



A parametric programming approach to bilevel optimisation with lower-level variables in the upper level

Bylling, Henrik C.; Gabriel, Steven A.; Boomsma, Trine K.

Published in:
Journal of the Operational Research Society

DOI:
[10.1080/01605682.2019.1590132](https://doi.org/10.1080/01605682.2019.1590132)

Publication date:
2020

Document version
Peer reviewed version

Document license:
[Other](#)

Citation for published version (APA):
Bylling, H. C., Gabriel, S. A., & Boomsma, T. K. (2020). A parametric programming approach to bilevel optimisation with lower-level variables in the upper level. *Journal of the Operational Research Society*, 71(5), 846-865. <https://doi.org/10.1080/01605682.2019.1590132>

A Parametric Programming Approach to Bilevel Optimization with Lower-level Variables in the Upper Level

Henrik C. Bylling^a and Steven A. Gabriel^{b,c} and Trine K. Boomsma^a

^aDepartment of Mathematical Sciences, University of Copenhagen, Denmark; ^bUniversity of Maryland, College Park, MD, USA; ^c Energy Transition Programme, Norwegian University of Science and Technology, Trondheim, Norway

ARTICLE HISTORY

Compiled May 28, 2019

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 linear. 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 the bilevel problem to a single-level mixed-integer linear program (MILP). 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.

KEYWORDS

optimization; linear programming; non-linear programming; investment;

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 provides a framework for the 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, Kalashnikov, Pérez-Valdés, and Kalashnykova (2015), Saharidis, Conejo, and Kozanidis (2013), Colson, Marcotte, and Savard (2007), Colson, Marcotte, and Savard (2005) and Bard (1998).

This paper considers 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 involves a bilinear product of lower-level primal and dual optimal solutions such as a revenue. For example, the primal and dual solutions represent quantities and market prices, respectively. Often, the lower-level problem includes a number of separate market-clearing conditions, e.g., for hours of a day or days of a year. Besides leader-follower games in economics, well-known problems of bilevel optimization or the related mathematical programming with equilibrium constraints (MPEC) are found in production and marketing, facility location, transportation design and robotics; see Luo, Pang, and Ralph (1996), Bard (1998) and Dempe (2018). In energy economics, examples of upper-level strategic decisions are generation capacities, as by Conejo, Baringo, Kazempour, and Siddiqui (2016), Kazempour, Conejo, and Ruiz (2011), Baringo and Conejo (2011) and Koschker and Möst (2016), transmission capacities; cf. Garcés, Conejo, García-Bertrand, and Romero (2009), or price-offering decisions as by Ruiz and Conejo (2009). Bilinear products of primal and dual variables are likewise found in many complementarity models and MPECs; cf. Ehrenmann and Smeers (2011), Gabriel, García-Bertrand, Sahakij, and Conejo (2006), Chen, Hobbs, Leyffer, and Munson (2006), Ruiz and Conejo (2009) and Gabriel and Leuthold (2010).

Our aim is to 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 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 piece-wise linearity of the upper-level objective function facilitates the application of parametric programming. In the spirit of Gal (1995) and Faísca, Dua, Rustem, Saraiva, and Pistikopoulos (2007), we iteratively determine neighboring critical regions and completely specify the parametric function. As a result, we reduce the bilevel problem to solving a single-level linear programming (LP) problem 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 fixed 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 MPEC with linear objective function, which can be reformulated to a mixed-integer linear program (MILP) under certain assumptions. The MPEC and MILP can be solved by standard software for non-linear or mixed-integer programming, 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 reformu-

lation is possible) and standard non-linear solvers to bilevel optimization problems.

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 MPECs) that are generally NP-hard; cf. Ben-Ayed and Blair (1990). When the lower-level problem is convex, a reformulation is obtained by replacing it 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 MPEC can be solved via an equivalent MILP or by descent algorithms or penalty function methods; see Colson et al. (2005). Another reformulation can be obtained by using the optimal value function; cf. Outrata (1988). A number of algorithms are particularly relevant for linearly constrained bilevel optimization problems. Examples include the vertex enumeration and descent method by Han, Liu, and Wang (2000) and enumeration of the basis matrices of the lower-level problem; see Liu and Spencer (1995) and Dempe (2018), which is of polynomial time. For reviews of solution methods to bilevel optimization, we refer to Vicente and Calamai (1994) and Dempe (2018) and for descent methods or penalty functions, also to Colson et al. (2005). Due to the non-linearity of the upper-level objective function, however, these methods may not directly apply to the general BPP-D and/or do not guarantee global optimality.

Closest to our work is the global optimization approach by Faísca et al. (2007) for linearly and quadratically constrained bilevel optimization problems. Like our approach, the method relies on parametric programming methods; see Gal (1995) or Dua, Bozinis, and Pistikopoulos (2002). Yet, we allow for bilinear terms of lower-level primal and dual variables in the upper-level objective, whereas Faísca et al. (2007) limit the focus to lower-level primal variables in a linear or quadratic upper-level objective function. Furthermore, no decomposition is offered.

The existing literature deals with bilinear objective terms in various ways. Certain problem structures allow for exact linearization by using optimality conditions and duality theory. This strategy can be found in many applications. For examples in energy markets; see Gabriel, Conejo, Fuller, Hobbs, and Ruiz (2013) and Conejo et al. (2016). For the specific problem of Hobbs, Metzler, and Pang (2000), the complementarity conditions of the lower-level problem allow the authors to replace the upper-level bilinear objective function by a concave quadratic function. Other approaches approximate the bilinear product via integer variables and/or logical constraints; cf. Koschker and Möst (2016) and Gabriel and Leuthold (2010), or using Schur's decomposition and special ordered sets of variables; see Gabriel et al. (2006). Whereas previous studies apply linearization of the bilinear terms to specific problems only, the present paper provides sufficient conditions for exact linearization in a general class of problems.

In many applications of bilevel optimization, the lower-level problem separates into a number of subproblems for a fixed upper-level solution. In such cases, decomposition can 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 therefore difficult. In contrast, our parametric programming approach easily extends to allow for decomposition in spite of non-convexity.

The remainder of this paper is organized as follows: In Section 3, we define and analyze the BPP-D. Section 4 presents the parametric programming approach, whereas Section 5 contains reformulation and linearization of the BPP-D. Examples and numerical results are provided in Section 6 and Section 7 concludes.

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:

$$\min \mathbf{c}^T \mathbf{x} + f(\mathbf{y}^*, \boldsymbol{\lambda}^*) \quad (1a)$$

$$\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b} \quad (1b)$$

$$\mathbf{x} \geq \mathbf{0} \quad (1c)$$

$$\mathbf{y}^* \in \operatorname{argmin}\{\mathbf{p}^T \mathbf{y} \quad (1d)$$

$$\text{s.t. } \mathbf{C}\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{e} \quad (1e)$$

$$\mathbf{y} \geq \mathbf{0}\} \quad (1f)$$

$$\boldsymbol{\lambda}^* \in \operatorname{argmax}\{\boldsymbol{\lambda}^T (\mathbf{D}\mathbf{x} + \mathbf{e}) \quad (1g)$$

$$\text{s.t. } \mathbf{C}^T \boldsymbol{\lambda} \leq \mathbf{p}\} \quad (1h)$$

The upper-level objective function (1a) involves a linear function of the upper-level variables, $\mathbf{x} \in \mathbb{R}^n$, and a function $f: \mathbb{R}^l \times \mathbb{R}^k \rightarrow \mathbb{R}$ of lower-level primal and dual optimal solutions, $\mathbf{y}^* \in \mathbb{R}^l$ and $\boldsymbol{\lambda}^* \in \mathbb{R}^k$, respectively. For simplicity, we assume that the upper-level constraints (1b) depend only on \mathbf{x} and not on \mathbf{y}^* and $\boldsymbol{\lambda}^*$. This assumption is common in the bilevel programming literature, as such dependence could result in a disconnected or empty feasible region of the BPP-D; cf. Colson et al. (2005) and Shi, Lu, and Zhang (2005).

The linear programming (LP) problem (1d)-(1f) is the lower-level primal problem and (1g)-(1h) is the corresponding dual problem. The upper-level variables \mathbf{x} are fixed parameters in the right-hand side of the lower-level primal problem and the objective function of the dual problem. Hence, \mathbf{y}^* and $\boldsymbol{\lambda}^*$ are implicitly parameterized by \mathbf{x} and so is the function $f(\mathbf{y}^*, \boldsymbol{\lambda}^*)$.

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

We define the feasibility sets

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\},$$

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

$$P(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^l \mid \mathbf{C}\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{e}, \mathbf{y} \geq \mathbf{0}\},$$

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

$$S^* = S \cap \{\mathbf{x} \in \mathbb{R}^n \mid P(\mathbf{x}) \neq \emptyset\},$$

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 3.1. *Assume that $\{\boldsymbol{\lambda} \in \mathbb{R}^l \mid \mathbf{C}^T \boldsymbol{\lambda} \leq \mathbf{p}\}$ is non-empty and S^* is non-empty and bounded.*

Moreover, for fixed $\mathbf{x} \in S^*$, the assumptions guarantee primal and dual feasibility and thereby also optimality of the lower-level problem.

