

Kleinert, Thomas; Schmidt, Martin

Article — Published Version

Why there is no need to use a big-M in linear bilevel optimization: a computational study of two ready-to-use approaches

Computational Management Science

Provided in Cooperation with:

Springer Nature

Suggested Citation: Kleinert, Thomas; Schmidt, Martin (2023) : Why there is no need to use a big-M in linear bilevel optimization: a computational study of two ready-to-use approaches, Computational Management Science, ISSN 1619-6988, Springer, Berlin, Heidelberg, Vol. 20, <https://doi.org/10.1007/s10287-023-00435-5>

This Version is available at:

<https://hdl.handle.net/10419/307511>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by/4.0/>



Why there is no need to use a big- M in linear bilevel optimization: a computational study of two ready-to-use approaches

Thomas Kleinert^{1,2} · Martin Schmidt³

Received: 7 June 2022 / Accepted: 28 December 2022 / Published online: 7 February 2023
© The Author(s) 2023

Abstract

Linear bilevel optimization problems have gained increasing attention both in theory as well as in practical applications of Operations Research (OR) during the last years and decades. The latter is mainly due to the ability of this class of problems to model hierarchical decision processes. However, this ability makes bilevel problems also very hard to solve. Since no general-purpose solvers are available, a “best-practice” has developed in the applied OR community, in which not all people want to develop tailored algorithms but “just use” bilevel optimization as a modeling tool for practice. This best-practice is the big- M reformulation of the Karush–Kuhn–Tucker (KKT) conditions of the lower-level problem—an approach that has been shown to be highly problematic by Pineda and Morales (2019). Choosing invalid values for M yields solutions that may be arbitrarily bad. Checking the validity of the big- M s is however shown to be as hard as solving the original bilevel problem in Kleinert et al. (2019). Nevertheless, due to its appealing simplicity, especially w.r.t. the required implementation effort, this ready-to-use approach still is the most popular method. Until now, there has been a lack of approaches that are competitive both in terms of implementation effort and computational cost. In this note we demonstrate that there is indeed another competitive ready-to-use approach: If the SOS-1 technique is applied to the KKT complementarity conditions, adding the simple additional root-node inequality developed by Kleinert et al. (2020) leads to a competitive performance—without having all the possible theoretical disadvantages of the big- M approach.

Keywords Bilevel optimization · Big- M · SOS-1 · Valid inequalities · Computational analysis

Mathematics Subject Classification 90Cxx · 90-08 · 90C11 · 90C46

✉ Martin Schmidt
martin.schmidt@uni-trier.de

Extended author information available on the last page of the article

1 The big- M reformulation: convenient but error-prone

We consider linear bilevel problems of the form

$$\min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} c^\top x + d^\top y \quad \text{s.t.} \quad Ax + By \geq a, \quad y \in \Psi(x), \quad (1)$$

where $\Psi(x)$ denotes the set of optimal solutions of the x -parameterized linear program

$$\max_y f^\top y \quad \text{s.t.} \quad Dy \leq b - Cx, \quad (2)$$

with $c \in \mathbb{R}^n$, $d, f \in \mathbb{R}^m$, $A \in \mathbb{R}^{k \times n}$, $B \in \mathbb{R}^{k \times m}$, $a \in \mathbb{R}^k$, $C \in \mathbb{R}^{\ell \times n}$, $D \in \mathbb{R}^{\ell \times m}$, and $b \in \mathbb{R}^\ell$. In this setting, the upper-level player (or leader) anticipates the optimal reaction y of the lower-level player (or follower). The set of optimal solutions $\Psi(x)$ is not a singleton if the follower is indifferent for a given x . In this case, Problem (1) establishes the so-called optimistic solution, i.e., the leader may select the solution $y \in \Psi(x)$ that is the most favorable one for the upper-level problem; see Dempe (2002). In general, bilevel problems are intrinsically nonconvex due to their hierarchical structure and even linear bilevel problems (1) are known to be strongly NP-hard; see Hansen et al. (1992). Although specialized techniques for this problem class exist, the “best-practice” still is the well-known mixed-integer single-level reformulation that relies on big- M values; see, e.g., Baringo and Conejo (2011), Garces et al. (2009), Baringo and Conejo (2011), Wogrin et al. (2011), Kazempour et al. (2011, 2012), Kazempour and Conejo (2012), Jenabi et al. (2013), Wogrin et al. (2013), Pozo et al. (2013), Pisciella et al. (2016), Maurovich-Horvat et al. (2015), Morales et al. (2014), Jaber Valinejad (2015), and the references therein for bilevel optimization problems in the field of power systems that are tackled with the classic big- M approach.

