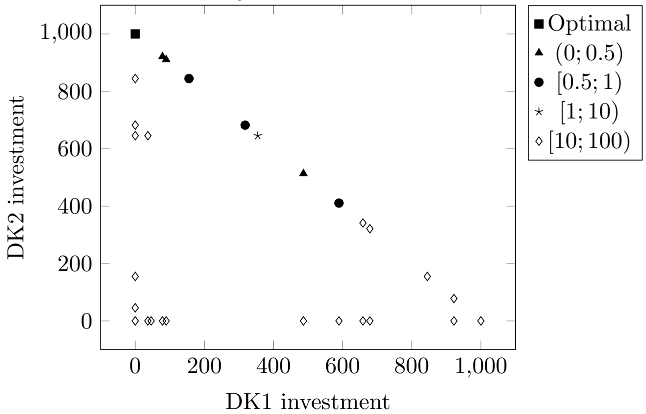
Decomposition by parametric programming solves all problems to optimality. The average relative deviation in upper-level objective function values from the optimal is 4.26% for the heuristic and 3.37% for the cases in which the MILP does not solve within the time limit (and thus only returns a feasible solution), see also Appendix B.3.

The parametric approach shows a significant increase in solution times as the number of subproblems increases. Although solution times behave similarly for the heuristic, the increase is less. In spite of this increase in solution times, the decomposition approaches are much less sensitive to increasing the number of subproblems than the MILP. The reason is that only the number of problems increases with decomposition, whereas the size of the problem increases with MILP.

The solution quality of the heuristic increases as the number of subproblems increases, whereas it is almost constant for the MILP. We explain this as follows. With a larger number of lower-level subproblems, the number of critical regions considered during the heuristic is larger. Solution quality may increase since the approximation error has less impact, this error resulting from using an optimal solution rather than inspecting all vertices of a critical region. The reason for a constant solution quality of the MILP is the structure of the investment problem. In this problem, the lower level problem represents a number of separate market-clearing conditions, e.g. for representative hours of a year. The problem is therefore almost the same, irrespective of the number of lower-level subproblems.

To further explore MILPs that do not solve within the time limit, we have solved the instance with 5000 lower-level problems and a higher time limit. The problem has 215,000 variables, including 90,000 binary, and 260,000 constraints. The MILP did not solve within a week's time limit (604,800 seconds) and, hence, is intractable in any practical aspect. In contrast, the decomposition by parametric programming solved this instance to global optimality in 6487 seconds and the heuristic with a deviation from the global optimal value of 3.61% in 1658 seconds.

Figure 4.4: Feasible solutions found by the parametric programming approach. Divided into intervals of deviation (in %) in objective function value from the optimal.



In Figure 4.4, we examine the case with 10 lower-level subproblems and depict the feasible solutions found by the parametric programming approaches by solving the restricted problems or by enumeration of the vertices. The solutions are plotted with the investment decisions in DK1 and DK2 on the x-axis and y-axis, respectively. For each feasible solution, we indicate the corresponding deviation (in percentage) in upper-level objective function value from the optimal. All parametric programming approaches find the optimal solution $(0, 1000)$, but only the exact approaches automatically provide near-optimal solutions. For example, the solution $(486, 514)$ has an optimality gap of less than 0.5% and may, therefore, be relevant for further inspection.

The figure also reveals the non-convexity of the upper-level objective function. In fact, the straight line from the optimal solution $(0, 1000)$ to the solution $(486, 514)$ includes objective function values with an optimality gap of up to 10%.

4.6.4 Case study II: Transmission investment

The next case study concerns investment in transmission capacity. A merchant investor decides how much capacity to install on selected connections between regions of the transmission network.

Data

In the second case study, we use data from the Danish, Swedish and Norwegian power markets and represent the two Danish price regions, Norway and Sweden with four nodes. We assume that two DC cables are already in place: One connecting the two Danish price regions and one connecting the western Danish pricing region and Sweden. Four potential DC cables can be installed, providing connections where not already.

Hourly demand data is the same as for Case study I.

Model

We continue to assume a perfectly competitive power market and market-clearing that accounts for network flow. For the problem to be linearly constrained, however, we use a more simple network representation than above.

The bilevel programming problem of the merchant investor is to decide on the upper-level transmission capacity while anticipating the lower-level market-clearing:

$$\min \sum_{(i,j)\in\mathcal{J}^1} \left(C_{ij}x_{ij} - \sum_{t=1}^{s} \rho_t f_{ijt} \left(\lambda_{it} - \lambda_{jt}\right) \right) \tag{4.24a}$$

$$\text{s.t. } 0 \le \sum_{(i,j)\in\mathcal{J}^1} C_{ij}x_{ij} \le x^{max} \tag{4.24b}$$

$$0 \le x_{ij} \le F_{ij}^{max} \quad \forall (i,j) \in \mathcal{J}^1 \tag{4.24c}$$

$$f_{ijt}, \lambda_{it} \text{ are optimal solutions to (4.25) } \forall i, j, t \tag{4.24d}$$

The lower-level subproblems are:

$$\min \sum_{g\in\mathcal{G}} c^g y_t^g \tag{4.25a}$$

$$\text{s.t. } \sum_{g\in\mathcal{G}_i} c^g y_t^g - \sum_{j\in\mathcal{J}_i} f_{ijt} = d_{it} \ : \lambda_{it} \quad \forall i \tag{4.25b}$$

$$0 \le y_t^g \le y_{max}^g \ : \mu_t^g \quad \forall g \in \mathcal{G} \tag{4.25c}$$

$$- x_{ij} \le f_{ijt} \le x_{ij} \ : \alpha_{ijt}^1, \beta_{ijt}^1 \quad \forall (i,j) \in \mathcal{J}^1 \tag{4.25d}$$

$$- F_{ij}^{max} \le f_{ijt} \le F_{ij}^{max} \ : \alpha_{ijt}, \beta_{ijt} \quad \forall (i,j) \notin \mathcal{J}^1 \tag{4.25e}$$

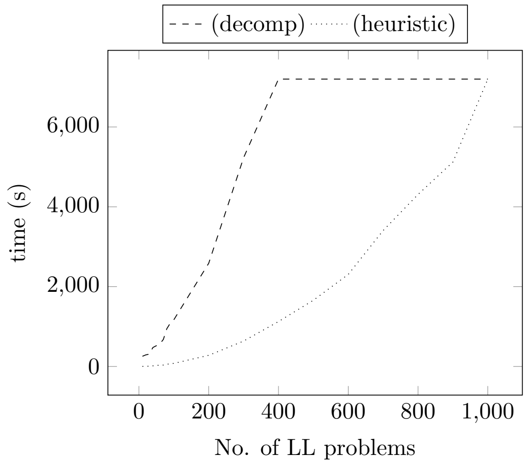
The set $\mathcal{J}^1$ defines all potential transmission lines of the merchant. The upper-level objective is to maximize the profits from building new transmission lines and receiving congestion rents, cf. Joskow and Tirole (2005). Congestion rents are defined as the flow on a line times the difference in nodal prices for the two connected nodes, see Khanabadi and Ghasemi (2011) and Sorokin et al. (2012). Constraint (4.24b) introduces an investment budget, whereas (4.24c) constrain the maximum capacity to be installed. The lower-level subproblems are simplifications of Case study I, obtained by replacing the DC representation by flow balancing only and assuming only one market participant in each node of the network.

Results

Note that the investment problem (4.24)-(4.25) satisfies Assumption 3 but not Assumption 4.5.2. More specifically, the submatrices C_{21} and H_{21} defined in Section 4.5 do not satisfy Assumption 4.5.2.2. For this reason, the liearization and the MILP and SOS reformulations do not apply. Moreover, based on the experience with NLP and MPEC from Case study I, we restrict attention to the parametric programming approaches. We report the results of decomposition by parametric programming and the parametric programming heuristic. We do not include those of parametric programming without decomposition, since method does not solve the problem within the time limit.

Figure 4.5 plots the solution times as functions of the number of lower-level subproblems.

Figure 4.5: Solution times (seconds) as a function of the number of lower-level (LL) problems.



We note that decomposition by parametric programming has a significantly longer solution times for the second case study than for the first, cf. Figure 4.5 versus Figure 4.3. For Case study II, the method only solves instances with less than 400 lower-level subproblems within the time limit. This is due to larger lower-level subproblems and thus critical regions of higher dimensions, which makes vertex enumeration computationally expensive. The expensive vertex enumeration is also the reason that the heuristic largely outperforms decomposition. In fact, the heuristic method solves problems with up to 900 lower-level subproblems within the time limit.

Considering this particular investment problem, for all instances solved by the parametric programming method within the time limit (and returning a global optimal solution), the heuristic method returns the same solution, and hence, also solves the instances to global optimality.

4.7 Conclusions

This paper examines linearly constrained bilevel programming problems in which the upper-level objective function depends on both the lower-level primal and dual optimal solutions. We provide a formal analysis of the BPP-D and suggest solution strategies based on parametric programming. In particular, we propose an exact algorithm and a heuristic that both facilitate decomposition of the problem and thereby the potential to provide computational advantages.

We assess the performance of the parametric programming approach on two case studies of strategic investment in electricity markets and benchmark against mixed-integer linear programming and standard non-linear solution methods to bilevel optimization problems. Preliminary results reveal substantially lower solution times to obtain high-quality feasible solutions and/or reach global optimality over state-of-the-art methods, especially when the lower-level separates into a large number of subproblems. Furthermore, we show that the parametric programming approach succeeds in solving problems to global optimality, for which linearization does not apply and non-linear methods fail.

For future work, one could test computational performance in other application domains. Furthermore, throughout the analysis we assume that the upper-level objective is affine in one of its arguments or even bilinear in both arguments, which covers a wide range of applications. One could, however, investigate whether the parametric programming approach maintains its advantages for more general BPP-Ds or other bilevel optimization problems.

Chapter 5

A Parametric Programming Approach to Bilevel Transmission Investment Problems in Power

H. BYLLING, T. BOOMSMA AND S. GABRIEL

Chapter Abstract

Nowadays, transmission investments are undertaken in a liberalized market environment, in which the transmission system operator, the market, producers and investors have different objectives. The transmission expansion problem can account for this through the use of bilevel programming, with an investor making expansion decisions in an upper-level while anticipating the result of a lower-level market-clearing. In this work, we formulate the stochastic transmission expansion problem of a merchant investor collecting congestion rents that are determined by the differences between nodal market prices. The corresponding bilevel program can be recast as a mathematical program with equilibrium constraints (MPEC), but does not allow for linearization and reformulation by mixed-integer linear programming. Instead, we present a parametric programming approach that facilitates decomposition with respect to both time periods and scenarios. A numerical study indicates its advantages over a non-linear MPEC representation.

5.1 Introduction

Transmission expansions in a power market may involve many players with different objectives. For instance, a system operator aims to improve the functioning of the power system, e.g. through social welfare maximization or with respect to reliability of the network. Generation companies assess the effects of transmission expansions on their profits, since changes to the network topology involve changes to supply and demand. In this chapter, we take the perspective of a merchant transmission investor, i.e. a company that installs new transmission lines in order to profit from their use. We assume a power network with nodal prices, also known as locational marginal prices (LMPs). As a producer sells its power in the node it is located and at the local LMP, flow of power to another node with a different LMP may involve profits to the owner of the line. Indeed, the profits from using both existing and newly installed transmission lines consist of congestion rents defined by differences in nodal prices (Sorokin et al., 2012). In many cases, a transmission system operator (TSO) owns and operates the network and the profits translate into financial transmission rights (FTRs) which are sold in a secondary market or auction. In contrast, the merchant investor perspective to transmission investments is based on profits from long-term financial transmission right (LTFTR) offsetting the investment costs (Rosellón et al., 2013). This market setup can for example be found in PJM, New York, California and New England (Kristiansen, 2004).