With these assumptions, we can define $\mathbf{y}^*(\mathbf{x})$ and $\boldsymbol{\lambda}^*(\mathbf{x})$ to be lower-level primal and dual optimal solutions for fixed $\mathbf{x} \in S^*$. Hence, the parameterized solutions $\mathbf{y}^* := \mathbf{y}^*(\mathbf{x})$ and $\boldsymbol{\lambda}^* := \boldsymbol{\lambda}^*(\mathbf{x})$ in (1a)-(1h) are well-defined. For non-unique primal and dual optimal solutions to (1d)-(1f) and (1g)-(1h), $\mathbf{y}^*(\mathbf{x})$ and $\boldsymbol{\lambda}^*(\mathbf{x})$ may be chosen as so-called *optimistic* or *pessimistic* solutions to a bilevel programming problem, as defined by Colson et al. (2005). The optimistic and pessimistic solutions are the best and worst lower-level solutions, respectively, with respect to the upper-level objective function. If these likewise also non-unique, one of them may simply be chosen.

By choosing an optimal solution to the lower-level problem, we can define the function $F : \mathbb{R}^n \rightarrow \mathbb{R}$ such that

$$F(\mathbf{x}) = f(\mathbf{y}^*(\mathbf{x}), \boldsymbol{\lambda}^*(\mathbf{x})), \quad \mathbf{x} \in S^*.$$

With this parametrization, the BPP-D becomes

$$\min \mathbf{c}^T \mathbf{x} + F(\mathbf{x}) \tag{2a}$$

$$\text{s.t. } \mathbf{x} \in S^* \tag{2b}$$

This is a single-level linearly constrained optimization problem with regard to \mathbf{x} , albeit $F(\mathbf{x})$ is generally not known in closed form. 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 $\mathbf{x} \in S^*$ be given. Assume that the rank of the matrix \mathbf{C} is k . Let $\mathbf{B} \in \mathbb{R}^{k \times k}$ be a basis for $P(\mathbf{x})$, i.e., a non-singular $k \times k$ submatrix of \mathbf{C} . The primal basic solution is $\mathbf{y}(\mathbf{x})^T = (\mathbf{y}_B(\mathbf{x})^T, \mathbf{y}_N(\mathbf{x})^T)$, where $\mathbf{y}_B(\mathbf{x}) = \mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e})$ and $\mathbf{y}_N(\mathbf{x}) = \mathbf{0}$ are the subvectors with k basic and $l-k$ non-basic variables, respectively. The complementary dual basic solution is $\boldsymbol{\lambda}(\mathbf{x}) = (\mathbf{B}^{-1})^T \mathbf{p}_B$, where \mathbf{p}_B is the subvector of \mathbf{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.

Definition 3.2. (Bertsimas & Tsitsiklis, 1997) Let $\mathbf{B} \in \mathbb{R}^{k \times k}$ be a basis for $P(\mathbf{x})$. A basic solution $\mathbf{y}(\mathbf{x})^T = (\mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e}), \mathbf{0})^T$ 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, \mathbf{B} is uniquely defined. For a

degenerate primal basic solution, however, \mathbf{B} is no longer uniquely defined and the complementary dual basic solution is non-unique.

The following defines critical regions.

Definition 3.3. (Gal, 1995) Let $\mathbf{B} \in \mathbb{R}^{k \times k}$ be a basis for $P(\mathbf{x})$. The critical region corresponding to \mathbf{B} is $\Lambda_{\mathbf{B}} = \{\mathbf{x} \in S^* \mid \mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e}) \geq \mathbf{0}\}$.

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

Now, let \mathcal{B} be the set of all optimal bases for $P(\mathbf{x})$ with $\mathbf{x} \in S^*$. Evidently,

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

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 \mathbf{B} , we denote the interior of $\Lambda_{\mathbf{B}}$ by $\Lambda_{\mathbf{B}}^{\circ}$.

Proposition 3.4. Let $\Lambda_{\mathbf{B}} \subseteq S^*$ be the critical region corresponding to the basis \mathbf{B} . Then, $F(\mathbf{x})$ is differentiable on $\Lambda_{\mathbf{B}}^{\circ}$ with gradient

$$\nabla F(\mathbf{x}) = (\mathbf{B}^{-1}\mathbf{D})^T \frac{\partial f}{\partial \mathbf{y}_{\mathbf{B}}}(\mathbf{y}^*(\mathbf{x}), \boldsymbol{\lambda}^*(\mathbf{x})), \quad \mathbf{x} \in \Lambda_{\mathbf{B}}^{\circ}.$$

where $\partial f / \partial \mathbf{y}_{\mathbf{B}}$ is the vector of the k derivatives of f with respect to the basic variables of $\mathbf{y}_{\mathbf{B}}$. If $f(\cdot, \boldsymbol{\lambda})$ is an affine function for any given $\boldsymbol{\lambda}$, then $F(\cdot)$ is an affine function on $\Lambda_{\mathbf{B}}^{\circ}$.

Proof. Let $\mathbf{x} \in \Lambda_{\mathbf{B}}^{\circ}$. The primal optimal basic solution is given as $\mathbf{y}^*(\mathbf{x})^T = (\mathbf{y}_{\mathbf{B}}^*(\mathbf{x})^T, \mathbf{y}_{\mathbf{N}}^*(\mathbf{x})^T) = (\mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e}), \mathbf{0})^T$, and hence, $\mathbf{y}_{\mathbf{N}}^*(\mathbf{x})$ does not depend on \mathbf{x} at optimality. The complementary dual basic solution is $\boldsymbol{\lambda}^*(\mathbf{x}) = (\mathbf{B}^{-1})^T \mathbf{p}_{\mathbf{B}}$, and so, $\boldsymbol{\lambda}^*(\mathbf{x})$ likewise does not depend on \mathbf{x} at optimality.

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

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

where $y_i^*, i \in B$ refer to the basic variables of $\mathbf{y}_{\mathbf{B}}$.

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

As a result of Proposition 3.4, F is a piece-wise linear (but not necessarily continuous) function 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 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 linearity 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 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 of the general form

$$f(\mathbf{y}, \boldsymbol{\lambda}) = \mathbf{d}^T \mathbf{y} + \mathbf{h}^T \boldsymbol{\lambda} + \boldsymbol{\lambda}^T \mathbf{H} \mathbf{y},$$

with $\mathbf{d} \in \mathbb{R}^l$, $\mathbf{h} \in \mathbb{R}^k$ and $\mathbf{H} \in \mathbb{R}^{k \times l}$. For a given basis \mathbf{B} and $\mathbf{x} \in \Lambda_{\mathbf{B}}^{\circ}$, by Proposition 3.4, the gradient of F is

$$\nabla F(\mathbf{x}) = (\mathbf{B}^{-1} \mathbf{D})^T (\mathbf{d}_{\mathbf{B}} + \mathbf{H}_{\mathbf{B}}^T \mathbf{p}_{\mathbf{B}}^T \mathbf{B}^{-1}),$$

where $\mathbf{d}_{\mathbf{B}}$ is the subvector of \mathbf{d} with entries corresponding to the basis \mathbf{B} and $\mathbf{H}_{\mathbf{B}}$ is the submatrix of \mathbf{H} with columns corresponding to the basis \mathbf{B} .

4. Parametric programming

To solve the BPP-D, we utilize (2) and propose a parametric programming approach. The approach takes advantage of the critical regions and Proposition 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 (1d)-(1f).

Definition 4.1. (Gal, 1995) Two critical regions, Λ_1, Λ_2 , are *neighbors* if the following holds for their corresponding bases $\mathbf{B}_1, \mathbf{B}_2$:

- (1) There exists an $\mathbf{x} \in S^*$ for which \mathbf{B}_1 and \mathbf{B}_2 both are optimal bases to (1d)-(1f).
- (2) It is possible to pass from \mathbf{B}_1 to \mathbf{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 (1d)-(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 successively determining all neighbors to the current basis. Then,

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

Furthermore, when a neighboring basis is defined in terms of an iteration of the dual simplex method, $\Lambda_{\mathbf{B}_1}^o \cap \Lambda_{\mathbf{B}_2}^o = \emptyset$ for $\mathbf{B}_1, \mathbf{B}_2 \in \mathcal{B}^*$, i.e., neighbors overlap only at their boundaries; cf. Gal (1995) (Theorem IV-5). Although it may 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 obtain a neighbor. Consider a basis \mathbf{B} and the corresponding critical region $\Lambda_{\mathbf{B}}$. Let $\mathbf{x} \in \Lambda_{\mathbf{B}}$ and consider the primal basic solution $\mathbf{y}^T := \mathbf{y}(\mathbf{x})^T = (\mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e}), \mathbf{0})^T$. To determine if a neighbor exists, we examine the potential variables to leave basis in the dual simplex method. The variable $y_i, i \in B$ can only leave the basis if

$$(\mathbf{B}^{-1}\mathbf{C})_{ij} < 0 \text{ for at least one non-basic variable } y_j, \quad (3)$$

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

$$(\mathbf{B}^{-1}(\mathbf{D}\mathbf{x}^* + \mathbf{e}))_i = 0, \quad (4)$$

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 \mathbf{x}^* , we solve the LP problem

$$\min (\mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e}))_i \quad (5a)$$

$$\text{s.t. } \mathbf{B}^{-1}(\mathbf{D}\mathbf{x} + \mathbf{e}) \geq \mathbf{0} \quad (5b)$$

$$\mathbf{A}\mathbf{x} = \mathbf{b} \quad (5c)$$

$$\mathbf{x} \geq \mathbf{0} \quad (5d)$$

If the optimal solution, \mathbf{x}^* , to (5) has an optimal value of zero, then $\Lambda_{\mathbf{B}}$ has a neighbor. The neighbor is obtained by replacing \mathbf{x} by \mathbf{x}^* in (1d)-(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 successively determining all neighbors.