1.1 The method

This single-level reformulation is derived as follows. First, the lower-level problem (2) is replaced by its necessary and sufficient Karush–Kuhn–Tucker (KKT) conditions. This yields the mathematical program with complementarity constraints (MPCC)

$$\min_{x, y, \lambda} c^\top x + d^\top y \quad (3a)$$

$$\text{s.t.} \quad Ax + By \geq a, \quad Cx + Dy \leq b, \quad (3b)$$

$$\lambda \in \Omega_D := \{\lambda \geq 0 : D^\top \lambda = f\}, \quad (3c)$$

$$\lambda^\top (b - Cx - Dy) = 0, \quad (3d)$$

in which Constraint (3c) denotes dual feasibility and (3d) are the KKT complementarity conditions of the lower-level problem. Next, the complementarity conditions are replaced by the mixed-integer reformulation

$$b - Cx - Dy \leq M_P(1 - u), \quad \lambda \leq M_D u, \quad u \in \{0, 1\}^\ell, \quad (4)$$

with sufficiently large big- M constants M_P and M_D . Problem (3) as well as the mixed-integer reformulation (4) have been mentioned first by Fortuny-Amat and McCarl (1981). The advantage of this approach for OR practitioners is obvious: The single-level problem (3) in which the complementarity conditions (3d) are replaced by (4) can be easily implemented and the resulting model can be solved without further ado by standard mixed-integer solvers. However, this approach has some severe issues.

On the one hand, choosing the big- M constants too small can result in suboptimal solutions of Problem (1) as shown in Pineda and Morales (2019). Their counterexample also shows that the loss of optimality can be arbitrarily large in terms of the resulting objective function values. Unfortunately, many works do not discuss the selection of the constants at all or use trial-and-error procedures without any guarantee that the derived values yield a correct reformulation; see, e.g., the references compiled in Pineda and Morales (2019). Recently, it is shown in Kleinert et al. (2019) that verifying the correctness of given big- M values is as hard as solving the original bilevel problem, which is strongly NP-hard in general. Thus, unless valid big- M constants can be derived from problem-specific knowledge as it is done, e.g., in Böttger et al. (2021); Kleinert and Schmidt (2019) or Siddiqui and Gabriel (2013), the stated mixed-integer reformulation cannot be expected to yield correct results. On the other hand, large values of M_P and M_D may cause numerical instabilities. In the extreme case, too large values can indeed yield “solutions” that are actually infeasible for the original bilevel problem due to the products in (4) of very large constants and binary variables that are relaxed up to a certain tolerance by mixed-integer solvers. We illustrate the impact of different values for M_P and M_D in the following computational study.

1.2 Performance and reliability evaluation

We first briefly describe the computational setup that we use throughout this work as well as the details of our evaluation procedure. All computations in this and the remaining sections are carried out on a compute cluster with Xeon E3-1240 v6 CPUs at 3.7 GHz and 32 GB RAM; see Regionales Rechenzentrum Erlangen (2020) for more details. The code used for the computational studies in this work is implemented in C++-11 and has been compiled with GCC 7.3.0. All optimization problems are solved by CPLEX 12.10.

For the evaluation of the computational results, we mainly use performance profiles according to Dolan and Moré (2002). For every test instance i , we compute ratios $r_{i,s} = t_{i,s} / \min\{t_{i,s} : s \in S\}$, where S is the set of solution approaches under consideration and $t_{i,s}$ is the running time required by approach $s \in S$ for instance i .

Each performance profile plot in this work shows the percentage of instances (y-axis) for which the performance ratio $r_{i,s}$ of approach s is within a factor $\tau \geq 1$ (log-scaled x-axis) of the best possible ratio.

For our analysis we use the linear bilevel instances described in Kleinert and Schmidt (2020). All of these instances stem from relaxing the integrality conditions of mixed-integer bilevel instances from the literature. We denote the various instance classes along with a reference to its original mixed-integer test set and relevant characteristics in Table 1. For our analysis, we clean up this instance set in the following way. We exclude 353 instances that can be solved by any of the methods tested in this work in less than 5 s. In addition, we exclude 26 instances that cannot be solved by any of the tested methods within the time limit of 1 h. This yields a total number of 698 instances in the cleaned test set \mathcal{I} .