As transmission expansions change the network topology, supply and demand is affected and the market adopts new LMPs. In particular, the installation of new transmission lines can connect producers to new nodes, in which the merit order and therefore the local LMP changes. To model this feedback mechanism between investment decisions and LMPs and the different objectives of the merchant and the market, we use bilevel programming. A bilevel programming problem (BPP) consists of an upper level and a lower level, often illustrated by the leader-follower problem (or Stackelberg game) in which a leader makes an upper-level decision while accounting for the reaction of a follower in the lower-level. We consider the merchant investor as a leader making upper-level investment decisions that anticipate the following lower-level market-clearing. Our problem of long-term transmission expansions is static, but account for short-term dynamics of the power system, including market clearing. Moreover, by including demand uncertainty, our problem becomes a two-stage stochastic program, with the first stage and second stages being the upper level and lower level, respectively.

A popular approach to solving a BPP is based on replacing the lower-level problem by its Karush-Kuhn-Tucker (KKT) optimality conditions, assuming these are sufficient (Dempe et al., 2015). The resulting problem is a mathematical program with equilibrium constraints (MPEC), for which solution approaches include reformulation by mixed integer programming (MIP), non-linear methods or heuristics. In case the BPP has linear constraints and objectives, a widely applied method uses linearization and reformulation by mixed-integer linear programming. In our case, the lower-level problem of the BPP is a linear program, meaning that the KKT conditions are necessary and sufficient for optimality. Also, the upper-level problem has linear constraints. However, the upper-level objective function includes bilinear congestion rents, determined by LMPs (lower-level dual variables) times line flow (lower-level primal variables). This bilinear term makes the resulting MPEC non-linear, and thus, difficult to solve to global optimality.

Instead, we propose a solution approach for the merchant investor BPP based on parametric programming. The method has the advantage that it solves a BPP with bilinear objective to global optimality. Furthermore, it allows for decomposition of the lower-level problem with respect to both time periods and scenarios. In a numerical study, we compare its performance to the non-linear MPEC representation.

The main objectives of this chapter are:

- To formulate a bilevel programming problem for transmission expansion of a merchant investor.
- To illustrate the application of parametric programming and its advantages for the transmission investment problem.
- To obtain numerical results for a case study of investments in transmission lines.

The rest of this chapter is organized as follows: Section 5.2 gives an overview of the existing literature and positions this chapter within recent research. In Section 5.3, we present the bilevel programming problem of transmission investment and in Section 5.4, the proposed solution method is described. Numerical results from a case study are given in Section 5.5 and Section 5.6 concludes.

5.2 Literature Review

In the current literature, BPPs have often been used to formulate transmission expansion problems. For a review of transmission expansion problems in general, we refer to (Hemmati et al., 2013).

For instance, Conejo et al. (2016) present a bilevel transmission and generation expansion problem with market clearing in the lower-level and profit maximization at the upper-level. Similarly, Garcés et al. (2009) propose a bilevel problem of a transmission planner who minimizes network expansion costs in the upper level subject to market clearing at the lower level. Also, Baringo and Conejo (2012a) consider a joint generation and transmission expansion with the objective to minimize consumer payments when installing wind power units and the required network reinforcements. Conejo et al. (2016), Garcés et al. (2009) and Baringo and Conejo (2012a) solve the bilevel programming problem by reformulation to a mixed-integer linear program (MILP) via KKT-conditions and disjunctive constraints. In fact, although the upper-level objective function by (Conejo et al., 2016) and (Baringo and Conejo, 2012a) is bilinear, it can be linearized using the KKT-conditions and strong duality of the lower-level problem. Unfortunately, to the best of our knowledge, the structure of our problem does not allow for linearization and reformulation by MILP.

The perspective of a merchant investor is proposed by Joskow and Tirole (2005). (Maurovich-Horvat et al., 2014) formulate a stochastic bilevel problem and use it to compare transmission expansions from a merchant investor and a TSO. Buijs and Belmans (2012) likewise propose a bilevel transmission expansion problem and analyze different upper-level objectives. More specifically, Rosellón et al. (2013) investigate a merchant mechanism to transmission expansion, using LTFTR as incentive to construct new lines. The resulting problem becomes an MPEC, which is solved via its KKT-conditions. Since the MPEC is non-convex, the KKT conditions may not be sufficient for optimality, and thus, the solution may not be globally optimal. In this chapter, we continue to consider a merchant perspective on transmission expansion, but introduce a new solution method that guarantees global optimality.

Other solution approaches to bilevel programming include genetic heuristics, e.g. (Buijs and Belmans, 2012). For a review of solution methods to BPPs, we refer to (Dempe et al., 2015) and (Colson et al., 2007). Neither of these solution methods, however, solve general bilevel transmission expansion problems with bilinear objective to global optimality.

5.3 Model

This section presents the bilevel programming model for transmission investments. Our model consists of two levels; a lower-level market-clearing problem and an upper-level investment problem. A nomenclature is provided in Table C.1 in the appendix.

In the lower-level market-clearing problem, we assume a perfectly competitive power market, such that producers offer generation at their marginal production cost. By further assuming inelastic demand, market-clearing can be formulated as a linear cost minimization problem, cf. Gabriel et al. (2013). In our setup, market clearing accounts for network flow, which is modeled using a DC representation. To capture the short-term dynamics, we consider a number of time periods, e.g. hours, for which the power market clears. To represent demand uncertainty, we assume a discrete distribution with a finite number of scenarios. For fixed upper-level decisions, the lower-level problem decomposes into a number of subproblems; one for each time period and each scenario. The lower-level subproblem of time period t and scenario s is the following:

$$\min_{\mathbf{y}_{ts},\mathbf{p}_{ts},\boldsymbol{\theta}_{ts}} \quad \sum_{g\in G} c_g y_{gts} \tag{5.1a}$$

$$\text{s.t.} \quad \sum_{g\in G(i)} y_{gts} - \sum_{j\in\Omega(i)} p_{ijts} = d_{its} \ : \lambda_{its} \quad \forall i \in \Omega \tag{5.1b}$$

$$0 \le y_{gts} \le y_g^{max} \ : \mu_{gts}^{y} \quad \forall g \in \mathcal{G} \tag{5.1c}$$

$$p_{ijts} = B_{ij}(\theta_{its} - \theta_{jts}) \ : \mu_{ijts}^{p} \quad \forall (i,j) \notin \mathcal{C} \tag{5.1d}$$

$$p_{ijts} = x_{ij} B_{ij}(\theta_{its} - \theta_{jts}) \ : \mu_{ijts}^{p,\mathcal{C}} \quad \forall (i,j) \in \mathcal{C} \tag{5.1e}$$

$$-F_{ij}^{max} \le p_{ijts} \le F_{ij}^{max} \ : \mu_{ijts}^{F,min}, \mu_{ijts}^{F,max} \quad \forall (i,j) \notin \mathcal{C} \tag{5.1f}$$

$$-\mathcal{F}_{ij} \le p_{ijts} \le \mathcal{F}_{ij} \ : \mu_{ijts}^{\mathcal{F},min}, \mu_{ijts}^{\mathcal{F},max} \quad \forall (i,j) \in \mathcal{C} \tag{5.1g}$$

$$-\pi \le \theta_{its} \le \pi \ : \mu_{its}^{\theta,min}, \mu_{its}^{\theta,max} \quad \forall i \in \Omega \tag{5.1h}$$

$$\theta_{its} = 0 \ : \mu_{ts}^{\theta,ref} \quad i = ref \tag{5.1i}$$

where $\mathbf{y}_{ts} = \{y_{gts}\}_g$, $\mathbf{p}_{ts} = \{p_{ijts}\}_{i,j}$ and $\boldsymbol{\theta}_{ts} = \{\theta_{its}\}_i$. The objective function minimizes production costs, while the constraints (5.1b) balances demand and supply at each node. Power generation is limited by the existing capacity for each generating unit in constraint (5.1c). Power flow from node i is defined for existing lines in constraint (5.1d) and for potential lines in (5.1e). Since the upper-level investment decision is fixed in the lower level, we know whether a candidate line has been installed ($x_{ij} = 1$) or not ($x_{ij} = 0$). The flow through each line is likewise constrained by the capacity (5.1f) for existing lines and in constraint (5.1g) for candidate lines. The voltage angle at each node is restricted by the constraint (5.1h), and finally, (5.1i) defines the voltage angle of some reference node of the network to be zero.

In the upper-level investment problem, the merchant maximizes profits, i.e. congestion rents less investment costs, subject to a total budget. The formulation of the upper-level problem is the following:

$$\max_{\mathbf{x},\boldsymbol{\mathcal{F}},\mathbf{p},\boldsymbol{\lambda}} \quad \sum_{t\in T} \rho_t \sum_{s\in S} \phi_s \sum_{i,j\in\Omega:i<j} p_{ijts}\left(\lambda_{its} - \lambda_{jts}\right) - \sum_{(i,j)\in\mathcal{C}} \left(K_{ij}x_{ij} + k_{ij}\mathcal{F}_{ij}\right) \tag{5.2a}$$

$$\text{s.t.} \quad 0 \le \sum_{(i,j)\in\mathcal{C}} K_{ij}x_{ij} + k_{ij}\mathcal{F}_{ij} \le K^{max} \tag{5.2b}$$

$$0 \le \mathcal{F}_{ij} \le x_{ij}\mathcal{F}_{ij}^{max} \quad \forall i,j \in \Omega \tag{5.2c}$$

$$x_{ij} \in \{0,1\}, \quad \forall i,j \in \Omega \tag{5.2d}$$

$$\mathbf{p}_{ts} \text{ primal solution to (5.1)}, \quad \forall t \in T, s \in S \tag{5.2e}$$
$$\boldsymbol{\lambda}_{ts} \text{ dual solution to (5.1)} \quad \forall t \in T, s \in S \tag{5.2f}$$

where $\mathbf{x} = \{x_{ij}\}_{i,j}, \boldsymbol{\mathcal{F}} = \{\mathcal{F}_{ij}\}_{i,j}, \mathbf{p} = \{p_{ijts}\}_{i,j,t,s}$ and $\boldsymbol{\lambda} = \{\lambda_{ts}\}_{t,s}$. The objective function maximizes the profits from installing new lines. Profits consists of accumulated hourly congestion rents determined by the differences between nodal market prices less fixed and linear investment costs. Constraint (5.2b) ensure compliance with the investment budget and (5.2c) constrain the maximum capacity installed at each line.

5.4 The parametric programming method

By replacing the lower-level problem of the BPP by its Karush-Kuhn-Tucker (KKT) optimality conditions, the resulting problem is a mathematical program with equilibrium constraints (MPEC). The bilinear term of the upper-level objective function makes the MPEC non-linear. To the best of our knowledge, it is not possible to linearize this bilinear term and the problem can only be solved to local optimality.

Instead, we propose a solution approach for the BPP based on parametric programming. The approach applies to a linearly constrained BPP with continuous variables in both levels, and thus, does not apply directly to the transmission investment problem with binary variables in the upper level. For a limited number of potential transmission lines, the number of binaries is moderate (for $|\mathcal{C}|$ potential lines, the number of binaries is $2^{|\mathcal{C}|}$). We therefore use the parametric programming approach in combination with an enumeration method. Our method has the advantage that it solves the problem with bilinear objective to global optimality. In Section 5.4.1 we present the parametric programming method for a BPP with only continuous variables and in Section 5.4.2 we briefly explain the enumeration method to deal with binary variables.

5.4.1 Continuous Upper Level

In this section, we fix the binary decisions to install a line or not $\mathbf{x} \in \{0, 1\}^{|\mathcal{C}|}$ and consider only the continuous line capacities $\boldsymbol{\mathcal{F}} \in \mathbb{R}^{|\mathcal{C}|}$ as upper-level decision variables.

We define the upper-level feasibility set $S \subseteq \mathbb{R}^{|\mathcal{C}|}$ as the set of upper-level solutions that satisfy the upper-level constraints (5.2b),(5.2c) and render the lower-level problem (5.1) feasible.

The idea behind the parametric programming method is to parameterize the lower-level primal and dual optimal solutions by the upper-level feasible solutions, i.e.

$$\mathbf{p}(\boldsymbol{\mathcal{F}}) \text{ and } \boldsymbol{\lambda}(\boldsymbol{\mathcal{F}}), \quad \boldsymbol{\mathcal{F}} \in S,$$

such the upper-level objective function can be expressed in terms of upper-level variables only.