Algorithm 4.2. Parametric Programming

- Step 0 (initialization) Set $h := 0$. Given an initial solution $\mathbf{x}_0 \in S^*$, let $\mathbf{x} := \mathbf{x}_0$ in (1d)-(1f) and solve the problem. Store an optimal basis \mathbf{B}_0 and determine $F(\mathbf{x}_0), \nabla F_{\mathbf{B}_0}(\mathbf{x}_0), \Lambda_{\mathbf{B}_0}$. Set $\mathcal{B} := \{\mathbf{B}_0\}$.
- Step 1 (iteration h) If $\mathcal{B} = \emptyset$, then stop. Otherwise, set $h := h + 1$, select $\mathbf{B}_h \in \mathcal{B}$ and set $\mathcal{B} := \mathcal{B} \setminus \{\mathbf{B}_h\}$.
- Step 2 (determine leaving variable) Let $\mathbf{B} := \mathbf{B}_h$. Select an $i \in B$ that has not yet been inspected and that satisfies (3). Solve the problem (5), let \mathbf{x}_{hi} be an optimal solution and determine if a neighbor exists. If not, return to 2. If all $i \in B$ have been inspected, return to Step 1.
- Step 3 (determine entering variable) Let $\mathbf{x} := \mathbf{x}_{hi}$ in (1d)-(1f) and carry out an iteration of the dual simplex method. Store a neighboring basis \mathbf{B}_j and determine $F(\mathbf{x}_{hi}), \nabla F_{\mathbf{B}_j}(\mathbf{x}_{hi}), \Lambda_{\mathbf{B}_j}$. Set $\mathcal{B} := \mathcal{B} \cup \{\mathbf{B}_j\}$. Return to Step 2.

In Algorithm 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 . As this 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 \mathbf{B}_h is selected. In Steps 2 and 3, all neighbors to the critical region $\Lambda_{\mathbf{B}_h}$ are obtained by iterating through potential variables to leave the basis. For each of these, the variable to enter the basis is determined by the minimum ratio test of the dual simplex method. For all neighbors, the algorithm stores an optimal basis, determines the critical region and evaluates the parametric function. However, only bases that have not yet been investigated need to be added to the pool. Algorithm 4.2 terminates when all neighboring critical regions of S^* are obtained. Finite termination of the approach 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 (1) while omitting (1d) and (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 the LP

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} + \mathbf{h}^T \boldsymbol{\lambda} \\ \text{s.t.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \\ & \mathbf{C} \mathbf{y} = \mathbf{D} \mathbf{x} + \mathbf{e} \\ & \mathbf{y} \geq \mathbf{0} \\ & \mathbf{C}^T \boldsymbol{\lambda} \leq \mathbf{p} \end{aligned}$$

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 (1).

Having determined all neighboring critical regions of S^* , the next step is to solve the restriction of (2) to each region. For critical region $\Lambda_{\mathbf{B}}$ and given $\mathbf{x}_{\mathbf{B}} \in \Lambda_{\mathbf{B}}$, the restricted problem is

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} + F_{\mathbf{B}}(\mathbf{x}_{\mathbf{B}}) + \nabla F_{\mathbf{B}}(\mathbf{x}_{\mathbf{B}})^T (\mathbf{x} - \mathbf{x}_{\mathbf{B}}) & (7a) \\ \text{s.t.} \quad & \mathbf{x} \in \Lambda_{\mathbf{B}} & (7b) \end{aligned}$$

This is an LP, as $\Lambda_{\mathbf{B}}$ is a polyhedron. Moreover, $\Lambda_{\mathbf{B}}$ is non-empty and bounded. Thus, there exists an optimal basic solution, i.e., in a vertex of the critical region. Recall 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 (7) for all neighboring critical regions and compare their optimal solutions.

The parametric programming approach relies on the critical regions of S^* . An upper bound on the number of regions is the number of distinct bases in the lower-level problem, given by the binomial coefficient

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

It can be computationally demanding to obtain all regions. 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.

This solution approach is particularly useful in two respects: Since the parametric programming method determines $F(\mathbf{x})$ for every $\mathbf{x} \in S^*$, 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 fixed upper-level solution. We propose two variants of the decomposition approach: An exact solution method that finds the global optimum and a heuristic with some computational advantages. In the results, we refer to the above as the full parametric programming method and the two variants as decomposition by parametric programming and the parametric programming heuristic, respectively.

4.1. Decomposition

Consider a BPP-D of the form

$$\min \mathbf{c}^T \mathbf{x} + \sum_{t=1}^s f_t(\mathbf{y}_t^*, \boldsymbol{\lambda}_t^*) \quad (8a)$$

$$\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b} \quad (8b)$$

$$\mathbf{x} \geq \mathbf{0} \quad (8c)$$

$$\mathbf{y}_t^* \in \operatorname{argmin}\{\mathbf{p}_t^T \mathbf{y}_t \quad (8d)$$

$$\text{s.t. } \mathbf{C}_t \mathbf{y}_t = \mathbf{D}_t \mathbf{x} + \mathbf{e}_t \quad (8e)$$

$$\mathbf{y}_t \geq \mathbf{0} \quad \forall t \quad (8f)$$

$$\boldsymbol{\lambda}_t^* \in \operatorname{argmax}\{\boldsymbol{\lambda}_t^T (\mathbf{D}_t \mathbf{x} + \mathbf{e}_t) \quad (8g)$$

$$\text{s.t. } \mathbf{C}_t^T \boldsymbol{\lambda}_t \leq \mathbf{p}_t \quad \forall t \quad (8h)$$

Note that for fixed $\mathbf{x} \in S^*$, the lower level separates into a number of LP problems indexed by $t = 1, \dots, s$. Moreover, the upper-level objective function involves a sum of functions $f_t : \mathbb{R}^{l_t} \times \mathbb{R}^{k_t} \rightarrow \mathbb{R}$ of lower-level primal and dual optimal solutions $\mathbf{y}_t^* \in \mathbb{R}^{l_t}$ and $\boldsymbol{\lambda}_t^* \in \mathbb{R}^{k_t}$. The lower-level cost vector and right-hand side have dimensions $\mathbf{p}_t \in \mathbb{R}^{l_t}$ and $\mathbf{e}_t \in \mathbb{R}^{k_t}$ and the constraint matrices $\mathbf{C}_t \in \mathbb{R}^{k_t \times l_t}$ and $\mathbf{D}_t \in \mathbb{R}^{k_t \times n}$. Thus, the vectors and matrices of the BPP-D are partitioned into blocks such that $\mathbf{p}^T = (\mathbf{p}_1^T, \mathbf{p}_2^T, \dots, \mathbf{p}_s^T)$, $\mathbf{e}^T = (\mathbf{e}_1^T, \mathbf{e}_2^T, \dots, \mathbf{e}_s^T)$ and

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \\ \mathbf{0} & \mathbf{0} & & \mathbf{C}_s \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \vdots \\ \mathbf{D}_s \end{pmatrix}.$$

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

$$P_t(\mathbf{x}) = \{\mathbf{y}_t \in \mathbb{R}^{l_t} \mid \mathbf{C}_t \mathbf{y}_t = \mathbf{D}_t \mathbf{x} + \mathbf{e}_t, \mathbf{y}_t \geq \mathbf{0}\},$$

and let

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

We further define the parametric function as the sum

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

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

$$F_t(\mathbf{x}) = f_t(\mathbf{y}_t^*(\mathbf{x}), \boldsymbol{\lambda}_t^*(\mathbf{x})), \quad \mathbf{x} \in S^*.$$

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

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 successively determining new neighbors to the current basis by Algorithm 4.2. By Gal (1995),

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

In solving a BPP-D of the form (8), the lower-level subproblems can be handled separately. In particular, the lower-level subproblem indexed by t is processed by Algorithm 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 4.3 shows that it is indeed sufficient to determine critical regions by iterating through neighbors, for one lower-level subproblem at a time.