We are now ready to compare the big- M approach for the choices $M = M_D = M_P \in \{10^4, 10^5, 10^6\}$. For better readability, we refer to the three instantiations by BigM-4, BigM-5, and BigM-6. As just discussed, the big- M approach may deliver wrong “solutions” in case the value M is chosen too small or “too large”. The latter may result in infeasible points because the lower-level complementarity may in fact not be fulfilled due to numerical tolerances. In order to circumvent this behavior, we tightened the integer feasibility tolerance of CPLEX to 10^{-9} . An ex-post evaluation of the complementarity conditions revealed that with this setting, all solutions fulfill every complementarity condition up to a tolerance of 10^{-6} . Thus, we consider these solutions to be feasible. In contrast, too small values for M may produce suboptimal solutions. Therefore, we run the following sanity check for every instance in \mathcal{I} . Let F_i^s be the objective value of the best feasible solution, i.e., the best upper bound, and let \underline{F}_i^s be the best lower bound found by approach $s \in \{\text{BigM-4, BigM-5, BigM-6}\}$ for instance $i \in \mathcal{I}$. We set unavailable values to $+\infty$ and $-\infty$, respectively. Further, let F_i^* and \underline{F}_i^* be the best upper and lower bound found by a provably correct solution approach. We will consider this provably correct approach as a black box for the moment and discuss it in more detail in the next section. We then check for every instance i and approach s , if

$$F_i^s \geq \underline{F}_i^* \quad \text{and} \quad \underline{F}_i^s \leq F_i^*$$

hold. If this is not the case, we consider instance i as not solved by s . Out of the 698 instances in \mathcal{I} , this happened 29 times for BigM-6, 50 times for BigM-5, and 147 times for BigM-4.

Figure 1 shows two performance profiles of the running times of the methods BigM-4, BigM-5, and BigM-6. The left performance profile is based on those 377 instances in \mathcal{I} that all three methods solve. Apparently, a lower big- M value is beneficial w.r.t. the running time and BigM-4 dominates the other two methods. Thus, there is an incentive to choose a small value of M . However, the picture changes, if we consider the 549 instances in \mathcal{I} that at least one of the three methods solves; see Fig. 1 (right). It can be seen that BigM-6 solves the largest number of instances among the three tested approaches. To be more specific, BigM-6 solves 524, BigM-5 solves 503, and BigM-4 solves 411 instances. The reason is, as discussed above, that for a smaller value of M , the ex-post optimality check fails more often. This results in more instances counted as

Table 1 Instance classes with relevant sizes and references

	Reference	# Instances		n		m		k		ℓ	
		Total	In \mathcal{I}	min	max	min	max	min	max	min	max
CLIQUE	Fischetti et al. (2018)	60	60	546	1593	586	1653	1	1	1326	3363
IMKP	Fischetti et al. (2019)	144	134	10	105	10	105	1	29	11	106
INTER-ASSIG	DeNegre (2011)	24	19	25	25	25	25	1	1	45	45
INTER-CLIQUE	Tang et al. (2016)	80	80	19	94	27	109	1	1	53	280
INTER-KP	DeNegre (2011)	99	65	10	50	10	50	1	1	11	51
KP	Fischetti et al. (2018)	450	219	100	500	100	500	1	1	101	501
MIPLIB	Fischetti et al. (2019)	60	22	3	78734	2	78733	0	0	15	4944
XULARGE	Xu and Wang (2014)	60	60	500	1000	500	1000	200	400	200	400
XUWANG	Fischetti et al. (2017)	100	39	10	460	10	460	4	184	4	184