To inspect the optimal solutions to the lower-level problem, let the upper-level solution $\boldsymbol{\mathcal{F}} \in S$ be fixed and let B be a basis for the lower-level linear programming problem, i.e. a set of linearly independent columns of the constraint matrix. We refer to the corresponding basis solution to the lower-level problem. Variables corresponding to columns of the basis are called basic variables. The remaining variables are called non-basic and equal zero.

The following definition stems from parametric programming (Gal, 1995).

Definition 5.4.1. The *critical region* $\Lambda_B \subseteq S$ corresponding to the basis B is the set of solutions for which the corresponding basis solution is optimal.

Thus, the critical region corresponding to a basis is the set of upper-level solutions for which the corresponding lower-level basis solution is optimal. It can be shown that a critical region is a polyhedron, cf. Gal (1995).

On each critical region we can characterize and explicitly express the upper-level objective function in terms of upper-level variables only. This result follows from Bylling et al. (2018).

Proposition 5.4.2. *Let Λ_B be the critical region corresponding to the basis B. Then the bilinear term $p_{ijts}(\boldsymbol{\mathcal{F}})(\lambda_{its}(\boldsymbol{\mathcal{F}}) - \lambda_{jts}(\boldsymbol{\mathcal{F}}))$ is an affine function of $\boldsymbol{\mathcal{F}}$ on the interior of Λ_B and for all i, j, t, s.*

In other words, the upper-level objective function is a piece-wise linear (but not necessarily continuous) function with affine segments on each critical region. Hence, with the gradient of the upper-level objective function on a critical region, we can obtain an exact description of the upper-level objective function. Furthermore, with an affine objective function and a polyhedral feasibility set, the restriction of the BPP to a single critical region is a linear programming problem. We use this to solve the BPP.

Our strategy is to find a cover of the upper-level feasibility set by critical regions, i.e. a set of bases $\mathcal{B}$ such

$$S = \bigcup_{B \in \mathcal{B}} \Lambda_B,$$

solve the restricted problems for all critical regions in the cover and finally obtain the global optimal solution by simply comparing solutions.

To find a cover of S by critical regions, we define neighboring critical regions (Gal, 1995).

Definition 5.4.3. Two critical regions, Λ_1, Λ_2, are *neighbors* if the following holds for their corresponding bases B_1, B_2:

1. There exists an $\boldsymbol{\mathcal{F}} \in S$ for which B_1 and B_2 are both optimal bases to (5.1).
2. It is possible to pass from B_1 to B_2 in one iteration of the dual simplex method.

By Gal (1995), the union of all neighboring critical regions forms a cover of S. Thus, it is unnecessary to consider all possible bases of the lower-level problem. Neighboring critical regions are obtained by the following algorithm (Gal, 1995), based on dual simplex.

Algorithm 5.4.4. Parametric programming algorithm

Step 0 (initialization) Set $h := 0$. Given an initial upper-level solution, solve the lower-level problem (5.1). Store an optimal basis B_0 and set $\mathcal{B} := \{B_0\}$.

Step 1 (iteration h) If $\mathcal{B} = \emptyset$, then stop. Otherwise, set $h := h + 1$, select $B_h \in \mathcal{B}$ and set $\mathcal{B} := \mathcal{B} \setminus \{B_h\}$.

Step 2 (determine leaving variable) Let $B := B_h$. Select a basic variable i and determine if a neighbor exists. If not, select another basic variable i. If all basic variables have been considered, return to Step 1.

Step 3 (determine entering variable) Carry out an iteration of the dual simplex method with basic variable i as the leaving variable. Store a neighboring basis B_j and set $\mathcal{B} := \mathcal{B} \cup \{B_j\}$. Return to Step 2.

For further details on our parametric programming approach, see Bylling et al. (2018).

Decomposition

For fixed upper-level decisions, the lower-level problem of the BPP decomposes into a number of subproblems; one for each time period and each scenario. We refer to the BPP with one time period and one scenario as a BPP subproblem. We process the subproblems individually, which allows for parallel computations and most likely provide computational advantages.

By processing a BPP subproblem, we obtain neighboring critical regions for one time period and scenario. By processing all subproblems, the union of all critical regions forms a cover of S. Observe that an optimal solution to the restricted BPP can found in a vertex of the critical region. Unfortunately, the optimal solution to the BPP may not be found among the optimal solutions to the restricted BPP subproblems. However, the vertices of the critical region must be found among the vertices of the critical regions obtained for one time period and scenario. Thus, to find an optimal solution to the BPP, we enumerate and evaluate all vertices of the critical regions of the BPP subproblems. This will provide a global optimal solution. For vertex enumeration, we use the procedure of (Avis and Fukuda, 1996).

The algorithm is as follows:

Algorithm 5.4.5. Decomposition

Step 1 (parametric programming) Apply the parametric programming algorithm 5.4.4 to the BPP subproblem.

Step 2 (vertex enumeration) Use vertex enumeration for each of the critical regions obtained in Step 1.

Step 3 (comparison) Collect all vertices from Step 2 and evaluate the upper-level objective function at these.

As an alternative to Algorithm 5.4.5, we also propose a heuristic that omits the computationally costly vertex enumeration. In Step 2, we obtain optimal solutions to the restricted BPP subproblems.

The heuristic can be summarized as:

Algorithm 5.4.6. Heuristic

Step 1 (parametric programming) Apply the parametric programming algorithm 5.4.4 to the BPP subproblem.

Step 2 (restricted optimization) Solve the BPP subproblem restricted to each critical region obtained in Step 1.

Step 3 (comparison) Collect all vertices from Step 2 and evaluate their upper-level objective function values.

5.4.2 Binary Upper Level

This section outlines the combination of the parametric programming approach and the enumeration method. The idea is simply to iterate through the upper-level solutions, i.e. all potential configurations of the network. For fixed binary decisions to install candidate lines or not $\mathbf{x} \in \{0,1\}^{|\mathcal{C}|}$, we apply parametric programming.

The procedure is as follows:

Algorithm 5.4.7. Enumeration

Step 1 (enumeration) Enumerate all binary solutions, $\mathbf{x}$.

Step 2 (parametric programming) For each solution, solve the BPP using the algorithm 5.4.5 or the heuristic 5.4.6.

Step 3 (comparison) Collect all solutions from Step 2 and their upper-level objective function values.

5.4.3 Non-Linear Programming

As benchmarks, we also implement a non-linear MPEC formulation of the problem and mixed-integer non-linear programming (MINLP) model. Both can be solved using standard software, with the upper-level variables $\mathbf{x}$ defined as binary variables. However, since these problem are non-convex, only local optimality is guaranteed.

A challenge for the standard solver is that all feasible points of the MPEC are non-regular, i.e. the gradients of the binding constraints are linearly dependent. Most non-linear optimization solvers even fail to obtain a locally optimal solution. A way to overcome the non-regularity is by the regularization approach of (Scholtes, 2001) and (Ralph and Wright, 2004). Using this approach, the equality constraints of complementary slackness are replaced by inequalities and the infeasibility gap is iteratively reduced. With inequality constraints, the MPEC is regular.

Alternatively, the complementary slackness constraints can be linearized using disjunctive constraints. Disjunctive constraints introduces a binary variable for each constraint and a large parameter. This parameter, usually denoted by M, has to be sufficiently large not cut off any feasible solutions. A too large M, however, may create computational difficulties, see (Pineda et al., 2017) for more details. The resulting problem is a MINLP (it is a mixed-integer problem but remains non-linear programming due to the bilinear term in the objective function) and can only be solved to local optimality.

5.5 Numerical Results

We present a case study of transmission expansion in the Nordic region, with 4 nodes representing Norway, Sweden and the two Danish pricing regions DK1 and DK2, cf. Nord Pool AS (2017).

5.5.1 Data

We assume that three DC cables are already in place: One connecting the two Danish price regions, one connecting the eastern Danish pricing region and Sweden and one connecting Sweden and Norway. These existing cables each have a capacity of 1.000 MW. Three additional DC cables can be installed, providing connections where not already. These are the cables $(N, DK1), (N, DK2)$ and $(SE, DK1)$, see Figure 5.1. The topology of the network is not as the current, but is chosen for the purpose of illustration. The variable investment costs of each candidate line is assumed to be 20.000 DKK/MW, whereas we disregard fixed investment costs. We likewise disregard a budget and limitations for installed capacities of candidate transmission lines.

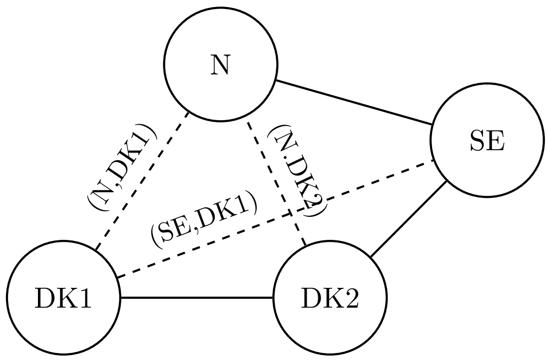


Figure 5.1: Network topology.

Hourly demand data at each node is available at (Nord Pool AS, 2017). We use the year of 2015. This data is clustered into a number of representative hours using k-means clustering (Hartigan and Wong, 1979). We obtain results for different numbers of representative hours. For simplicity, we disregard demand uncertainty.

Generation capacity and costs for DK1 and DK2 is taken from (Ea Energianalyse, 2014) that divides generation into and decentralized units. Generation capacities are adjusted to the Norwegian and Swedish nodes by considering historical and production data. As opposed to Denmark, Norway and Sweden have considerable amounts of hydropower, which is reflected in the lower production costs of the centralized plants. The generation capacities and production costs are found Table 5.1.

	DK1	DK2	SE	N
Centralized cap. (MW)	1.800	2.400	13.800	15.000
Decentralized cap. (MW)	1.200	1.600	10.000	9.200
Centralized cost (DKK/MWh)	500	450	400	300
Decentralized cost (DKK/MWh)	760	760	700	700
Average demand (MWh)	2299	1526	15275	14369

Table 5.1: Generation capacity and production costs.

5.5.2 Investments

The parametric programming approach to decomposition and the heuristic have been implemented in R using the interfaces by (Berkelaar, 2015) to solve LPs and (Robere, 2015) for vertex enumeration. The software is open source and free. The MPEC and MINLP have been implemented in GAMS (GAMS, 2017) and solved using the DICOPT solver. All problems are solved on an HP ProLiant server with 4 AMD 2.50 GHz CPUs and with a total of 64 cores and 256 GB RAM.

For the most detailed case with 1000 representative hours, the optimal solution is

$$(x_{N,DK1}, x_{N,DK2}, x_{SE,DK1}) = (1, 1, 1)$$
$$(\mathcal{F}_{N,DK1}, \mathcal{F}_{N,DK2}, \mathcal{F}_{SE,DK1}) = (1000, 1000, 16),$$

meaning investment are made in all candidate lines with maximum capacity on $(N, DK1)$ and $(N, DK2)$ and investments of 16 MW capacity on $(SE, DK1)$. We use this solution as benchmark.

The investments in all candidate lines are justified by total congestion rents offsetting the investment costs. In fact, the transmission of power and differences in nodal prices generate significant revenues for the merchant investor. We explain this as follows.

Since the costs of centralized generation is significantly lower than those of decentralized generation, demand is satisfied by central production unless generation capacity is binding. As the production costs of Norway and Sweden are lower than those of the Danish nodes, demand of all nodes is satisfied by central production in Norway and Sweden, using both existing and newly constructed transmission lines. Thus, power is transmitted from Norway and Sweden to Denmark unless transmission capacities are binding, i.e. congestion occurs. As a result, the nodal prices are determined by the marginal costs of centralized Norwegian and Swedish generation in many of the representative hours. When congestion occurs, however, market prices of the Danish nodes are higher than for Norway and Sweden.

Average nodal prices are given in Table 5.2. As expected, average prices are higher for the Danish nodes than for the Norwegian and Swedish nodes, but the same for the Danish nodes.