Proposition 4.3. *For a BPP-D of the form (8), the following holds*

$$S^* = \bigcap_{t=1}^s S_t^* = \bigcap_{t=1}^s \left(\bigcup_{\mathbf{B}_t \in \mathcal{B}_t^*} \Lambda_{\mathbf{B}_t} \right) = \bigcup_{(\mathbf{B}_1, \mathbf{B}_2, \dots, \mathbf{B}_s) \in \mathcal{B}_1^* \times \mathcal{B}_2^* \times \dots \times \mathcal{B}_s^*} \left(\bigcap_{t=1}^s \Lambda_{\mathbf{B}_t} \right).$$

By Proposition 4.3, the BPP-D can be handled by solving restrictions of (8). For critical regions $\Lambda_{\mathbf{B}_1}, \Lambda_{\mathbf{B}_2}, \dots, \Lambda_{\mathbf{B}_s}$, the restricted problem is

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} + \sum_{t=1}^s F_t(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \Lambda_{\mathbf{B}_1} \cap \Lambda_{\mathbf{B}_2} \cap \dots \cap \Lambda_{\mathbf{B}_s} \end{aligned}$$

where F_t is affine on $\Lambda_{\mathbf{B}_t}$ and therefore F is affine on $\Lambda_{\mathbf{B}_1} \cap \Lambda_{\mathbf{B}_2} \cap \dots \cap \Lambda_{\mathbf{B}_s}$. 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 the restricted problem, we may find all vertices of $\Lambda_{\mathbf{B}_1} \cap \Lambda_{\mathbf{B}_2} \cap \dots \cap \Lambda_{\mathbf{B}_s}$ and evaluate these. These are contained in the vertices of the sets $\Lambda_{\mathbf{B}_1}, \Lambda_{\mathbf{B}_2}, \dots, \Lambda_{\mathbf{B}_s}$. We find all vertices of all critical regions

corresponding to bases in $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_s$ and thereby avoid the examination of an exponential number of combinations of critical regions. 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

$$\begin{aligned} \min \mathbf{c}^T \mathbf{x} + F_{t, \mathbf{B}_t}(\mathbf{x}_{\mathbf{B}_t}) + \nabla F_{t, \mathbf{B}_t}(\mathbf{x}_{\mathbf{B}_t})^T (\mathbf{x} - \mathbf{x}_{\mathbf{B}_t}) \\ \text{s.t. } \mathbf{x} \in \Lambda_{\mathbf{B}_t} \end{aligned}$$

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

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. For example, when the resulting MIP is neither linear nor convex, it is very 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 5.1. *Assume that $f(\cdot, \cdot)$ is a bilinear function on the form*

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

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

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

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

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

$$\begin{aligned} K_1 &\triangleq \{j \in J_0 : C_{ij} = 0 \ \forall i \in \bar{I}_0\}, \\ K_2 &\triangleq J_0 \setminus K_1. \end{aligned}$$

With the definitions of $J_0, I_0, \bar{J}_0, \bar{I}_0, K_1$ and K_2 , we construct 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:

$$\mathbf{H} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{cc} \bar{J}_0 & J_0 \\ \mathbf{H}_{11} & \mathbf{H}_{12} \\ \mathbf{H}_{21} & \mathbf{H}_{22} \end{array} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{cc} \bar{J}_0 & J_0 \\ \mathbf{H}_{11} & \mathbf{0} \\ \mathbf{H}_{21} & \mathbf{0} \end{array}$$

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

$$\mathbf{C} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{ccc} \bar{J}_0 & K_1 & K_2 \\ \mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13} \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23} \end{array} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{ccc} \bar{J}_0 & K_1 & K_2 \\ \mathbf{C}_{11} & \mathbf{0} & \mathbf{C}_{13} \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23} \end{array}$$

$$\mathbf{D} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \mathbf{D}_1 \\ \mathbf{D}_2 \end{array} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \mathbf{D}_1 \\ \mathbf{0} \end{array}$$

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

$$\begin{array}{c} \boldsymbol{\lambda} \\ \mathbf{y} \end{array} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \boldsymbol{\lambda}_1 \\ \boldsymbol{\lambda}_2 \end{array} \quad \begin{array}{c} \mathbf{e} \\ \mathbf{p} \end{array} = \begin{array}{c} \bar{I}_0 \\ I_0 \end{array} \begin{array}{c} \mathbf{e}_1 \\ \mathbf{e}_2 \end{array}$$

$$\begin{array}{c} \mathbf{y} \\ \mathbf{p} \end{array} = \begin{array}{c} \bar{J}_0 \\ K_1 \\ K_2 \end{array} \begin{array}{c} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \end{array} \quad \begin{array}{c} \mathbf{p} \\ \mathbf{p} \\ \mathbf{p} \end{array} = \begin{array}{c} \bar{J}_0 \\ K_1 \\ K_2 \end{array} \begin{array}{c} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{array}$$

5.1. Complementarity constraints and linearization

The following assumptions allow for linearization.

Assumption 5.2. *Assume that:*

- (1) $\mathbf{H}_{11} = \mathbf{0}$
- (2) $\mathbf{C}_{21} = \alpha \mathbf{H}_{21}$ for some $\alpha \in \mathbb{R} \setminus \{0\}$
- (3) $\mathbf{C}_{23} = \mathbf{0}$

By replacing the lower level by its KKT-conditions and linearizing the bilinear term, we obtain an MPEC with linear objective function and constraints, except for its bilinear 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 5.3. *Under Assumptions 5.1 and 5.2, an equivalent formulation of the BPP-D is:*

$$\min \mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} + \mathbf{h}^T \boldsymbol{\lambda} + (\boldsymbol{\lambda}_2^T \mathbf{e}_2 - \mathbf{p}_2^T \mathbf{y}_2) / \alpha \quad (11a)$$

$$\text{s.t. } \mathbf{A} \mathbf{x} = \mathbf{b} \quad (11b)$$

$$\mathbf{x} \geq \mathbf{0} \quad (11c)$$

$$\mathbf{C} \mathbf{y} = \mathbf{D} \mathbf{x} + \mathbf{e} \quad (11d)$$

$$\mathbf{0} \leq \mathbf{y} \perp \mathbf{p} - \mathbf{C}^T \boldsymbol{\lambda} \geq \mathbf{0} \quad (11e)$$

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

$$\mathbf{C}\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{e} \quad (12a)$$

$$\mathbf{y} \geq \mathbf{0} \quad (12b)$$

$$\mathbf{C}^T \boldsymbol{\lambda} \leq \mathbf{p} \quad (12c)$$

$$\mathbf{p} - \mathbf{C}^T \boldsymbol{\lambda} - \boldsymbol{\mu} = \mathbf{0} \quad (12d)$$

$$\mathbf{y}^T \boldsymbol{\mu} = 0 \quad (12e)$$

including primal feasibility (12a)-(12b), dual feasibility (12c), stationarity (12d) and complementarity (12e), and are equivalent to (11d)-(11e).

To linearize the upper-level objective function, we utilize the strong duality property for the lower-level LP (second equality), the lower-level constraints (12a) (fourth equality) and the definitions of I_0 and K_1 :

$$\begin{aligned} \mathbf{p}_1^T \mathbf{y}_1 + \mathbf{p}_2^T \mathbf{y}_2 + \mathbf{p}_3^T \mathbf{y}_3 &= \mathbf{p}^T \mathbf{y} \\ &= \boldsymbol{\lambda}^T (\mathbf{D}\mathbf{x} + \mathbf{e}) \\ &= \boldsymbol{\lambda}_1^T (\mathbf{D}_1 \mathbf{x} + \mathbf{e}_1) + \boldsymbol{\lambda}_2^T \mathbf{e}_2 \\ &= \boldsymbol{\lambda}_1^T (\mathbf{C}_{11} \mathbf{y}_1 + \mathbf{C}_{13} \mathbf{y}_3) + \boldsymbol{\lambda}_2^T \mathbf{e}_2. \end{aligned}$$

By rearranging terms and noting that $\mathbf{y}_1^T \mathbf{C}_{11}^T \boldsymbol{\lambda}_1 = \boldsymbol{\lambda}_1^T \mathbf{C}_{11} \mathbf{y}_1$, we obtain

$$\mathbf{y}_1^T (\mathbf{p}_1 - \mathbf{C}_{11}^T \boldsymbol{\lambda}_1) = -\mathbf{p}_2^T \mathbf{y}_2 - \mathbf{p}_3^T \mathbf{y}_3 + \boldsymbol{\lambda}_1^T \mathbf{C}_{13} \mathbf{y}_3 + \boldsymbol{\lambda}_2^T \mathbf{e}_2. \quad (13)$$

From the stationarity and complementarity constraints (12d)-(12e) and Assumption 4.2 we have that

$$\mathbf{y}_1^T (\mathbf{p}_1 - \mathbf{C}_{11}^T \boldsymbol{\lambda}_1) = \alpha \mathbf{y}_1^T \mathbf{H}_{21}^T \boldsymbol{\lambda}_2. \quad (14)$$

The constraints (12d)-(12e) and Assumption 4.3 likewise give us

$$\mathbf{y}_3^T \mathbf{p}_3 = \mathbf{y}_3^T \mathbf{C}_{13}^T \boldsymbol{\lambda}_1. \quad (15)$$

Substituting (14) and (15) into (13) we arrive at

$$\alpha \mathbf{y}_1^T \mathbf{H}_{21}^T \boldsymbol{\lambda}_2 = -\mathbf{p}_2^T \mathbf{y}_2 + \boldsymbol{\lambda}_2^T \mathbf{e}_2, \quad (16)$$

and, finally, using the definitions of J_0 and Assumption 4.1, (16) becomes

$$\boldsymbol{\lambda}^T \mathbf{H} \mathbf{y} = (\boldsymbol{\lambda}_2^T \mathbf{e}_2 - \mathbf{p}_2^T \mathbf{y}_2) / \alpha. \quad (17)$$

□

The MPEC (11) 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 (11) is a non-convex optimization problem, NLP and MPEC solvers can only guarantee local optimality.