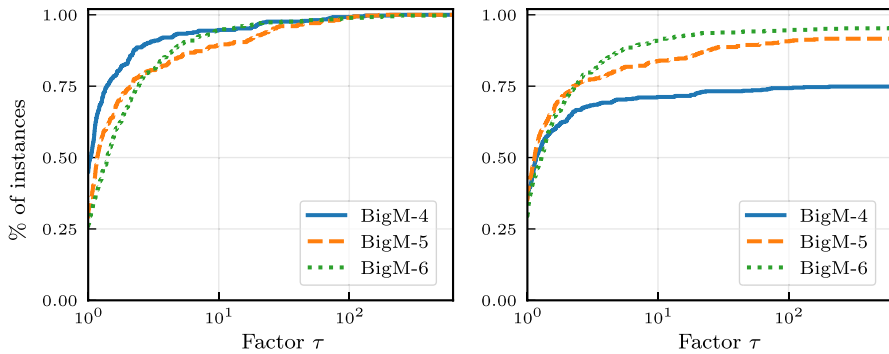


Fig. 1 Performance profile of running times of the big- M -based methods for the instances in \mathcal{I} that all three methods solve (left) and at least one method solves (right)

not solved. Note that there is, however, no dominance across the three methods w.r.t. the solved instances. There are 18 instances that BigM-5 solves but BigM-6 does not and 21 instances that BigM-4 solves but BigM-5 does not. Consequently, although it is not the fastest method, from a practical point of view, BigM-6 is the best choice. This is in line with results of a comparison of various values for M on a different test set in Pineda et al. (2018). We recap however that even BigM-6 produces 29 “solutions” that are indeed not optimal.

2 The SOS1 reformulation: a lame duck?

Another way to solve the MPCC (3) is to omit the complementarity conditions (3d) initially and then branch on them instead.

2.1 The method

A first sketch of this approach has been proposed by Fortuny-Amat and McCarl (1981) and a more detailed evaluation along with first numerical results can be found in Bard and Moore (1990). A modern and convenient way to apply this method is to exploit special ordered sets of type 1 (SOS1). Such sets are introduced in Beale and Tomlin (1970) and require that at most one variable of a set of variables takes a nonzero value. With this, the complementarity conditions (3d) can be rephrased as follows:

$$s_i = (b - Cx - Dy)_i, \quad \{s_i, \lambda_i\} \text{ is SOS1}, \quad i = 1, \dots, \ell. \quad (5)$$

This technique is proposed in a more general MPCC context in Siddiqui and Gabriel (2013) and used in a bilevel context in Pineda et al. (2018). We highlight that it is big- M -free and fairly easy to implement. Modern mixed-integer solvers handle SOS1 constructs by automatically reformulating it to the mixed-integer formulation (4) if provably correct bounds on s_i and λ_i are available or by branching on s_i and λ_i otherwise. In this way, the benefits of using a highly evolved mixed-integer

solver can be exploited, while the correctness of the obtained solutions is guaranteed—in contrast to the approach stated in Sect. 1. However, the theoretical worst-case complexity is the same for both approaches. We label this big- M -free method as SOS1 and evaluate its performance in comparison to BigM-6 in the following.

2.2 Performance evaluation

All results presented in this section follow the setup described in Sect. 1.2. Figure 2 shows a performance profile of the running times of BigM-6 and SOS1 on the 566 instances that at least one of the two methods solves. It can be seen that BigM-6 is the fastest method for more than 50% of the instances (see the leftmost point of the solid curve). In addition, it solves around 90% of the instances (see the rightmost point of the solid curve), although some “solved” instances are considered as not solved due to the ex-post optimality check described in Sect. 1.2. The valid black box used in this optimality check is exactly the SOS1-based approach. According to Fig. 2, BigM-6 can be considered the distinct winner approach, which is in line with the analysis in Pineda et al. (2018). Overall, this justifies why the big- M approach equipped with a large constant M is chosen instead of SOS1 in a lot of applications of OR.

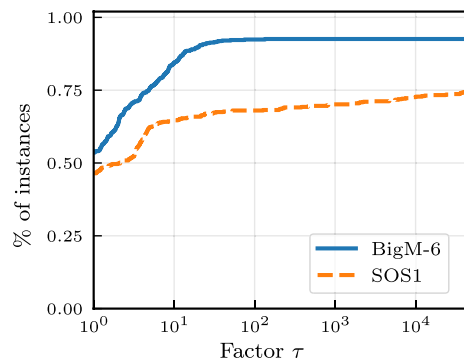
3 The game changer: valid inequalities

The results presented up to this point only re-iterate folklore knowledge: Although it has its downsides, the big- M approach is the method of choice in practical linear bilevel optimization.