DK1	DK2	SE	N
491	491	451	357

Table 5.2: Average prices at the four nodes in DKK/MWh

In Table 5.3, we list the number of hours (out of the 1000 representative hours) for which the transmission line are congested. Furthermore, the direction of the power flow is indicated by the number of hours with positive and negative flow. We note that power

	(N,SE)	(N,DK1)	(N,DK2)	(SE,DK1)	(SE,DK2)	(DK1,DK2)
Congested Lines	947	980	997	1000	812	0
Positive Flow	981	990	1000	1000	1000	90
Negative Flow	19	10	0	0	0	907

Table 5.3: Number of hours (our of 1000) with congested lines, positive flow and negative flow on all transmission lines.

always flows into the Danish nodes from the $(N, DK1)$, $(S, DK1)$ and $(S, DK2)$ lines, clearly confirming the low-cost generation from Norway and Sweden supplied to the Danish market. The (N, S) and $(N, DK1)$ lines mainly have flow from Norway to DK1 and Sweden (in 981 and 990 out of 1000 hours, respectively). All but the $(DK1, DK2)$ line have power flow during all hours and the $(DK1 - DK2)$ line only have 3 out of 1000 hours without flow of power. Thus, the markets exploits the network at all times.

As can be seen, the line connecting Sweden and DK1 is always congested, meaning the merchant investor collects congestion rents in all hours. Also, the transmission lines connecting Norway and Sweden, Norway and DK1, Norway and DK2 and Sweden and

DK2 are almost always congested (between 812 and 997 hours out of the 1000 representative hours). The only line that is never congested is the one connecting the two Danish regions, DK1 and DK2.

5.5.3 Solution Methods

We suggested two solution methods based on parametric programming: The parametric programming approach to decomposition (abbreviated Decomp.) guaranteeing global optimality and the heuristic. We compare with the tree non-linear programming methods: A standard MPEC solver, a regularization approach (Reg. MPEC) and a reformulation by disjunctive constraints (MINLP). We solve the BPP with all these methods, varying the number of representative hours by 10 from 10 to 1000, the result of which is a total of 19 problem instances of increasing size.

The standard MPEC solver returned local infeasibility for all instances, and thus, we do not report further results of using this solution method. The MINLP method likewise did not provide any results, with the solver reporting that the search stopped as the objective function of the NLP subproblems started to deteriorate. While the regularization approach returned local optimal solutions for all 19 instances, all these solutions were

$$(x_{N,DK1}, x_{N,DK2}, x_{SE,DK1}) = (0, 0, 0)$$
$$(\mathcal{F}_{N,DK1}, \mathcal{F}_{N,DK2}, \mathcal{F}_{SE,DK1}) = (0, 0, 0),$$

i.e. no investments were made. This results in an optimality gap of 99% and is of no practical use.

To compare the solutions of the decomposition approach and the heuristic, we report the investment capacities of the three candidate lines in Table 5.4. We see that the two solution methods agree in 14 out of 19 cases, as also indicated by the optimality gap in Table 5.5. For both methods, investments are made in lines $(N, DK1)$ and $(N, DK2)$ at maximum capacity in all but one instance (50 representative days). The investment in line $(SE, DK1)$ is of a smaller capacity, although in many instances (14 and 11 out of 19 for the decomposition approach and the heuristic, respectively), some investment is profitable. In fact, a small capacity is enough to create congestion and generate some revenue. The two methods, however, occasionally disagree regarding investment in line $(SE, DK1)$. We conclude that the heuristic does well in terms of detecting congested lines, but fails to obtain optimal investment capacities when these are small. The larger the instances, the worse the heuristic solutions. In fact, for 600-1000 representative days, the exact approach justifies investment a low-capacity line, whereas in four out of five instances, the heuristic does not invest at all.

Table 5.5 provides the solution times of the exact parametric programming approach and the heuristic and their differences in objective function values, i.e. optimality gaps. For a number of representative days higher than 500, the optimality gaps produced by the heuristic varies from 0.1% to 10.1%. When the number of representative days is 500 or lower, the heuristic obtains the optimal solution. Since this is a heuristic, we can only attribute this to the model structure and data. For instances with 100 representative days or lower, the heuristic obtains the optimal solution 15-70 times as fast. For problems with 200 representative days or more, the heuristic maintains lower solution time for almost all instances but with a factor between 4 and 7. While the heuristic provides no guarantees of optimality, our case study suggests that for small to moderate sized bilevel problems, it works very well. Furthermore, it solves even large problems relatively fast

No. of rep. days	Decomp L1	Decomp L2	Decomp L3	Heuristic L1	Heuristic L2	Heuristic L3
10	1000	1000	275	1000	1000	275
20	1000	1000	126	1000	1000	126
30	1000	1000	33	1000	1000	33
40	1000	1000	128	1000	1000	128
50	1000	994	0	994	1000	0
60	1000	1000	77	1000	1000	77
70	1000	1000	0	1000	1000	0
80	1000	1000	0	1000	1000	0
90	1000	1000	51	1000	1000	51
100	1000	1000	11	1000	1000	11
200	1000	1000	38	1000	1000	38
300	1000	1000	38	1000	1000	62
400	1000	1000	0	1000	1000	0
500	1000	1000	0	1000	1000	4
600	1000	1000	9	769	1000	0
700	1000	1000	10	913	1000	0
800	1000	1000	11	1000	1000	4
900	1000	1000	16	993	1000	0
1000	1000	1000	16	901	1000	0

Table 5.4: Investment decisions from the two solution methods, the decomposition approach (Decomp.) and the heuristic (Heuristic). All numbers in MW.

No. of rep. days	Sol. time decomp. (s)	Sol. time heuristic (s)	Optimality Gap (%)
10	150.7	2.2	0
20	134.7	3.6	0
30	177.6	6	0
40	201.1	9.1	0
50	363.8	15.5	0
60	373.3	16.3	0
70	378.9	21.7	0
80	421.8	28.2	0
90	537.1	34	0
100	838.3	40.7	0
200	1013.9	143	0
300	1182.5	305.5	0
400	2308.9	528.2	0
500	3391	818.7	0.1
600	6764.8	1146.4	1.7
700	9142.5	1587.8	7.8
800	10222	2567	0
900	13543	3215.5	10.1
1000	16733.8	4238.5	7

Table 5.5: Solution times and optimality gaps.

and provides solutions within a 10% optimality gap. Its main disadvantage is that the solutions may be structurally different from the optimal, and thus, it may be better suited for cost assessments than for investment planning.

5.6 Conclusion

This chapter adopts a merchant investor perspective on transmission expansion. Investment is incentivized by the merchant collecting congestion rents on the installed transmission lines. We formulate a bilevel programming problem in which investment decisions are made in an upper level and in anticipation of lower-level market-clearing. With the inclusion of congestion rents, the formulation involves a bilinear revenue term in the upper-level objective function. This makes the problem difficult to solve to global optimality by standard approaches, such as MPEC or MILP reformulations.

Instead, we present an exact algorithm based on parametric programming that solves the bilinear bilevel programming problem to global optimality. Furthermore, it allows for decomposition of the lower-level problem and thereby has potential to provide computational advantages. We also present a faster, but heuristic version of the algorithm.

We illustrate the problem and the solution methods on a case study of the Nordic region. The numerical results demonstrates the profitability of being a merchant investor in a transmission network. The parametric programming approach is able to solve problem instances with up to 1000 representative days within 4.5 hours while the heuristic terminates in 1.2 hours and with an optimality gap of 7%. The heuristic found the optimal solution in 14 out of 19 cases with significantly lower solution time than the parametric approach. For large instances, however, the structure of the solutions produced by the heuristic may differ from the optimal.

Bibliography

Akbari, T., Rahimi-Kian, A., and Tavakoli Bina, M. (2012). Security-constrained transmission expansion planning: A stochastic multi-objective approach. *International Journal of Electrical Power and Energy Systems*, 43(1):444–453.

Aneiros, G., Vilar, J., and Raña, P. (2016). Short-term forecast of daily curves of electricity demand and price. *International Journal of Electrical Power and Energy Systems*, 80:96–108.

Avis, D. (2000). lrs: A Revised Implementation of the Reverse Search Vertex Enumeration Algorithm. *DMV Seminar*, 29:177–198.

Avis, D. and Fukuda, K. (1996). Reverse search for enumeration. *Discrete Applied Mathematics*, 65(1-3):21–46.

Babrowski, S., Jochem, P., and Fichtner, W. (2015). Electricity storage systems in the future German energy sector. *Computers & Operations Research*, 66:1–13.

Bard, J. F. (1991). Some properties of the bilevel programming problem. *Journal of Optimization Theory and Applications*, 68(2):371–378.

Bard, J. F. (1998). *Practical Bilevel Optimization: Algorithms and Applications*. Springer.

Bard, J. F. and Falk, J. E. (1982). An explicit solution to the multi-level programming problem. *Computers & Operations Research*, 9(1):77–100.

Bard, J. F. and Moore, J. T. (1990). A Branch and Bound Algorithm for the Bilevel Programming Problem. *SIAM Journal on Scientific and Statistical Computing*, 11(2):281–292.

Baringo, L. and Conejo, A. A. J. (2011). Wind power investment within a market environment. *Applied Energy*, 88(9):3239–3247.

Baringo, L. and Conejo, A. J. (2012a). Transmission and wind power investment. *IEEE Transactions on Power Systems*, 27(2):885–893.

Baringo, L. and Conejo, A. J. (2012b). Wind Power Investment: A Benders Decomposition Approach. *IEEE Transactions on Power Systems*, 27(1):433–441.

Baringo, L. and Conejo, A. J. (2013). Correlated wind-power production and electric load scenarios for investment decisions. *Applied Energy*, 101:475–482.

Baringo, L. and Conejo, A. J. (2014). Strategic Wind Power Investment. *IEEE Transactions on Power Systems*, 29(3):1250–1260.

Ben-Ayed, O. and Blair, C. E. (1990). Computational Difficulties of Bilevel Linear Programming. *Operations Research*, 38(3):556–560.

Berkelaar, M. (2015). *lpSolve: Interface to 'Lp_solve' v. 5.5 to Solve Linear/Integer Programs*.

Bertsch, V. and Fichtner, W. (2015). A participatory multi-criteria approach for power generation and transmission planning. *Annals of Operations Research*, pages 177–207.

Bertsimas, D. P. and Tsitsiklis, J. N. (1997). *Introduction to Linear Optimization*. Athena Scientific.

Bialas, W. F. and Karwan, M. H. (1984). Two-Level Linear Programming. *Management Science*, 30(8):1004–1020.

Birge, J. R. and Louveaux, F. F. (2011). *Introduction to Stochastic Programming*. Springer, New York, NY, USA, second edition.

Bludszuweit, H., Dominiguez-Navarro, J. A., and Llombart, A. (2008). Statistical Analysis of Wind Power Forecast Error. *Power Systems, IEEE Transactions on*, 23(3):983–991.

Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. (1994). *Time Series Analysis: Forecasting {&} Control*. John Wiley {&} Sons, Ltd, 3rd edition.

Boyd, S. and Vandenberghe, L. (2009). *Convex optimization*. Cambridge university press.

Buijs, P. and Belmans, R. (2012). Transmission investments in a multilateral context. *IEEE Transactions on Power Systems*, 27(1):475–483.

Bylling, H. C., Gabriel, S. A., and Boomsma, T. K. (2018). A Parametric Programming Approach to Bilevel Optimization with lower-level Variables in the upper Level. *wip*.

Calvete, H. I., Galé, C., and Mateo, P. M. (2008). A new approach for solving linear bilevel problems using genetic algorithms. *European Journal of Operational Research*, 188(1):14–28.

Candler, W. and Townsley, R. (1982). A linear two-level programming problem. *Computers & Operations Research*, 9(1):59–76.

Chaton, C. and Doucet, J. (2003). Uncertainty and Investment in Electricity Generation with an Application to the Case of Hydro-Quebec. *Annals of Operations Research*, 120:59–80.

Chen, Y., Hobbs, B. F., Leyffer, S., and Munson, T. S. (2006). Leader-follower equilibria for electric power and NOx allowances markets. *Computational Management Science*, 3(4):307–330.

Colson, B., Marcotte, P., and Savard, G. (2005a). A trust-region method for nonlinear bilevel programming: Algorithm and computational experience. *Computational Optimization and Applications*, 30:211–227.