5.2. Integer programming formulations

The equivalent MPEC (11) 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:

$$\min \mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} + \mathbf{h}^T \boldsymbol{\lambda} + (\boldsymbol{\lambda}_2^T \mathbf{e}_2 - \mathbf{p}_2^T \mathbf{y}_2) / \alpha \quad (18a)$$

$$\text{s.t. } \mathbf{A} \mathbf{x} = \mathbf{b} \quad (18b)$$

$$\mathbf{C} \mathbf{y} = \mathbf{D} \mathbf{x} + \mathbf{e} \quad (18c)$$

$$0 \leq y_j \leq M \delta_j \quad \forall j \quad (18d)$$

$$0 \leq p_j - \sum_{i=1}^k C_{ij} \lambda_i \leq M(1 - \delta_j) \quad \forall j \quad (18e)$$

$$\mathbf{x} \geq 0, \mathbf{y} \geq 0, \boldsymbol{\delta} \in \{0, 1\}^l \quad (18f)$$

Here, M is a sufficiently large constant and $\boldsymbol{\delta}$ is a vector of binary variables. The choice of M can be difficult, since too small a constant may cut off feasible solutions while too large a constant may produce an ill-conditioned or weak LP-relaxation. For a general method to find suitable M 's; see also Pineda, Bylling, and Morales (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:

$$\min \mathbf{c}^T \mathbf{x} + \mathbf{h}^T \mathbf{y} + \mathbf{k}^T \boldsymbol{\lambda} + (\boldsymbol{\lambda}_2^T \mathbf{e}_2 - \mathbf{p}_2^T \mathbf{y}_2) / \alpha \quad (19a)$$

$$\text{s.t. } \mathbf{A} \mathbf{x} = \mathbf{b} \quad (19b)$$

$$\mathbf{C} \mathbf{y} = \mathbf{D} \mathbf{x} + \mathbf{e} \quad (19c)$$

$$s_j(1) = y_j \quad \forall j \quad (19d)$$

$$s_j(2) = p_j - \sum_{i=1}^k C_{ij} \lambda_i \quad \forall j \quad (19e)$$

$$\mathbf{y} \geq 0, \mathbf{x} \geq 0, \quad (19f)$$

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.

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 and 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.

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):

$$\min 40000x + 8760(10y_1^* - \lambda_1^*y_1^*) \quad (20a)$$

$$\text{s.t. } 0 \leq x \leq 250 \quad (20b)$$

$$y_1^* \in \operatorname{argmin}\{10y_1 + 12y_2 + 15y_3 \quad (20c)$$

$$\text{s.t. } y_1 + y_2 + y_3 = 200, 0 \leq y_1 \leq x, \quad (20d)$$

$$0 \leq y_2 \leq 150, 0 \leq y_3 \leq 100\} \quad (20e)$$

$$\lambda_1^* \in \operatorname{argmax}\{200\lambda_1 + x\lambda_2 + 150\lambda_3 + 100\lambda_4 \quad (20f)$$

$$\text{s.t. } \lambda_1 + \lambda_2 \leq 10, \lambda_1 + \lambda_3 \leq 12, \quad (20g)$$

$$\lambda_1 + \lambda_4 \leq 15, \lambda_2, \lambda_3, \lambda_4 \geq 0\} \quad (20h)$$

The upper-level variable, x , is investment in 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 (20a) 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 (20b) are upper and lower bounds on investment capacity. The lower-level primal problem consists of minimizing total production costs; cf. (20c), such that supply meets demand and production is bounded by the generation capacities in (20d)-(20e). The lower-level dual problem, (20f)-(20h), determines the market price of production. Note that the upper-level variable, x , is a parameter of both the lower-level primal (20c)-(20d) and dual (20f)-(20h).

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 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 1 (b) is likewise piecewise affine but discontinuous. For details on how to obtain the critical regions and parametric function by Algorithm 4.2, see Appendix A.

[Figure 1 about here.]

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 of producer 1 (the investor) 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, in 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

$$\binom{6}{4} = 15, \quad (21)$$

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

With the parametrization of F , problem (A2) becomes

$$\min \quad 40000x + 8760F(x) \quad (22a)$$

$$\text{s.t.} \quad 0 \leq x \leq 250. \quad (22b)$$

The restrictions of (22) to the critical regions are

$$\min \quad 40000x + 8760(10 - 15)x \quad (23a)$$

$$\text{s.t.} \quad 0 \leq x \leq 50, \quad (23b)$$

$$\min \quad 40000x + 8760(10 - 12)50 \quad (23c)$$

$$\text{s.t.} \quad 50 \leq x \leq 200, \quad (23d)$$

$$\min \quad 40000x + 8760(10 - 10)50 \quad (23e)$$

$$\text{s.t.} \quad 200 \leq x \leq 250, \quad (23f)$$

which are clearly LP problems. Their optimal solutions are $x^* = 50, 50$ and 200 with objective function values $-190,000, 1,124,000$ and $8,000,000$, respectively. The optimal solution to (22) is therefore $x^* = 50$ 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 5.2 holds, note that

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

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, $\mathbf{H}_{11} = \mathbf{C}_{23} = 0$ and $\mathbf{C}_{21} = -8760\mathbf{H}_{21}$.

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 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 2 about here.]

6.2. Implementation

We proceed with the two case studies. To obtain numerical results, the parametric programming approach of Section 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 in all runs.

The above has to be updated, as you implement dual simplex-iterations

6.3. Case study I: Investment in production capacity

The first case study is an extension of the problem in Section 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.

6.3.1. Data

We use data from the Danish power market. This market is divided into two price regions connected by a DC cable with a capacity of 600 MW; 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 the Danish District Heating Association and Danish Energy Association (2016), which also provide production cost estimates.

Hourly demand data is available from Nord Pool AS (2017). We aggregate this data using k-means clustering 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.

6.3.2. Model

Our model is a simplified version of 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. 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:

$$\min \sum_{i=1}^n \left(C_i x_i + \sum_{t=1}^s \rho_t (c_i^1 y_{it}^1 - \lambda_{it} y_{it}^1) \right) \quad (24a)$$

$$\text{s.t. } 0 \leq \sum_{i=1}^n x_i \leq x^{\max} \quad (24b)$$

$$y_{it}^1, \lambda_{it} \text{ are optimal solutions to (25) } \forall i, t \quad (24c)$$

The lower-level problem involves market-clearing for each representative hour t . Subproblem t is

$$\min \sum_{i=1}^n \sum_{g \in \mathcal{G}} c_i^g y_{it}^g \quad (25a)$$

$$\text{s.t. } \sum_{g \in \mathcal{G}} y_{it}^g - \sum_{j \in \mathcal{I}_i} B_{ij} (\theta_{it} - \theta_{jt}) = d_{it} : \lambda_{it} \quad \forall i \quad (25b)$$

$$0 \leq y_{it}^1 \leq x_i : \mu_{it}^1 \quad \forall i \quad (25c)$$

$$0 \leq y_{it}^g \leq y_i^{\max, g} : \mu_{it}^g \quad \forall i, g \neq 1 \quad (25d)$$

$$B_{ij} (\theta_{it} - \theta_{jt}) \leq F_{ij}^{\max} : \eta_{ijt} \quad \forall j \in \mathcal{I}_i, i \quad (25e)$$

$$-\pi \leq \theta_{it} \leq \pi : \alpha_{it}, \beta_{it} \quad \forall i \quad (25f)$$

$$\theta_{it} = 0 : \gamma_t \quad i = \text{ref} \quad (25g)$$

A complete nomenclature is provided in Appendix B.1. 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 hour t , from the investor, y_{it}^1 , and its rivals in the market, $y_{it}^g, g \in \mathcal{G}, g \neq 1$. Moreover, λ_{it} is the market's nodal price of production and $\mu_{it}^g, \eta_{ijt}, \alpha_{it}, \beta_{it}, \gamma_t$ are the remaining dual variables. The upper-level objective (24a) is to maximize (negative) profits from investment and operation, operational profits being the sum of profits for each representative hour t weighted by the number of hours represented, ρ_t . The upper-level constraint (24b) bounds total investment capacity. The lower-level constraints include a power balance for each node, (25b). The set \mathcal{I}_i defines all nodes connected to node i , B_{ij} denotes the susceptance of the transmission line between nodes i and $j \in \mathcal{I}_i$ and θ_{it} is the voltage angle in node i and hour t , determining the flow. Thus, the sum on the left-hand-side of the equality is the net outflow from node i . Further constraints include generation capacity constraints, (25c) and (25d), transmission capacity constraints, (25e), and bounds on the voltage angles, (25f). Finally, (25g) fixes the voltage angle of some reference node to zero. For more details regarding the model, we refer to Conejo et al. (2016).