This section aims to change this thinking. We show that adding a simple inequality to the MPCC (3) changes the results drastically. This inequality has been proposed recently in Kleinert et al. (2020) and exploits the strong-duality condition of the follower problem. We briefly summarize its derivation in the following. For any leader decision x , the dual follower problem is given by

$$\min_{\lambda} \quad \lambda^T(b - Cx) \quad \text{s.t.} \quad \lambda \in \Omega_D = \{\lambda \geq 0 : D^T \lambda = f\}.$$

Fig. 2 Performance profile of running times of the methods BigM-6 and SOS1 for the instances in \mathcal{I} that at least one method solves



For every lower-level primal-dual feasible point (y, λ) , weak duality

$$f^\top y \leq \lambda^\top b - \lambda^\top Cx$$

holds. Thus, strong duality can be enforced by

$$f^\top y \geq \lambda^\top b - \lambda^\top Cx.$$

This is a bilinear constraint with products of primal leader variables x and dual follower variables λ . However, one can derive a linear valid inequality from it by replacing each term $C_{i \cdot} x$ with an upper bound $C_i^+ \geq C_{i \cdot} x$; see Kleinert et al. (2020). This yields the inequality

$$f^\top y \geq \lambda^\top b - \lambda^\top C^+, \quad (6)$$

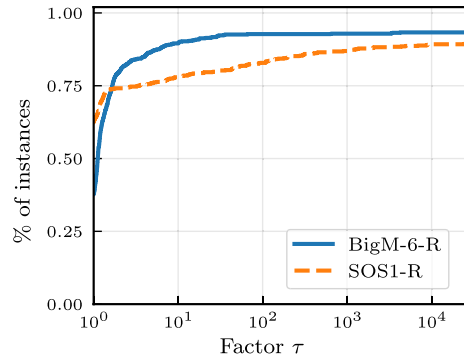
in which C^+ denotes the vector of upper bounds C_i^+ . The linearization of the bilinear term $\lambda^\top Cx$ in (6) can be seen as a special case of a McCormick inequality McCormick (1976) as discussed in Kleinert et al. (2020). Of course, other techniques to deal with these bilinearities are possible as well such as spatial branching Horst and Tuy (2013) or other iterative approaches based on convex optimization, see, e.g., Constante-Flores et al. (2022). The bounds required in (6) can be obtained, e.g., by exploiting variable bounds on x or by solving auxiliary linear problems

$$\max \quad C_{i \cdot} x \quad \text{s.t.} \quad Ax + By \geq a, \quad Cx + Dy \geq b, \quad (7)$$

see also Kleinert et al. (2020). This requires the joint feasible set $\{(x, y) : Ax + By \geq a, Cx + Dy \geq b\}$ of the upper and lower level to be bounded, which is the case for every tested instance in our instance set \mathcal{I} . Although solving the additional auxiliary LPs (7) can be done in polynomial time in general, it might be time consuming in practice. However, it has been shown in Kleinert et al. (2020) to pay off to add Inequality (6) to the SOS1 reformulation of Problem (1). A preliminary computational analysis revealed that this strategy is also very effective for the BigM-6 approach. Consequently, 688 instances in \mathcal{I} are solved by at least one of the two methods BigM-6-R and SOS1-R compared to 566 for the methods BigM-6 and SOS1. More precisely, BigM-6-R solves 642 instances compared to 524 instances solved by BigM-6 and SOS1-R solves 615 instances compared to 421 instances solved by SOS1. Note that the “R” denotes that the valid inequality (6) has been added at the root node. In more detail, this means that the problem solved by the BigM-6 method is Problem (3) with (3d) replaced with (4) and the additional constraint (6). The SOS1 method solves Problem (3), extended by (6), with (3d) replaced with (5).

Figure 3 shows a performance profile that compares the running times of BigM-6-R and SOS1-R on the 688 instances that at least one method solves. We observe several interesting aspects. First, the SOS1-based approach is the faster method for over 65% of the instances (see the leftmost point of the dashed curve). This is very much in contrast to the results without the valid inequality. Second, the reliability advantage of the big- M -based approach decreases significantly.

Fig. 3 Performance profile of running times of the methods BigM-6-R and SOS1-R for the instances in \mathcal{I} that at least one method solves



BigM-6-R solves 73 instances that SOS1-R cannot solve within the time limit, but SOS1-R also solves 46 instances that BigM-6-R cannot solve.