Colson, B., Marcotte, P., and Savard, G. (2005b). Bilevel programming: A survey. *4or*, 3(2):87–107.

Colson, B., Marcotte, P., and Savard, G. (2007). An overview of bilevel optimization. *Annals of Operations Research*, 153(1):235–256.

Conejo, A. J., Baringo, L., Kazempour, S. J., and Siddiqui, A. S. (2016). *Investment in Electricity Generation and Transmission*. Springer.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to algorithms*. MIT press.

Danish District Heating Association and Danish Energy Association (2016). Indspil til strategi for fremtidens kraftvarme. Technical report, Danish Energy Association.

Danish Energy Agency (2012). Technology data for energy plants. *Energinet.dk*, 1(May):212.

Dempe, S. (2002). *Foundations of bilevel programming*. Springer.

Dempe, S. (2003). Annotated Bibliography on Bilevel Programming and Mathematical Programs with Equilibrium Constraints. *Optimization*, 52(3):333–359.

Dempe, S. (2018). Bilevel optimization: theory, algorithms and applications.

Dempe, S. and Dutta, J. (2010). Is bilevel programming a special case of a mathematical program with complementarity constraints? *Mathematical Programming*, 131(1-2):37–48.

Dempe, S., Dutta, J., and Mordukhovich, B. S. (2007). New necessary optimality conditions in optimistic bilevel programming. *Optimization*, 56(5-6):577–604.

Dempe, S. and Franke, S. (2014). Solution algorithm for an optimistic linear Stackelberg problem. *Computers & Operations Research*, 41:277–281.

Dempe, S., Kalashnikov, V., Pérez-Valdés, G. A., Kalashnikova, N., and Kalashnykova, N. (2015). *Bilevel Programming Problems: Theory, Algorithms and Applications to Energy Networks*. Energy Systems. Springer Berlin Heidelberg.

Dempe, S., Mordukhovich, B. S., and Zemkoho, A. B. (2014). Necessary optimality conditions in pessimistic bilevel programming. *Optimization*, 63(4):505–533.

Dempe, S. and Zemkoho, a. B. (2012). The bilevel programming problem: reformulations, constraint qualifications and optimality conditions. *Mathematical Programming*, 138(1-2):447–473.

Dua, V., Bozinis, N. A., and Pistikopoulos, E. N. (2002). A multiparametric programming approach for mixed-integer quadratic engineering problems. *Computers and Chemical Engineering*, 26(4-5):715–733.

Dupacova, J., Groewe-Kuska, N., and Roemisch, W. (2003). Scenario Reduction in Stochastic Programming: An Approach using Probability Metrics. *Mathematical Programming, Ser. A*, 95(4):493–511.

Ea Energianalyse (2014). Elproduktionsomkostninger. *Energistyrelsen*, 1(1):1–42.

Ehrenmann, A. and Smeers, Y. (2011). Generation Capacity Expansion in a Risky Environment: A Stochastic Equilibrium Analysis. *Operations Research*, 59(6):1332–1346.

ElNozahy, M. S., Salama, M. M. A., and Seethapathy, R. (2013). A probabilistic load modelling approach using clustering algorithms. In *2013 IEEE Power Energy Soc. Gen. Meet.*, pages 1–5. IEEE.

Entsoe (2016). European Network of Transmission System Operators for Electricity.

European Commision (2014). Communication from the Commission to the European Parliament, the Council, the European economic and social Committee and the Committee of the Regions - A policy framework for climate and energy in the period from 2020 to 2030. *European Commission*, page Brussels.

Faísca, N. P., Dua, V., Rustem, B., Saraiva, P. M., and Pistikopoulos, E. N. (2007). Parametric global optimisation for bilevel programming. *Journal of Global Optimization*, 38(4):609–623.

Fletcher, R. and Leyffer, S. (2002). Numerical experience with solving MPECs as NLPs. Technical report, Department of Mathematics and Computer Science, University of Dundee, Dundee.

Fletcher, R. and Leyffer, S. (2004). Solving mathematical programs with complementarity constraints as nonlinear programs. *Optimization Methods and Software*, 19(1):15–40.

Fortuny-Amat, J. and McCarl, B. (1981). A representation and economic interpretation of a two-Level programming problem. *Journal of the Operational Research Society*, 32(9):783–792.

Gabriel, S. A., Conejo, A. J., Fuller, J. D., Hobbs, B. F., and Ruiz, C. (2013). *Complementarity Modeling in Energy Markets*, volume 157 of *International Series in Operations Research & Management Science*. Springer.

Gabriel, S. A., Fuller, J. D., Shim, Y., Conejo, A. J., de la Torre, S., García-Bertrand, R., de la Torre, S., and García-Bertrand, R. (2010). A Benders Decomposition Method for Solving Stochastic Complementarity Problems with an Application in Energy. *Journal of the Operational Research Society*, 35(9):1404–1419.

Gabriel, S. A., García-Bertrand, R., Sahakij, P., and Conejo, A. J. (2006). A Practical Approach to Approximate Bilinear Functions in Mathematical Programming Problems by Using Schur's Decomposition and SOS Type 2 Variables. *The Journal of the Operational Research Society*, 57(8):995–1004.

Gabriel, S. A. and Leuthold, F. U. (2010). Solving discretely-constrained MPEC problems with applications in electric power markets. *Energy Economics*, 32(1):3–14.

Gal, T. (1995). *Postoptimal Analyses, Parametric Programming, and Related Topics*. Walter de Gruyter, New York.

GAMS (2017). General Algebraic Modeling System.

Garcés, L. P., Conejo, A. J. J., García-Bertrand, R., Romero, R., Garces, L. P., Conejo, A. J. J., Garcia-Bertrand, R., Romero, R., and Garcés, L. P. (2009). A Bilevel Approach to Transmission Expansion Planning Within a Market Environment. *IEEE Trans. Power Syst.*, 24(3):1513–1522.

Ghofrani, M., Arabali, A., Etezadi-Amoli, M., and Fadali, M. S. (2013). A framework for optimal placement of energy storage units within a power system with high wind penetration. *IEEE Transactions on Sustainable Energy*, 4(2):434–442.

Goldreich, O. (2008). Computational complexity: A conceptual perspective. *ACM Sigact News*, 39(3):35–39.

Han, J., Liu, G., and Wang, S. (2000). A new descent algorithm for solving quadratic bilevel programming problems. *Acta Mathematicae Applicatae Sinica*, 16(3):235–244.

Hansen, P., Jaumard, B., and Savard, G. (1992). New Branch-and-Bound Rules for Linear Bilevel Programming. *SIAM Journal on Scientific and Statistical Computing*, 13(5):1194–1217.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

Hasan, E., Galiana, F. D., and Conejo, A. J. (2008). Electricity Markets Cleared by Merit Order -Part I: Finding the Market Outcomes Supported by Pure Strategy Nash Equilibria. *Power Systems, IEEE Transactions on*, 23(2):361–371.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer Series in Statistics. Springer New York.

Haydt, G., Leal, V., Pina, A., and Silva, C. A. (2011). The relevance of the energy resource dynamics in the mid/long-term energy planning models. *Renewable Energy*, 36(11):3068–3074.

Hejazi, S. R., Memariani, a., Jahanshahloo, G., and Sepehri, M. M. (2002). Linear bilevel programming solution by genetic algorithm. *Computers & Operations Research*, 29(13):1913–1925.

Hemmati, R., Hooshmand, R. A., and Khodabakhshian, A. (2013). Comprehensive review of generation and transmission expansion planning. *IET Generation, Transmission and Distribution*, 7(April):955–964.

Hemmati, R., Hooshmand, R. A., and Khodabakhshian, A. (2014). Market based transmission expansion and reactive power planning with consideration of wind and load uncertainties. *Renewable and Sustainable Energy Reviews*, 29:1–10.

Hobbs, B. F., Metzler, C. B., and Pang, J.-S. J.-S. (2000). Strategic Gaming Analysis for Electric Power Systems: An MPEC Approach. *IEEE Transactions on Power Systems*, 15(2).

Hu, X. M. and Ralph, D. (2004). Convergence of a Penalty Method for Mathematical Programming with Complementarity Constraints. *Journal of Optimization Theory and Applications*, 123(2):365–390.

IBM Corp. (2015). The ILOG CPLEX.

Jabr, R. A., Džafić, I. I., and Pal, B. C. (2015). Robust optimization of storage investment on transmission networks. *IEEE Transactions on Power Systems*, 30(1):531–539.

Jenabi, M., Fatemi Ghomi, S. M. T., and Smeers, Y. (2013). Bi-Level Game Approaches for Coordination of Generation and Transmission Expansion Planning Within a Market Environment. *IEEE Transactions on Power Systems*, 28(3):2639–2650.

Jeroslow, R. G. (1985). The polynomial hierarchy and a simple model for competitive analysis. *Mathematical Programming*, 32(2):146–164.

Jiang, Y., Li, X., Huang, C., and Wu, X. (2013). Application of particle swarm optimization based on CHKS smoothing function for solving nonlinear bilevel programming problem. *Applied Mathematics and Computation*, 219(9):4332–4339.

Jin, S., Ryan, S. M., Watson, J.-P. J.-P., and Woodruff, D. L. (2011). Modeling and solving a large-scale generation expansion planning problem under uncertainty. *Energy Systems*, 2(3-4):209–242.

Joskow, P. and Tirole, J. (2005). Merchant transmission investment. *Journal of Industrial Economics*, 53(2):233–264.

Kazempour, S. J. and Conejo, A. J. (2012). Strategic generation investment under uncertainty via Benders decomposition. *IEEE Transactions on Power Systems*, 27(1):424–432.

Kazempour, S. J., Conejo, A. J., and Ruiz, C. (2011). Strategic generation investment using a complementarity approach. *IEEE Transactions on Power Systems*, 26(2):940–948.

Khanabadi, M. and Ghasemi, H. (2011). Transmission congestion management through optimal transmission switching. *IEEE Power and Energy Society General Meeting*, pages 1–5.

Koschker, S. and Möst, D. (2016). Perfect competition vs . strategic behaviour models to derive electricity prices and the influence of renewables on market power. *OR Spectrum*, 38(3):661–686.

Kristiansen, T. (2004). Markets for Financial Transmission Rights. *Energy Studies Review*, page 35.

Li, H. and Fang, L. (2012). An evolutionary algorithm for solving bilevel programming problems using duality conditions. *Mathematical Problems in Engineering*, 2012.

Liu, Y., Sioshansi, R., and Conejo, A. J. (2017). Hierarchical Clustering to Find Representative Operating Periods for Capacity-Expansion Modeling. *IEEE Trans. Power Syst.*, 99:1–8.

Liu, Y.-H. and Spencer, T. H. (1995). Solving a bilevel linear program when the inner decision maker controls few variables. *European Journal of Operational Research*, 81(3):644–651.

Lorenczik, S., Malischek, R., and Trüby, J. (2014). Modeling strategic investment decisions in spatial markets. Technical Report 14/09, University of Cologne, Köln.

Ludig, S., Haller, M., Schmid, E., and Bauer, N. (2011). Fluctuating renewables in a long-term climate change mitigation strategy. *Energy*, 36(11):6674–6685.

Luo, Z.-Q., Pang, J.-S., and Ralph, D. (1996). *Mathematical programs with equilibrium constraints*. Cambridge University Press.

Lv, Y., Hu, T., Wang, G., and Wan, Z. (2007). A penalty function method based on Kuhn-Tucker condition for solving linear bilevel programming. *Applied Mathematics and Computation*, 188(1):808–813.

Ma, J., Silva, V., Belhomme, R., Kirschen, D. S., and Ochoa, L. F. (2013). Evaluating and planning flexibility in sustainable power systems. *IEEE Transactions on Sustainable Energy*, 4(1):200–209.

Manas, M. and Nedoma, J. (1968). Finding All Vertices of a Convex Polyhedron. *Numerische Mathematik*, 12:226–229.