6.3.3. 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 and 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 1.

[Table 1 about here.]

As seen from the table, the NLP and MPEC methods return local infeasibility for all instances, even when we provide an initial feasible solution like in the parametric programming approaches. As shown by Scheel and Scholtes (2000), all feasible points of the MPEC (11) are non-regular (i.e., the gradients of the binding constraints are linearly dependent), which is likely a 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 global optimality.

We further investigate the MILP and parametric programming approaches. Table 2 and Figure 3 show the solution times for the MILP, the decomposition by parametric programming and the parametric programming heuristic, as functions of the number of lower-level subproblems. Figure 3 (a) and (b) cover the number of lower-level problems from 10 to 100 and from 200 to 5000, respectively. Table 2 further lists the relative deviations in the objective function values from the optimal, for the heuristic and the MILP. 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). All running times above the time limit are marked as not available (N/A). Note also that the results do not include the full parametric programming method 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.

[Table 2 about here.]

[Figure 3 about here.]

From the results, we notice that for less than 100 subproblems, the MILP has the lowest 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 MILP when it does not solve within the time limit (and thus only returns a feasible solution); see also Table 2.

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 significant. 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* increases with the MILP.

The solution quality of the heuristic increases (i.e., the average relative deviation in upper-level objective function values from the optimal decreases) as the number of subproblems increases, whereas it is almost constant for the MILP. We explain this as follows. With a higher number of lower-level subproblems, the number of critical regions considered during the heuristic is higher. 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 the 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 6,487 seconds and the heuristic with a deviation from the global optimal value of 3.61% in 1,658 seconds.

[Figure 4 about here.]

In Figure 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 report the relative deviation in upper-level objective function value from the optimal. All parametric programming approaches obtain 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% (i.e., with significantly higher objective function values than the two solutions).

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 the regions of the transmission network.

6.4.1. Data

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.

6.4.2. 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:

$$\min \sum_{(i,j) \in \mathcal{J}^1} \left(C_{ij} x_{ij} - \sum_{t=1}^s \rho_t f_{ijt} (\lambda_{it} - \lambda_{jt}) \right) \quad (26a)$$

$$\text{s.t. } 0 \leq \sum_{(i,j) \in \mathcal{J}^1} C_{ij} x_{ij} \leq x^{max} \quad (26b)$$

$$0 \leq x_{ij} \leq F_{ij}^{max} \quad \forall (i,j) \in \mathcal{J}^1 \quad (26c)$$

$$f_{ijt}, \lambda_{it} \text{ are optimal solutions to (27) } \forall i, j, t \quad (26d)$$

The lower-level subproblem t is:

$$\min \sum_{g \in \mathcal{G}} c^g y_t^g \quad (27a)$$

$$\text{s.t. } \sum_{g \in \mathcal{G}_i} c^g y_t^g - \sum_{j \in \mathcal{I}_i} f_{ijt} = d_{it} : \lambda_{it} \quad \forall i \quad (27b)$$

$$0 \leq y_t^g \leq y_{max}^g : \mu_t^g \quad \forall g \in \mathcal{G} \quad (27c)$$

$$-x_{ij} \leq f_{ijt} \leq x_{ij} : \alpha_{ijt}^1, \beta_{ijt}^1 \quad \forall (i,j) \in \mathcal{J}^1 \quad (27d)$$

$$-F_{ij}^{max} \leq f_{ijt} \leq F_{ij}^{max} : \alpha_{ijt}, \beta_{ijt} \quad \forall (i,j) \notin \mathcal{J}^1 \quad (27e)$$

A complete nomenclature is provided in the Appendix B.2. The set \mathcal{J}^1 defines all potential transmission lines of the merchant. The upper-level objective is to maximize the (negative) 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, Rebennak, Pardalos, and Iliadis (2012). Constraint (26b) introduces an investment budget, whereas (26c) constrain the maximum capacity to be installed. The lower-level subproblems are simplifications of those in 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.

6.4.3. Results

Note that the investment problem (26)-(27) satisfies Assumption 5.1 but not Assumption 5.2. More specifically, the submatrices \mathbf{C}_{21} and \mathbf{H}_{21} defined in Section 5 do not satisfy Assumption 5.2.2. For this reason, the linearization 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 the full parametric programming without decomposition, since this method does not solve the problem within the time limit.

[Table 3 about here.]

Table 3 and Figure 5 show the solution times as functions of the number of lower-level subproblems.

[Figure 5 about here.]

We note that decomposition by parametric programming has significantly longer solution times for the second case study than for the first; cf. Figure 5 versus Figure 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.

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

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 have 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 can 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.

Acknowledgements

The authors wish to thank two anonymous reviewers for their very useful comments and suggestions, which contributed significantly to improving this paper. Furthermore, T. K. Boomsma acknowledges support from the project AHEAD, Analyses of Hourly Electricity Demand, funded by Energiteknologisk Udviklings- og Demonstrationsprogram (EUDP) under the Danish Energy Agency.