According to Fig. 3, the only reason for choosing BigM-6-R over the SOS1-based approach is that it “solves” more instances. However, this is a very ambivalent statement, because there is always the possibility that the “solutions” obtained by big- M -based methods are simply wrong due to a too small or too large value for M . In contrast, we highlight that the SOS1-based method either provides the correct optimal solution or terminates with a suboptimal solution and a trustworthy optimality gap. In fact, if we look at the 73 instances that only BigM-6-R solves, it turns out that the objective function values of the solutions provided by the SOS1-based solver always exactly match the objective function value of the “globally optimal” solutions provided by BigM-6-R.

4 Practical implications for linear bilevel optimization

The results presented in this note indicate the following. In general, the big- M approach is faster and “solves” more instances, especially if the value of M can be chosen small. Thus, whenever one is able to determine valid values of M , e.g., based on problem-specific knowledge, then one should use the big- M approach. On the other hand, if one does not have such problem-specific knowledge, one really should not use the big- M approach. In this note, we showed that there is also no need anymore to use it: The SOS1 approach equipped with the discussed root-node inequalities lead to comparable results on our test set. Moreover, this extended SOS1 approach is also easy to implement and, especially in the light of the validity of the obtained results, the small amount of extra work is worth the effort. To sum up, we hope that these results will change the “best-practice” in applied linear bilevel optimization and thus will lead to more trustworthy results of bilevel models in OR applications.

Acknowledgements This research has been performed as part of the Energie Campus Nürnberg and is supported by funding of the Bavarian State Government. The authors thank the DFG for their support within project A05 and B08 in CRC TRR 154. Finally, we thank Fränk Plein for his contributions to the implementation of the SOS1 approach and the valid inequality.

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bard JF, Moore JT (1990) A branch and bound algorithm for the bilevel programming problem. *SIAM J Sci Stat Comput* 11(2):281–292. <https://doi.org/10.1137/0911017>
- Baringo L, Conejo AJ (2011) Wind power investment within a market environment. *Appl Energy* 88(9):3239–3247. <https://doi.org/10.1016/j.apenergy.2011.03.023>
- Beale EML, Tomlin JA (1970) Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables. In: *Proceedings of the fifth international conference on operational research*. J. Lawrence (eds.) Tavistock Publications, 447–454
- Böttger T, Grimm V, Kleinert T, Schmidt M (2021) The Cost of Decoupling Trade and Transport in the European Entry-Exit Gas Market with Linear Physics Modeling. *European J Oper Res*. <https://doi.org/10.1016/j.ejor.2021.06.034>
- Constante-Flores G, Conejo AJ, Constante-Flores S (2022) Solving certain complementarity problems in power markets via convex programming. In: *TOP* 30.3, pp. 465–491. <https://doi.org/10.1007/s11750-022-00627-3>
- Dempe S (2002) *Foundations of Bilevel Programming*. Springer. <https://doi.org/10.1007/b101970>
- DeNegre S (2011) *Interdiction and discrete bilevel linear programming*. PhD thesis. Lehigh University
- Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. *Math Program* 91(2):201–213. <https://doi.org/10.1007/s101070100263>
- Fischetti M, Ljubic I, Monaci M, Sinnl M (2017) A new general-purpose algorithm for mixed-integer bilevel linear programs. *Oper Res* 65(6):1615–1637. <https://doi.org/10.1287/opre.2017.1650>
- Fischetti M, Ljubic I, Monaci M, Sinnl M (2019) Interdiction games and monotonicity, with application to knapsack problems. *INFORMS J Comput* 31(2):390–410. <https://doi.org/10.1287/ijoc.2018.0831>
- Fischetti M, Monaci M, Sinnl M (2018) A dynamic reformulation heuristic for Generalized Interdiction Problems. *Eur J Oper Res* 267(1):40–51. <https://doi.org/10.1016/j.ejor.2017.11.043>
- Fortuny-Amat J, McCarl B (1981) A representation and economic interpretation of a two-level programming problem. *J Oper Res Soc* 32(9):783–792. <https://doi.org/10.1057/jors.1981.156>
- Garces LP, Conejo AJ, Garcia-Bertrand R, Romero R (2009) A bilevel approach to transmission expansion planning within a market environment. *IEEE Trans Power Syst* 24(3):1513–1522. <https://doi.org/10.1109/TPWRS.2009.2021230>