Maurovich-Horvat, L., Boomsma, T. K., Fleten, S.-E. S.-E. E., Siddiqui, A. S., Fleten, S.-E. S.-E. E., and Siddiqui, A. S. (2014). Transmission and wind investment in a deregulated electricity industry. *IEEE Transactions on Power Systems*, PP(99):1–7.

Mersha, A. G. and Dempe, S. (2006). Linear bilevel programming with upper level constraints depending on the lower level solution. *Applied Mathematics and Computation*, 180(1):247–254.

Mirrlees, J. A. (1999). The theory of moral hazard and unobservable behaviour: Part I. *The Review of Economic Studies*, 66(1):3–21.

Moiseeva, E., Hesamzadeh, M. R., and Biggar, D. R. (2015). Exercise of Market Power on Ramp Rate in Wind-Integrated Power Systems. *IEEE Transactions on Power Systems*, 30(3):1614–1623.

Morales, J. M., Pineda, S., Conejo, A. J., and Carrión, M. (2009). Scenario reduction for futures market trading in electricity markets. *IEEE Transactions on Power Systems*, 24(2):878–888.

Morales, J. M., Zugno, M., Pineda, S., and Pinson, P. (2014). Electricity market clearing with improved scheduling of stochastic production. *European Journal of Operational Research*, 235(3):765–774.

Motto, A. L. L., Arroyo, J. M. M., and Galiana, F. D. D. (2005). A Mixed-Integer LP Procedure for the Analysis of Electric Grid Security Under Disruptive Threat. *IEEE Trans. Power Syst.*, 20(3):1357–1365.

Murphy, F. H. and Smeers, Y. (2005). Generation Capacity Expansion in Imperfectly Competitive Restructured Electricity Markets. *Operations Research*, 53(4):646–661.

Nahmmacher, P., Schmid, E., Hirth, L., and Knopf, B. (2016). Carpe diem: A novel approach to select representative days for long-term power system modeling. *Energy*, 112:430–442.

Nord Pool AS (2017). Nordpool.

Orfanos, G. A., Georgilakis, P. S., and Hatziargyriou, N. D. (2013). Transmission expansion planning of systems with increasing wind power integration. *IEEE Transactions on Power Systems*, 28(2):1355–1362.

Outrata, J. (2000). On mathematical programs with complementarity constraints. *Optimization Methods and Software*, 14(1):117–137.

Outrata, J. V. (1988). A note on the usage of nondifferentiable exact penalties in some special optimization problems. *Kybernetika*, 24(4):251–258.

Pfenninger, S., Hawkes, A., and Keirstead, J. (2014). Energy systems modeling for twenty-first century energy challenges. *Renewable and Sustainable Energy Reviews*, 33:74–86.

Pina, A., Silva, C. A., and Ferrão, P. (2013). High-resolution modeling framework for planning electricity systems with high penetration of renewables. *Applied Energy*, 112:215–223.

Pineda, S., Bylling, H. C., and Morales, J. M. (2017). Efficiently solving linear bilevel programming problems using off-the-shelf optimization software. *Optimization and Engineering*.

Pineda, S. and Morales, J. M. (2016). Capacity expansion of stochastic power generation under two-stage electricity markets. *Computers and Operations Research*, 70:101–114.

Pineda, S., Morales, J. M., Ding, Y., and Stergaard, J. (2014). Impact of equipment failures and wind correlation on generation expansion planning. *Electric Power Systems Research*, 116:451–458.

Pisciella, P., Bertocchi, M., and Vespucci, M. T. (2014). A leader-followers model of power transmission capacity expansion in a market driven environment. *Computational Management Science*, pages 1–32.

Poncelet, K., Delarue, E., Six, D., Duerinck, J., and D'haeseleer, W. (2016). Impact of the level of temporal and operational detail in energy-system planning models. *Applied Energy*, 162:631–643.

Poncelet, K., Hoschle, H., Delarue, E., Virag, A., D'haeseleer, W., and Drhaeseleer, W. (2017). Selecting Representative Days for Capturing the Implications of Integrating Intermittent Renewables in Generation Expansion Planning Problems. *IEEE Trans. Power Syst.*, 32(3):1936–1948.

Pozo, D. and Contreras, J. (2011). Finding Multiple Nash Equilibria in Pool-Based Markets: A Stochastic EPEC Approach. *IEEE Trans. Power Syst.*, 26(3):1744–1752.

Pozo, D., Sauma, E., and Contreras, J. (2013). A Three-Level Static MILP Model for Generation and Transmission Expansion Planning. *IEEE Transactions on Power Systems*, 28(1):202–210.

Pritchard, G., Zakeri, G., and Philpott, A. (2010). A Single-Settlement, Energy-Only Electric Power Market for Unpredictable and Intermittent Participants. *Operations Research*, 58(4-Part-2):1210–1219.

Ralph, D. and Wright, S. J. (2004). Some properties of regularization and penalization schemes for MPECs. *Optimization Methods and Software*, 19(5):527–556.

Robere, R. (2015). *vertexenum: Vertex Enumeration of Polytopes*.

Römisch, W. (2002). Scenred2. *GAMS Development Corporation*.

Rosellón, J., Kristiansen, T., Rosellon, J., and Kristiansen, T. (2013). *Financial Transmission Rights Analysis, Experiences and Prospects*. Lecture Notes in Energy. Springer.

Rosenthal, R. E. (2014). GAMS A User ' s Guide. *GAMS Development Corporation*, 1(September).

Ruiz, C. and Conejo, A. J. (2009). Pool strategy of a producer with endogenous formation of locational marginal prices. *IEEE Transactions on Power Systems*, 24(4):1855–1866.

Ruiz, C. and Conejo, A. J. (2014). Robust Transmission Expansion Planning. *European Journal of Operational Research*, 242(2):390–401.

Ruiz, C., Conejo, A. J., and Smeers, Y. (2012). Equilibria in an Oligopolistic Electricity Pool With Stepwise Offer Curves. *IEEE Trans. Power Syst.*, 27(2):752–761.

Saharidis, G. K. D., Conejo, A. J., and Kozanidis, G. (2013). Exact solution methodologies for Linear and (Mixed) Integer Bilevel Programming. In *Metaheuristics for Bi-level Optimization*, volume 482, chapter 8, pages 221–245. Springer.

Scheel, H. and Scholtes, S. (2000). Mathematical Programs with Complementarity Constraints: Stationarity, Optimality, and Sensitivity. *Mathematics of Operations Research*, 25(1):1–22.

Scholtes, S. (2001). Convergence Properties of a Regularization Scheme for Mathematical Programs with Complementarity Constraints. *SIAM Journal on Optimization*, 11(4):918–936.

Schröder, A., Kunz, F., Meiss, J., Mendelevitch, R., von Hirschhausen, C., von Hirschhausen, C., von Hirschhausen, C., and von Hirschhausen, C. (2013). Current and Prospective Costs of Electricity Generation until 2050 2013. *Deutsches Institut für Wirtschaftsforschung*.

Schwarz, H., Bertsch, V., and Fichtner, W. (2018). Two-stage stochastic, large-scale optimization of a decentralized energy system: a case study focusing on solar PV, heat pumps and storage in a residential quarter. *OR Spectrum*, 40(1):265–310.

Shi, C., Lu, J., and Zhang, G. (2005a). An extended Kth-best approach for linear bilevel programming. *Applied Mathematics and Computation*, 164(3):843–855.

Shi, C., Lu, J., and Zhang, G. (2005b). An extended Kuhn-Tucker approach for linear bilevel programming. *Applied Mathematics and Computation*, 162(1):51–63.

Shi, C., Lu, J., Zhang, G., and Zhou, H. (2006). An extended branch and bound algorithm for linear bilevel programming. *Applied Mathematics and Computation*, 180(2006):529–537.

Shi, C., Zhang, G., and Lu, J. (2005c). On the definition of linear bilevel programming solution. *Applied Mathematics and Computation*, 160(1):169–176.

Siddiqui, S. and Gabriel, S. A. (2013). An SOS1-Based Approach for Solving MPECs with a Natural Gas Market Application. *Networks and Spatial Economics*, 13(2):205–227.

Sinha, A., Malo, P., and Deb, K. (2013). Efficient Evolutionary Algorithm for Single-Objective Bilevel Optimization. *arXiv*, 1303.3901.

Sisternes, F. J. D., Webster, M. D., Sisternes, F. J. D., and Webster, M. D. (2013). Optimal Selection of Sample Weeks for Approximating the Net Load in Generation Planning Problems. *ESD Working Paper Series*, 1(January).

Slednev, V., Bertsch, V., and Fichtner, W. (2017). A Multi-objective Time Segmentation Approach for Power Generation and Transmission Models. In Dörner, K. F., Ljubic, I., Pflug, G., and Tragler, G., editors, *Operations Research Proceedings 2015*, pages 707–714, Cham. Springer International Publishing.

Sorokin, A., Rebennak, S., Pardalos, P., and Iliadis, N. A. (2012). *Handbook of networks on power systems*. Springer.

Stackelberg, H. V. (1934). *Market Structure and Equilibrium*. Springer.

Stoft, S. (2002). Power system economics. *Journal of energy litterature*, 8:94–99.

Strekalovsky, A. S., Orlov, A. V., and Malyshev, A. V. (2010a). Numerical solution of a class of bilevel programming problems. *Numerical Analysis and Applications*, 3(2):165–173.

Strekalovsky, A. S., Orlov, A. V., and Malyshev, A. V. (2010b). On computational search for optimistic solutions in bilevel problems. *Journal of Global Optimization*, 48(1):159–172.

The Danish Government (2013). The Danish Climate Policy Plan. *The Ministry of Climate, Energy and Building*, 1(August):61.

Valinejad, J. and Barforoushi, T. (2015). Generation expansion planning in electricity markets: A novel framework based on dynamic stochastic MPEC. *International Journal of Electrical Power & Energy Systems*, 70:108–117.

Vicente, L. N. and Calamai, P. H. (1994). Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization*, 5(3):291–306.

Wang, J., Shahidehpour, M., Li, Z., and Botterud, A. (2009). Strategic generation capacity expansion planning with incomplete information. *Power Systems, IEEE Transactions on*, 24(2):1002–1010.

White, D. J. and Anandalingam, G. (1993). A penalty function approach for solving bi-level linear programs. *Journal of Global Optimization*, 3(4):397–419.

Wogrin, S., Barquín, J., Centeno, E., Barquin, J., and Centeno, E. (2013). Capacity Expansion Equilibria in Liberalized Electricity Markets: An EPEC Approach. *Power Systems, IEEE Transactions on*, 28(2):1531–1539.

Wogrin, S., Centeno, E., and Barquín, J. (2011). Generation capacity expansion in liberalized electricity markets: A stochastic MPEC approach. *IEEE Transactions on Power Systems*, 26(4):2526–2532.

Wogrin, S., Duenas, P., Delgadillo, A., and Reneses, J. (2014). A new approach to model load levels in electric power systems with high renewable penetration. *IEEE Transactions on Power Systems*, 29(5):2210–2218.

Zhang, G., Lu, J., and Gao, Y. (2015). *Multi-Level Decision Making: Models, Methods and Applications*. Intelligent Systems Reference Library. Springer Berlin Heidelberg.

Zugno, M., Morales, J. M., Pinson, P., and Madsen, H. (2013). Pool Strategy of a Price-Maker Wind Power Producer. *Power Systems, IEEE Transactions on*, 28(3):3440–3450.

Appendix A

Appendix to Chapter 2

A.1 Results tables

All results listed in two tables: one for non-zero balancing costs and one for balancing costs equal to zero.

Table A.1: Investment decisions and runtimes for the different models in the case of $c_g^+ = 0.05c_g/r_g^u$ and $c_g^- = 0.05c_g/r_g^d$.