References

- Avis, D. (2000). lrs: A Revised Implementation of the Reverse Search Vertex Enumeration Algorithm. *DMV Seminar*, 29, 177–198.
- Avis, D., & Fukuda, K. (1996). Reverse search for enumeration. *Discrete Applied Mathematics*, 65(1-3), 21–46.
- Bard, J. (1998). *Practical bilevel optimization: applications and algorithms*. Kluwer Academic Press, Boston.
- Baringo, L., & Conejo, A. J. (2011). Wind power investment within a market environment. *Applied Energy*, 88(9), 3239–3247.
- Baringo, L., & Conejo, A. J. (2013). Correlated wind-power production and electric load scenarios for investment decisions. *Applied Energy*, 101, 475–482.
- Ben-Ayed, O., & 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 [Computer software manual]. Retrieved from <https://cran.r-project.org/package=lpSolve>
- Bertsimas, D. P., & Tsitsiklis, J. N. (1997). *Introduction to Linear Optimization*. Athena Scientific.
- Chen, Y., Hobbs, B. F., Leyffer, S., & 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., & Savard, G. (2005). Bilevel programming: A survey. *4or*, 3(2), 87–107.
- Colson, B., Marcotte, P., & Savard, G. (2007). An overview of bilevel optimization. *Annals of Operations Research*, 153(1), 235–256.
- Conejo, A. J., Baringo, L., Kazempour, S. J., & Siddiqui, A. S. (2016). *Investment in Electricity Generation and Transmission*. Springer.
- Danish District Heating Association, & Danish Energy Association. (2016). *Indspil til strategi for fremtidens kraftvarme* (Tech. Rep.). Frederiksberg, Denmark: Danish Energy Association.
- Dempe, S. (2018). *Bilevel optimization: theory, algorithms and applications*. (preprint 2018-11, <http://tu-freiburg.de/fakult1>)
- Dempe, S., Kalashnikov, V., Pérez-Valdés, G. A., & Kalashnykova, N. (2015). *Bilevel Programming Problems*. Springer.
- Dua, V., Bozinis, N. A., & Pistikopoulos, E. N. (2002). A multiparametric programming approach for mixed-integer quadratic engineering problems. *Computers and Chemical Engineering*, 26(4-5), 715–733.
- Ehrenmann, A., & Smeers, Y. (2011). Generation capacity expansion in a risky environment: A stochastic equilibrium analysis. *Operations Research*, 59(6), 1332–1346.
- Faisca, N. P., Dua, V., Rustem, B., Saraiva, P. M., & Pistikopoulos, E. N. (2007). Parametric global optimisation for bilevel programming. *Journal of Global Optimization*, 38(4), 609–623.
- Fortuny-Amat, J., & 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., & Ruiz, C. (2013). *Complementarity Modeling in Energy Markets*. Springer.
- Gabriel, S. A., García-Bertrand, R., Sahakij, P., & 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. *Journal of the Operational Research Society*, 57(8), 995–1004.
- Gabriel, S. A., & 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*. New York: Walter de Gruyter.
- GAMS. (2017). *General Algebraic Modeling System*. Retrieved from gams.com/
- Garcés, L. P., Conejo, A. J., García-Bertrand, R., & Romero, R. (2009). A bilevel approach to transmission expansion planning within a market environment. *IEEE Transactions on Power Systems*, 24(3), 1513–1522.
- Han, J., Liu, G., & Wang, S. (2000). A new descent algorithm for solving quadratic bilevel programming problems. *Acta Mathematicae Applicatae Sinica*, 16(3), 235–244.
- Hobbs, B. F., Metzler, C. B., & Pang, J.-S. (2000). Strategic gaming analysis for electric power systems: An MPEC approach. *IEEE Transactions on Power Systems*, 15(2).
- Joskow, P., & Tirole, J. (2005). Merchant transmission investment. *Journal of Industrial Economics*, 53(2), 233–264.
- Kazempour, S. J., & 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., & Ruiz, C. (2011). Strategic generation investment using a complementarity approach. *IEEE Transactions on Power Systems*, 26(2), 940–948.
- Khanabadi, M., & Ghasemi, H. (2011). Transmission congestion management through optimal transmission switching. *IEEE Power and Energy Society General Meeting*, 1–5.
- Koschker, S., & 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.
- Liu, Y.-H., & 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.
- Luo, Z.-Q., Pang, J.-S., & Ralph, D. (1996). *Mathematical programs with equilibrium constraints*. Cambridge University Press.
- Manas, M., & Nedoma, J. (1968). Finding all vertices of a convex polyhedron. *Numerische Mathematik*, 12, 226–229.
- Mirrlees, J. A. (1999). The theory of moral hazard and unobservable behaviour: Part i. *The Review of Economic Studies*, 66(1), 3–21.
- Nord Pool AS. (2017). *Nord pool*. Retrieved from nordpoolspot.com/
- Outrata, J. V. (1988). A note on the usage of nondifferentiable exact penalties in some special optimization problems. *Kybernetika*, 24(4), 251–258.
- Pineda, S., Bylling, H. C., & Morales, J. M. (2017). Efficiently solving linear bilevel programming problems using off-the-shelf optimization software. *Optimization and Engineering*.
- Robere, R. (2015). vertexenum: Vertex Enumeration of Polytopes [Computer software manual]. Retrieved from <https://cran.r-project.org/package=vertexenum>
- Ruiz, C., & 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.
- Saharidis, G. K. D., Conejo, A. J., & Kozanidis, G. (2013). Exact solution methodologies for Linear and (Mixed) Integer Bilevel Programming. In *Metaheuristics for bi-level optimization* (Vol. 482, pp. 221–245). Springer.
- Scheel, H., & Scholtes, S. (2000). Mathematical Programs with Complementarity Constraints: Stationarity, Optimality, and Sensitivity. *Mathematics of Operations Research*, 25(1), 1–22.
- Shi, C., Lu, J., & Zhang, G. (2005). An extended Kuhn-Tucker approach for linear bilevel programming. *Applied Mathematics and Computation*, 162(1), 51–63.
- Siddiqui, S., & 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.
- Sorokin, A., Rebennak, S., Pardalos, P. M., & 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 Literature*, 8, 94–99.
- Vicente, L. N., & Calamai, P. H. (1994). Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization*, 5(3), 291–306.

Appendix A. Example: Investment in production capacity

This section illustrates how to compute the critical regions and parametric function of the example by Algorithm 4.2.

By introducing slack-variables and converting minimization to maximization, the lower-level problem is

$$\begin{aligned}
 \max \quad & -10y_1 - 12y_2 - 15y_3 \\
 \text{s.t.} \quad & y_1 + y_2 + y_3 + y_4 = 200 \ (\lambda_1^+) \\
 & -y_1 - y_2 - y_3 + y_5 = -200 \ (\lambda_1^-) \\
 & y_1 + y_6 = x \ (\lambda_2) \\
 & y_2 + y_7 = 150 \ (\lambda_3) \\
 & y_3 + y_8 = 100 \ (\lambda_4) \\
 & y_1, \dots, y_8 \geq 0
 \end{aligned}$$

with $\lambda_1^+, \lambda_1^-, \lambda_2, \lambda_3, \lambda_4$ being the complementary dual variables.

Let $x_0 = 0$. An optimal basis is

$$\mathbf{B}_0 = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

with the corresponding Simplex tableau

	z	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	
	-1	0	0	0	0	15	5	3	0	-2550
y_1	0	1	0	0	0	0	1	0	0	0
y_2	0	0	1	0	0	0	0	1	0	150
y_3	0	0	0	1	0	-1	-1	-1	0	50
y_4	0	0	0	0	1	1	0	0	0	0
y_8	0	0	0	0	0	1	1	1	1	50

The primal and dual optimal basic solutions are $y_2^* = 150, y_3^* = 50, y_8^* = 50, y_1^* = y_4^* = \dots = y_7^* = 0$ and $(\lambda_1^-)^* = 15, \lambda_2^* = 5, \lambda_3^* = 3, (\lambda_1^+)^* = \lambda_4^* = 0$, respectively.

Thus, $F(0) = 8760(10 - 15)0 = 0, F'(0) = 8760(10 - 15) = -43,800$ and

$$\Lambda_{\mathbf{B}_0} = \{x : 0 \leq x \leq 250, \mathbf{B}_0^{-1}(200, -200, x, 150, 100)^T \geq 0\} = [0, 50].$$

It is clear from the Simplex tableau that y_3 is the only candidate for leaving basis. We determine if the right-hand-side of the tableau can become zero, i.e. if a neighbor exists, by solving

$$\begin{aligned}
 \min \quad & 50 - x \\
 \text{s.t.} \quad & 0 \leq x \leq 250
 \end{aligned}$$

with $x^* = 50$ and an optimal value of zero. Hence, $\Lambda_{\mathbf{B}_0}$ has a neighbor which is obtained by letting y_7 enter the basis.

With $x_1 = 50$, an optimal basis is

$$\mathbf{B}_1 = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ -1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Recomputing the right-hand-side for $x_1 = 50$, the Simplex tableau becomes

	z	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	
	-1	0	0	0	0	15	5	3	0	-2550
y_1	0	1	0	0	0	0	1	0	0	50
y_2	0	0	1	0	0	0	0	1	0	150
y_3	0	0	0	1	0	-1	-1	-1	0	0
y_4	0	0	0	0	1	1	0	0	0	0
y_8	0	0	0	0	0	1	1	1	1	100

and by re-optimization using the dual Simplex method

	z	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	
	-1	0	0	3	0	12	2	3	0	-2550
y_1	0	1	0	0	0	0	1	0	0	50
y_2	0	0	1	1	0	-1	-1	0	0	150
y_7	0	0	0	-1	0	1	1	1	0	0
y_4	0	0	0	0	1	1	0	0	0	0
y_8	0	0	0	1	0	0	0	0	1	100

The primal and dual optimal basic solutions are $y_1^* = 50, y_2^* = 150, y_3^* = 100, y_3^* = \dots = y_7^* = 0$ and $(\lambda_1^-)^* = 12, \lambda_2^* = 2, (\lambda_1^+)^* = \lambda_3^* = \lambda_4^* = 0$, respectively. We proceed as above.

Appendix B. Nomenclatures of the case studies

B.1. Case study I: Investment in production capacity

Indices

- t Time periods/hours
- g Producers/generators
- i, j Transmission nodes

Parameters

- C_i Annualized investment costs of the strategic investor in node i [\$/MWh]
- c_i^g Production cost of the strategic investor or a rival g in node i [\$/MWh]
- ρ_t Weight of time period t [hours]
- d_{it} Demand in time period t and node i [MWh]

x^{max}	Total maximum capacity of the strategic investor [MW]
$y_i^{max,g}$	Capacity of the strategic investor or a rival g in node i [MW]
B_{ij}	Subceptance of the transmission line from node i to node j
F_{ij}^{max}	Maximum flow on the transmission line from node i to node j [MW]
Primal variables	
x_i	Investment capacity of the strategic producer in node i [MW]
y_{it}^g	Power produced by the strategic investor or a rival g in time period t in node i [MWh]
θ_{it}	Voltage angle in time period t and node i
Dual variables	
λ_{it}	Dual variable of the balancing constraint/market-clearing price in time period t and node i [\$/MWh].
μ_{it}^g	Dual variable of the generation capacity constraint for the strategic investor or a rival g in time period t and node i
η_{ijt}	Dual variable of the transmission capacity constraints from node i to node j in time period t
α_{it}	Dual variable of the minimum voltage angle constraint in time period t and node i
β_{it}	Dual variable of the maximum voltage angle constraint in time period t and node i
γ_{it}	Dual variable of the reference voltage angle constraint in time period t and node i

B.2. Case study II: Transmission investment

Indices

t	Time periods/hours
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]
c^g	Production cost of the strategic investor or a rival g [\$/MWh]
ρ_t	Weight of time period t [hours]
d_{it}	Demand in time period t and node i [MWh]
x^{max}	Total maximum capacity of the strategic investor [MW]
$y^{max,g}$	Capacity of the strategic investor or a rival g [MW]
F_{ij}^{max}	Maximum flow on the transmission line between node i and j [MW]