Model	wind	coal	gas	nuclear	Runtime	TC	IC	OC	LSC	WCC	Meas. Wind Pen.
DC-365	2561	935	747	1417	217	990.22	666.45	311.59	4.41	7.76	0.3
DS-365	2562	983	796	1354	30859	989.68	664.57	315.65	2.66	6.8	0.3
HC-240	2575	259	963	1971	0	1033.85	690.07	322.76	1.26	19.76	0.3
HC-480	2550	276	979	1978	0	1029.83	690.81	321.24	0.72	17.05	0.298
HC-720	2557	262	977	1994	0	1032.87	692.44	321.73	0.76	17.95	0.298
HC-960	2564	270	972	1990	1	1032.87	693.34	320.79	0.75	17.98	0.299
HC-1200	2567	271	965	1989	1	1033.06	693.23	320.63	0.83	18.38	0.299
HC-1440	2566	271	965	1989	1	1033.03	693.19	320.64	0.83	18.37	0.299
HC-1680	2566	271	965	1989	2	1033.01	693.17	320.65	0.83	18.37	0.299
HC-1920	2566	266	965	1994	2	1033.71	693.41	320.88	0.83	18.58	0.299
HC-2160	2566	266	965	1995	3	1033.8	693.45	320.91	0.83	18.61	0.299
HC-2400	2567	267	964	1995	3	1033.76	693.52	320.8	0.83	18.61	0.299
HC-8760	2559	241	902	1929	32	1035.44	676.94	326.48	6.94	25.08	0.298
HS-240	2693	383	976	1796	2	1031.47	692.38	317.77	2.01	19.31	0.314
HS-480	2721	344	955	1856	8	1040.91	699.57	316.61	2.07	22.65	0.317
HS-720	2695	346	967	1842	18	1036.38	695.16	318.14	2.05	21.03	0.314
HS-960	2670	335	970	1844	22	1033.81	691.27	319.92	2.25	20.37	0.311
HS-1200	2656	328	944	1861	33	1033.75	690.09	319.59	2.88	21.19	0.31
HS-1440	2621	335	933	1857	74	1028.56	685.26	319.98	3.27	20.05	0.306
HS-1680	2613	338	919	1847	94	1027.59	682.42	320.39	4.37	20.41	0.305
HS-1920	2593	340	902	1846	135	1025.64	679.14	320.57	5.42	20.51	0.302
HS-2160	2594	333	935	1847	109	1024.82	680.31	321.6	3.72	19.19	0.302
HS-2400	2594	331	941	1845	134	1024.89	680.23	321.98	3.61	19.07	0.303
HS-8760	2559	339	939	1841	1829	1019.59	675.99	322.4	3.54	17.66	0.299
DC-10	2653	843	420	1511	0	1091.08	665.52	300.38	101.18	23.99	0.309
DC-20	2652	925	447	1402	1	1085.5	659.12	306.6	100.91	18.87	0.31
DC-30	2685	871	732	1464	1	1006.98	681.19	306.31	6.78	12.71	0.314
DC-40	2652	870	779	1420	2	1002.21	672.61	311.89	6.7	11.01	0.31
DC-50	2590	876	790	1404	3	994.71	663.82	315.31	6.63	8.96	0.303
DC-60	2576	910	763	1396	4	992.77	663.2	314.54	6.61	8.41	0.301
DC-70	2566	892	766	1413	5	991.84	662.59	314.28	6.6	8.36	0.3
DC-80	2552	908	763	1400	8	990.11	660.54	315.11	6.59	7.87	0.299
DC-90	2562	904	746	1420	9	991.44	663.57	312.94	6.6	8.33	0.3
DC-100	2561	911	735	1424	12	991.32	664.27	312.12	6.6	8.34	0.3
DS-10	2654	904	537	1436	6	1034.79	666.77	306.22	45.8	16.01	0.31
DS-20	2652	944	551	1383	26	1033.37	663.53	309.48	45.74	14.62	0.31
DS-30	2686	926	780	1382	77	1004.84	677.28	311.25	5.06	11.25	0.314
DS-40	2653	933	816	1338	243	1000.73	669.19	316.73	5.08	9.72	0.31
DS-50	2591	939	830	1327	450	993.5	661.17	319.93	4.6	7.8	0.303
DS-60	2576	959	808	1329	455	991.76	660.66	319.08	4.62	7.4	0.302
DS-70	2566	945	809	1341	631	990.67	659.89	318.86	4.63	7.29	0.3
DS-80	2552	961	794	1333	825	989.32	657.79	319.37	5.19	6.97	0.299
DS-90	2563	959	786	1344	1017	990.43	660.07	317.87	5.18	7.31	0.3
DS-100	2562	962	779	1347	2094	990.28	660.41	317.39	5.18	7.3	0.3

Table A.2: Investment decisions and runtimes for the different models in the case of zero balancing costs for all units.

Model	wind	coal	gas	nuclear	Runtime	TC	IC	OC	LSC	WCC	Meas. Wind Pen.
DC-365	2561	935	747	1417	88	981.02	666.45	303.41	4.4	6.75	0.3
DS-365	2561	950	777	1406	24512	980.7	667.92	303.91	2.67	6.2	0.3
HC-240	2575	259	963	1971	0	1020.68	690.07	311.68	1.04	17.9	0.3
HC-480	2550	276	979	1978	0	1016.82	690.81	310.27	0.56	15.18	0.298
HC-720	2557	262	977	1994	0	1019.75	692.44	310.68	0.57	16.06	0.298
HC-960	2564	270	972	1990	0	1019.73	693.34	309.71	0.57	16.11	0.299
HC-1200	2567	271	965	1989	1	1019.9	693.23	309.53	0.63	16.52	0.3
HC-1440	2566	271	965	1989	1	1019.88	693.19	309.54	0.63	16.51	0.3
HC-1680	2566	271	965	1989	1	1019.85	693.17	309.55	0.63	16.51	0.299
HC-1920	2566	266	965	1994	1	1020.53	693.41	309.77	0.63	16.71	0.3
HC-2160	2566	266	965	1995	2	1020.61	693.45	309.79	0.63	16.74	0.3
HC-2400	2567	267	964	1995	2	1020.57	693.52	309.68	0.63	16.74	0.3
HC-8760	2559	241	902	1929	32	1022.39	676.94	315.62	6.55	23.28	0.298
HS-240	2693	295	974	1886	1	1028.88	696.45	309.56	1.91	20.97	0.314
HS-480	2721	250	949	1956	4	1040.62	704.28	308.73	2.05	25.56	0.317
HS-720	2695	247	965	1943	9	1036.3	699.67	310.89	2	23.74	0.314
HS-960	2670	241	967	1941	14	1033.38	695.66	312.57	2.18	22.96	0.311
HS-1200	2656	232	943	1958	23	1034.04	694.39	312.49	2.82	24.34	0.309
HS-1440	2621	238	932	1955	33	1028.95	689.58	312.92	3.19	23.26	0.305
HS-1680	2613	244	917	1942	46	1027.67	686.66	313.01	4.28	23.72	0.304
HS-1920	2593	247	900	1941	55	1025.87	683.39	313.09	5.35	24.04	0.302
HS-2160	2594	242	932	1941	74	1024.61	684.61	314.24	3.63	22.14	0.302
HS-2400	2594	237	939	1941	98	1024.94	684.5	314.88	3.52	22.03	0.302
HS-8760	2559	244	937	1937	1162	1019.49	680.24	315.38	3.45	20.43	0.298
DC-10	2653	843	420	1511	0	1080.53	665.52	290.84	100.97	23.21	0.31
DC-20	2652	925	447	1402	1	1075.97	659.12	298.05	100.62	18.18	0.31
DC-30	2685	871	732	1464	1	996.84	681.19	297.24	6.72	11.7	0.314
DC-40	2652	870	779	1420	1	992.59	672.61	303.31	6.68	9.99	0.31
DC-50	2590	876	790	1404	2	985.49	663.82	307.11	6.61	7.95	0.303
DC-60	2576	910	763	1396	2	983.66	663.2	306.44	6.6	7.42	0.302
DC-70	2566	892	766	1413	3	982.64	662.59	306.09	6.59	7.37	0.3
DC-80	2552	908	763	1400	5	981.07	660.54	307.09	6.57	6.87	0.299
DC-90	2562	904	746	1420	5	982.2	663.57	304.7	6.58	7.34	0.3
DC-100	2561	911	735	1424	6	982.05	664.27	303.86	6.58	7.34	0.3
DS-10	2653	868	505	1505	3	1026.69	671.44	292.4	45.59	17.26	0.31
DS-20	2652	918	531	1428	13	1024.54	666.52	297.57	45.49	14.97	0.31
DS-30	2685	903	749	1436	30	995.55	681.26	298.4	5.05	10.84	0.314
DS-40	2652	902	792	1394	68	991.55	672.9	304.28	5.08	9.29	0.31
DS-50	2590	910	809	1377	115	984.47	664.4	308.23	4.61	7.23	0.303
DS-60	2576	933	787	1376	303	982.77	663.85	307.5	4.62	6.81	0.302
DS-70	2566	918	788	1389	260	981.69	663.1	307.25	4.63	6.71	0.3
DS-80	2552	935	772	1381	329	980.39	661.09	307.73	5.18	6.39	0.299
DS-90	2562	931	764	1392	412	981.47	663.3	306.26	5.17	6.73	0.3
DS-100	2561	936	757	1395	536	981.32	663.66	305.77	5.17	6.72	0.3

Appendix B

Appendix to Chapter 3

B.1 Nomenclatures of the case studies

B.1.1 Case study I: Investment in production capacity

Indices

t	Operating time periods.
g	Producers/generators
i, j	Transmission nodes

Parameters

C_i	Annualized investment costs of the strategic investor at node i [\$/MWh].
c_i^g	Production cost of the strategic investor or its rivals at node i[\$/MWh].
ρ_t	Weight of time period t [hours].
x^{max}	Total maximum capacity allowed by the strategic investor [MW].
d_{it}	Demand in time period t at node i [MWh].
$y_i^{max,g}$	Capacity of the strategic investor or its rivals at node i [MW].
B_{ij}	Subsceptance of the transmission line from node i to node j
F_{ij}^{max}	Maximum flow of the transmission line from node i to node j [MW].

Primal continuous variables

x_i	Investment capacity of the strategic producer at node i [MW].
y_{it}^g	Power produced by the strategic investor or its rivals in time period t at node i [MWh].
θ_{it}	Voltage angle in time period t at node i

Dual continuous variables

λ_{it}	Dual variable of the balancing constraint/market-clearing price in time period t at node i [\$/MWh].
μ_{it}^g	Dual variable of generation capacity constraint in time period t at node i.
η_{ijt}	Dual variable of the transmission capacity constraints between node i and node j in time period t.
α_{it}	Dual variable to the minimum nodal angle constraint in time period t at node i.
β_{it}	Dual variable to the maximum nodal angle constraint in time period t at node i.
γ_{it}	Dual variable to the reference nodal angle constraint in time period t.

B.1.2 Case study II: Transmission investment

Indices

t	Operating time periods.
g	Producers/generators
i, j	Transmission nodes
(i, j)	Transmission lines
Parameters	
C_{ij}	Annualized investment costs of the strategic investor between node i and node j [$/MWh].
ρ_t	Weight of time period t [hours].
x^{max}	Total maximum capacity allowed by the strategic investor [MW].
F_{ij}^{max}	Maximum flow of the transmission line from node i to node j [MW].
c^g	Production cost of the strategic investor or its rivals [$/MWh].
d_{it}	Demand in time period t at node i [MWh].
$y^{max,g}$	Capacity of the strategic investor or its rivals [MW].
Primal continuous variables	
x_{ij}	Investment capacity of the strategic producer between node i and node j [MW].
y_t^g	Power produced by the strategic investor or its rivals in time period t [MWh].
f_{ijt}	Flow of the transmission line from node i to node j at time period t [MWh]
θ_{it}	Voltage angle in time period t at node i
Dual continuous variables	
λ_{it}	Dual variable of the balancing constraint/market-clearing price in time period t at node i [$/MWh].
μ_t^g	Dual variable of generation capacity constraint in time period t.
η_{ijt}	Dual variable of the transmission capacity constraints between node i and node j in time period t.
α_{ijt}^g	Dual variable to the minimum flow constraint in time period t between node i and node j.
β_{ijt}^g	Dual variable to the maximum flow constraint in time period t between node i and node j.