Primal 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 a rival g in time period t [MWh]
f_{ijt}	Flow on the transmission line between node i and j in time period t [MWh]
θ_{it}	Voltage angle in time period t and node i

Dual variables

λ_{it}	Dual variable of the balancing constraint/market-clearing price in time period t and node i [\$/MWh]
----------------	--

μ_t^g	Dual variable of the generation capacity constraint for the strategic investor or a rival g in time period t
η_{ijt}	Dual variable of the transmission capacity constraints in time period t between node i and node j
α_{ijt}^g	Dual variable of the minimum flow constraint for the strategic investor or a rival g in time period t between node i and node j
β_{ijt}^g	Dual variable of the maximum flow constraint for the strategic investor or a rival g in time period t between node i and node j

Appendix C. MILP formulations

For Case study I, the MILP formulation is the following:

$$\begin{aligned}
\min \quad & \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} d_{it} + \sum_{g \neq 1} \mu_{it}^g y_i^{max,g} + \sum_{j \in \mathcal{I}_i} F_{ij}^{max} \eta_{ijt} + \pi(\alpha_{it} + \beta_{it}) \right) \right) \\
\text{s.t.} \quad & 0 \leq \sum_{i=1}^n x_i \leq x^{max} \\
& \sum_{g \in \mathcal{G}} y_{it}^g - \sum_{j \in \mathcal{I}_i} B_{ij}(\theta_{it} - \theta_{jt}) = d_{it} \quad : \lambda_{it} \quad \forall i \\
& 0 \leq y_{it}^1 \leq x_i \quad \forall i \\
& 0 \leq y_{it}^g \leq 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{I}_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, g \\
& \sum_{j \in \mathcal{I}_i} [B_{ij}(\lambda_i - \lambda_j) + B_{ij}(\eta_{ijt} - \eta_{jit})] - \alpha_{it} + \beta_{it} = 0 \quad \forall t, i \neq \text{ref}. \\
& \sum_{j \in \mathcal{I}_i} [B_{ij}(\lambda_i - \lambda_j) + B_{ij}(\eta_{ijt} - \eta_{jit})] - \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_i^{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}(\theta_{it} - \theta_{jt}) \leq u_{ijt} M_1 \quad \forall t, i, j \in \mathcal{I}_i \\
& \eta_{ijt} \leq (1 - u_{ijt}) M_2 \quad \forall t, i, j \in \mathcal{I}_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
\end{aligned}$$

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.

Tables

Table 1. Number of instances for which a solution method terminates (*solved*), the time limit is reached (*time limit*) and the method returns infeasibility (*local infeasible*), respectively.

	solved	time limit	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

Table 2. Solution times (*time*, in secs) and deviations in the objective function values from the optimal (*dev.*, in %), for different solutions methods and as a function of the number of lower-level problems (*LL*). The solution methods include the full parametric programming method without decomposition (*full*), parametric programming method with decomposition (*decomp.*), the heuristic with decomposition (*heuristic*) and the *MILP*. N/A means that the method did not solve within the time limit of 2 hours.

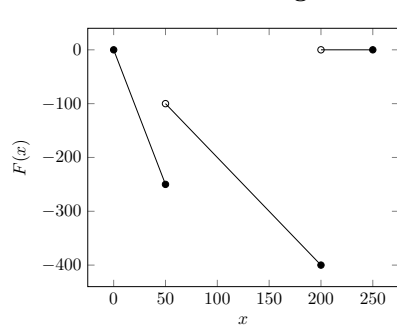
LL	full, time (s)	decomp., time (s)	heuristic, time (s)	heuristic, dev. (%)	MILP, time (s)	MILP, dev. (%)
10	0.57	13.84	0.92	0	0.75	0
20	10.07	16.2	0.85	16.43	0.86	0
30	42.52	19.87	1.12	11.15	0.66	0
40	129.96	28.34	1.45	7.39	0.89	0
50	398.19	16.24	1.62	7.11	1.2	0
60	752.59	21.27	2.01	3.94	0.99	0
70	1905.29	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

Table 3. Solution times (*time*, in secs) and deviations in the objective function values from the optimal (*dev.*, in %), for different solutions methods and as a function of the number of lower-level problems (*LL*). The solution methods include the full parametric programming method without decomposition (*full*), parametric programming method with decomposition (*decomp.*), the heuristic with decomposition (*heuristic*) and the *MILP*. 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.

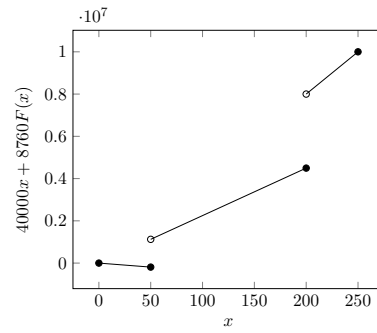
LL	decomp., time (s)	heuristic, time (s)	heuristic, dev. (%)
10	255.37	2.45	0
20	291.18	4.9	0
30	308.74	7.81	0
40	475.1	15.5	0
50	520.52	32.4	0
60	565.94	28.05	0
70	677.45	35.35	0
80	946.85	56.95	0
90	1087.08	64.91	0
100	1169.27	77.56	0
200	2585.28	279.32	0
300	5230.5	636.27	0
400	N/A	1125.35	N/A
500	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

Figures

Figure 1. Parametric function.



(a) Operational (negative) profit.



(b) Upper-level objective function.

Figure 2. Parametric functions.

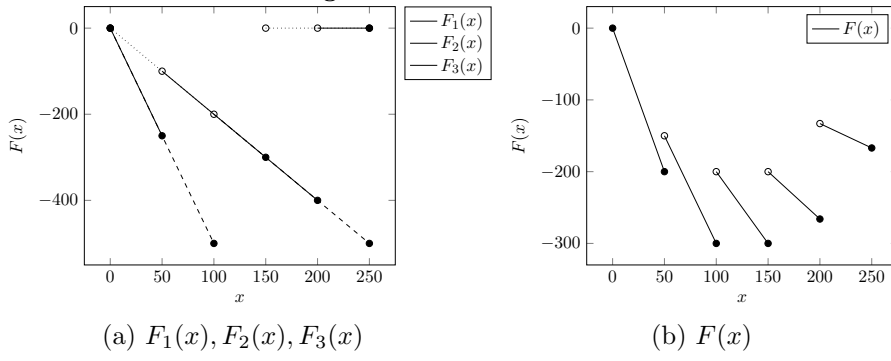
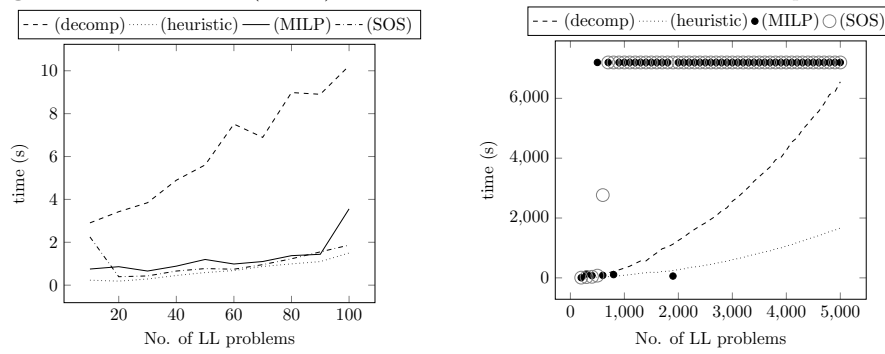


Figure 3. Solution times (in secs) as a function of the number of lower-level subproblems (LL).



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

Figure 4. Feasible solutions found by the parametric programming approach. Divided into intervals of deviation in objective function value from the optimal (in %).

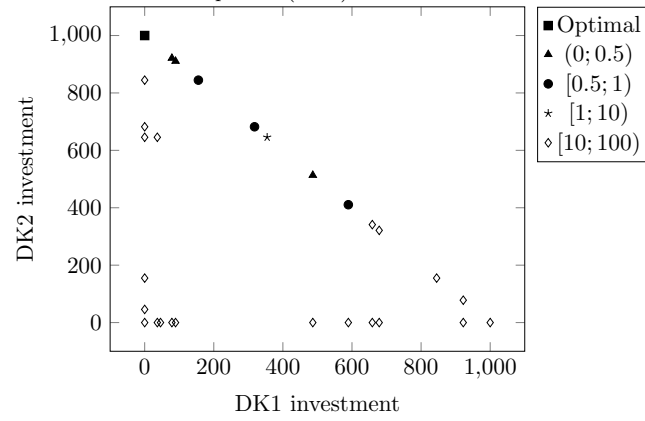


Figure 5. Solution times (in secs) as a function of the number of lower-level problems (LL).