B.2 MILP formulations

For Case study I, the MILP formulation is the following:

$$\min \sum_{i=1}^{n}\left(C_i x_i + \sum_{t=1}^{s}\rho_t\left(\sum_{g\in\mathcal{G}} c_i^g y_{it}^g - \lambda_{it}y_{it}^1 + \sum_{j\in\mathcal{J}_i} F_{ij}^{max}\eta_{ijt} + \pi(\alpha_{it}+\beta_{it})\right)\right)$$

$$\text{s.t. } 0 \le \sum_{i=1}^{n} x_i \le x^{max}$$

$$\sum_{g\in\mathcal{G}} y_{it}^g - \sum_{j\in\mathcal{J}_i} B_{ij}(\theta_{it}-\theta_{jt}) = d_{it} \;:\; \lambda_{it} \quad \forall i$$

$$0 \le y_{it}^1 \le x_i \quad \forall i$$

$$0 \le y_{it}^g \le y_i^{max,g} \quad \forall i, g \neq 1$$

$$B_{ij}(\theta_{it} - \theta_{jt}) \leq F_{ij}^{max} \quad \forall j \in \mathcal{J}_i$$
$$-\pi \leq \theta_{it} \leq \pi \quad \forall i$$
$$\theta_{it} = 0 \quad i = \text{ref}$$
$$C_i^g - \lambda_{it} + \mu_{it}^g \geq 0 \quad \forall t, i$$
$$\sum_{j \in \mathcal{J}_i} \left[B_{ij}\left(\lambda_i - \lambda_j\right) + B_{ij}\left(\eta_{ijt} - \eta_{jit}\right)\right]$$
$$-\alpha_{it} + \beta_{it} = 0 \quad \forall t, \forall i \neq \text{ref.}$$
$$\sum_{j \in \mathcal{J}_i} \left[B_{ij}\left(\lambda_i - \lambda_j\right) + B_{ij}\left(\eta_{ijt} - \eta_{jit}\right)\right]$$
$$-\alpha_{it} + \beta_{it} + \gamma_t = 0 \quad \forall t, i = \text{ref.}$$
$$x_i - y_{it}^1 \leq \delta_{it}^1 M_1 \quad \forall t, i$$
$$y^{max,g} - y_{it}^g \leq \delta_{it}^g M_1 \quad \forall t, i, g \neq 1$$
$$\mu_{it}^g \leq (1 - \delta_{it}^g) M_2 \quad \forall t, i, g$$
$$y_{it}^g \leq \nu_{it}^g M_1 \quad \forall t, i, g$$
$$c_i^g - \lambda_{it} + \mu_{it}^g \leq (1 - \nu_{it}^g) M_2 \quad \forall t, i, g$$
$$F_{ij}^{max} - B_{ij}\left(\theta_{it} - \theta_{jt}\right) \leq u_{ijt} M_1 \quad \forall t, i, j \in \mathcal{J}_i$$
$$\eta_{ijt} \leq (1 - u_{ijt}) M_2 \quad \forall t, i, j \in \mathcal{J}_i$$
$$\pi + \theta_{it} \leq v_{it} M_1 \quad \forall t, i$$
$$\alpha_{it} \leq (1 - v_{it}) M_2 \quad \forall t, i$$
$$\pi - \theta_{it} \leq w_{it} M_1 \quad \forall t, i$$
$$\beta_{it} \leq (1 - w_{it}) M_2 \quad \forall t, i$$
$$\delta_{it}^g, \nu_{it}^g, u_{ijt}, v_{it}, w_{it} \in \{0, 1\} \quad \forall t, i, g$$

Note that the constants are set to $M_1 = 1000$ and $M_2 = 5000$ as by Conejo et al. (2016).

For brevity, we leave out the MILP formulation of Case study II.

B.3 Results

B.3.1 Case study I

Table B.3 lists the solution times for all solution methods and varying the number of lower-level problems. It further lists the relative deviations in the objective function values from the optimal, for the heuristic and the MILP methods. For the full parametric programming method, no solution time is provided with more than 100 lower-level subproblems since these instances reached the time limit. This is also the case for the MILP, for which 43 out of 59 cases did not solve within the time limit of 2 hours. All running times above the time limit of 2 hours are marked as not available (N/A).

Table B.3: Solution times (in secs) and deviations (in %) in the objective function values from the optimal, for different solutions methods and as a function of the number of lower-level (LL) problems. The solution methods include the *full* parametric programming method without decomposition, parametric programming method with *decomposition*, the *heuristic* with decomposition and the *MILP*. N/A means that the method did not solve within the time limit of 2 hours.

No. of LL prob.	full time (s)	decomp. time (s)	heuristic time (s)	heuristic dev. (%)	MILP time (s)	MILP dev. (%)
10	0.57	13.84	0.92	0	0.75	0
20	10.07	16.2	0.85	16.43	0.86	0
30	42.52	19.87	1.12	11.15	0.66	0
40	129.96	28.34	1.45	7.39	0.89	0
50	398.19	16.24	1.62	7.11	1.2	0
60	752.59	21.27	2.01	3.94	0.99	0
70	1905.29	21.87	2.22	8.51	1.1	0
80	3114.58	22.94	2.5	3.91	1.38	0
90	4218.53	22.9	2.33	5.71	1.44	0
100	10288.31	25.2	2.72	5.29	3.56	0
200	N/A	45.47	3.67	4.68	5.63	0
300	N/A	73.12	7.36	3.57	129.87	0
400	N/A	93.71	11.64	4.12	73.84	0
500	N/A	142.81	18.68	3.24	N/A	2.93
600	N/A	170.73	23.44	3.86	84.64	0
700	N/A	213.11	33.32	3.73	N/A	3.34
800	N/A	273.44	43.95	3.9	108.4	0
900	N/A	314.73	58.81	3.77	N/A	3.3
1000	N/A	377.02	67.7	3.79	N/A	3.4
1100	N/A	442.21	81.43	3.78	N/A	3.31
1200	N/A	526.41	100.58	3.78	N/A	3.34
1300	N/A	604.18	117.83	3.65	N/A	3.29
1400	N/A	664.8	134.71	3.72	N/A	3.38
1500	N/A	743.72	149.28	3.85	N/A	3.38
1600	N/A	826.94	175.7	3.76	N/A	3.42
1700	N/A	962.94	214.32	3.68	N/A	3.33
1800	N/A	1020.66	216.63	3.76	N/A	3.38
1900	N/A	1114.72	244.04	3.76	60.04	0
2000	N/A	1238.18	273.74	3.77	N/A	3.4
2100	N/A	1348.78	298.37	3.66	N/A	3.35
2200	N/A	1485	342.66	3.75	N/A	3.38
2300	N/A	1603.75	364.89	3.78	N/A	3.44
2400	N/A	1717.35	385.07	3.73	N/A	3.37
2500	N/A	1818.17	420.66	3.62	N/A	3.28
2600	N/A	1936.25	447.28	3.65	N/A	3.29
2700	N/A	2054.09	486.73	3.72	N/A	3.34
2800	N/A	2215.49	523.73	3.65	N/A	3.28
2900	N/A	2320.08	551.85	3.72	N/A	3.41
3000	N/A	2536.97	606.4	3.64	N/A	3.29
3100	N/A	2666.36	640.12	3.68	N/A	3.36
3200	N/A	2834.04	678.75	3.71	N/A	3.37
3300	N/A	3000.7	734.29	3.64	N/A	3.27
3400	N/A	3195.28	772.95	3.61	N/A	3.32
3500	N/A	3352.07	819.54	3.63	N/A	3.38
3600	N/A	3513.92	863.1	3.67	N/A	3.39
3700	N/A	3670.31	898.6	3.64	N/A	3.36
3800	N/A	3905.5	963.87	3.63	N/A	3.35
3900	N/A	3981.64	995.31	3.62	N/A	3.35
4000	N/A	4221.82	1059.53	3.65	N/A	3.31
4100	N/A	4493.08	1103.64	3.63	N/A	3.48
4200	N/A	4637.07	1175.24	3.65	N/A	3.36
4300	N/A	4907.6	1225.41	3.66	N/A	3.38
4400	N/A	5103.96	1293.24	3.64	N/A	3.37
4500	N/A	5323.46	1347.49	3.65	N/A	3.41
4600	N/A	5512.33	1402.82	3.58	N/A	3.45
4700	N/A	5765.68	1468.47	3.61	N/A	3.34
4800	N/A	6065.63	1538.58	3.62	N/A	3.72
4900	N/A	6154.55	1587.39	3.61	N/A	3.84
5000	N/A	6486.67	1658.31	3.61	N/A	3.32

B.3.2 Case Study II

Table B.4 lists the solution times for decomposition by parametric programming and the parametric programming heuristic. For the exact parametric programming approach, no solution time is provided with more than 300 lower-level subproblems since these instances reached the time limit. In contrast, the heuristic solves all instances with less than

900 lower-level subproblems.

Table B.4: Solution times (in secs) and deviations (in %) in the objective function values from the optimal, for different solutions methods and as a function of the number of lower-level (LL) problems. The solution methods include the parametric programming method with *decomposition* and the *heuristic* with decomposition. N/A means that the method did not solve within the time limit of 2 hours. Deviations are not available for more than 300 lower-level problems since the exact method did not solve within the time limit.

No. of LL prob.	decomp. time (s)	heu. time (s)	heu. dev. (%)
10	255.37	2.45	0
20	291.18	4.9	0
30	308.74	7.81	0
40	475.1	15.5	0
50	520.52	32.4	0
60	565.94	28.05	0
70	677.45	35.35	0
80	946.85	56.95	0
90	1087.08	64.91	0
100	1169.27	77.56	0
200	2585.28	279.32	0
300	5230.5	636.27	0
400	N/A	1125.35	N/A
500	N/A	1658.92	N/A
600	N/A	2306.44	N/A
700	N/A	3406.25	N/A
800	N/A	4309.51	N/A
900	N/A	5122.1	N/A
1000	N/A	N/A	N/A

Appendix C

Appendix to Chapter 5

C.1 Nomenclature

Table C.1: Nomenclature

Sets	
T	set of time periods
S	set of scenarios
Ω	set of network nodes
$\Omega(i)$	set of nodes connected to node i
$\mathcal{C}$	set of candidate lines
G	set of all production units
$G(i)$	set of production units at node i
Parameters	
ρ_t	duration of time period t (p.u.)
ϕ_s	probability of scenario s (p.u.)
k_{ij}	linear investment cost for candidate line between nodes i and j (€/MW)
K_{ij}	fixed investment cost for candidate line between nodes i and j (€)
K^{max}	investment budget (€)
$\mathcal{F}_{ij}^{max}$	capacity available for candidate transmission line between nodes i and j (MW)
F_{ij}^{max}	maximum capacity of existing transmission line between nodes i and j (MW)
c_g	linear production cost for unit g (€/MWh)
y_g^{max}	maximum production for generation unit g (MW)
d_{its}	demand for time t, scenario s at node i (MW)
Variables	
x_{ij}	binary investment decision on candidate line between nodes i and j
$\mathcal{F}_{ij}$	installed capacity of candidate transmission line between nodes i and j (MW)
p_{ijts}	power flow between node i and j in time t and scenario s (MWh)
θ_{its}	voltage angle at node i in time t and scenario s
y_{gts}	production for unit g for time t and scenario s
λ_{its}	shadow price/dual variable of the balancing constraint at node i time t and scenario s (€/MWh)
μ_{gts}^{y}	dual variable of the capacity constraint of unit g time t and scenario s
$\mu_{ijts}^{p}, \mu_{ijts}^{p,\mathcal{C}}$	dual variable of the flow constraint between nodes i and j time t and scenario s
$\mu_{ijts}^{F,min}, \mu_{ijts}^{F,max}$,	dual variable of the capacity constraint between nodes i and j time t and scenario s
$\mu_{ijts}^{\mathcal{F},min}, \mu_{ijts}^{\mathcal{F},max}$	dual variable of the capacity constraint between nodes i and j time t and scenario s
$\mu_{its}^{\theta,min}, \mu_{its}^{\theta,max}, \mu_{ts}^{\theta,ref}$	dual variable of the voltage angle constraint of nodes i time t and scenario s

Note: Bold-face represent vectors of variables, e.g. $\mathbf{x} = \{x_{ij}\}_{i,j}$ and $\lambda = \{\lambda_{its}\}_{i,t,s}$.