- Hansen P, Jaumard B, Savard G (1992) New branch-and-bound rules for linear bilevel programming. *SIAM J Sci Stat Comput* 13(5):1194–1217. <https://doi.org/10.1137/0913069>
- Horst R, Tuy H (2013) Global optimization: deterministic approaches. Springer Science & Business Media. <https://doi.org/10.1007/978-3-662-03199-5>
- Jaber Valinejad TB (2015) Generation expansion planning in electricity markets: a novel framework based on dynamic stochastic MPEC. *Int J Electr Power Energy Syst* 70:108–117. <https://doi.org/10.1016/j.ijepes.2015.02.002>
- Jenabi M, Fatemi Ghomi SMT, Smeers Y (2013) Bi-level game approaches for coordination of generation and transmission expansion planning within a market environment. *IEEE Trans Power Syst* 28(3):2639–2650. <https://doi.org/10.1109/TPWRS.2012.2236110>
- Kazempour SJ, Conejo AJ (2012) Strategic generation investment under uncertainty via benders decomposition. *IEEE Trans Power Syst* 27(1):424–432. <https://doi.org/10.1109/TPWRS.2011.2159251>
- Kazempour SJ, Conejo AJ, Ruiz C (2011) Strategic generation investment using a complementarity approach. *IEEE Trans Power Syst* 26(2):940–948. <https://doi.org/10.1109/TPWRS.2010.2069573>
- Kazempour SJ, Conejo AJ, Ruiz C (2012) Strategic generation investment considering futures and spot markets. *IEEE Trans Power Syst* 27(3):1467–1476. <https://doi.org/10.1109/TPWRS.2011.2182664>
- Kleinert T, Labbé M, Plein F, Schmidt M (2019) There's no free lunch: on the hardness of choosing a correct big- M in bilevel optimization. *Oper Res*. <https://doi.org/10.1287/opre.2019.1944>
- Kleinert T, Labbé M, Plein F, Schmidt M (2020). Closing the gap in linear bilevel optimization: a new valid primal-dual inequality. Tech Rep. http://www.optimization-online.org/DB_HTML/2020/06/7826.html. Submitted
- Kleinert T, Schmidt M (2019) Global optimization of multilevel electricity market models including network design and graph partitioning. *Discret Optim* 33:43–69. <https://doi.org/10.1016/j.disopt.2019.02.002>
- Kleinert T, Schmidt M (2020) Computing feasible points of bilevel problems with a penalty alternating direction method. *INFORMS J Comput*. <https://doi.org/10.1287/ijoc.2019.0945>
- Maurovich-Horvat L, Boomsma TK, Siddiqui AS (2015) Transmission and wind investment in a deregulated electricity industry. *IEEE Trans Power Syst* 30(3):1633–1643. <https://doi.org/10.1109/TPWRS.2014.2367107>
- McCormick GP (1976) Computability of global solutions to factorable nonconvex programs: part I-Convex underestimating problems. *Math Program* 10(1):147–175. <https://doi.org/10.1007/BF01580665>
- Morales JM, Zugno M, Pineda S, Pinson P (2014) Electricity market clearing with improved scheduling of stochastic production. *Eur J Oper Res* 235(3):765–774. <https://doi.org/10.1016/j.ejor.2013.11.013>
- Pineda S, Bylling H, Morales J (2018) Efficiently solving linear bilevel programming problems using off-the-shelf optimization software. *Optim Eng* 19(1):187–211. <https://doi.org/10.1007/s11081-017-9369-y>
- Pineda S, Morales JM (2019) Solving linear bilevel problems using big- M s: not all that glitters is gold. *IEEE Trans Power Syst*. <https://doi.org/10.1109/TPWRS.2019.2892607>
- Pisciella P, Bertocchi M, Vespucci MT (2016) A leader-followers model of power transmission capacity expansion in a market driven environment. *CMS* 13(1):87–118. <https://doi.org/10.1007/s10287-014-0223-9>
- Pozo D, Sauma EE, Contreras J (2013) A three-level static MILP model for generation and transmission expansion planning. *IEEE Trans Power Syst* 28(1):202–210. <https://doi.org/10.1109/TPWRS.2012.2204073>
- Regionales Rechenzentrum Erlangen (2020). Woodcrest Cluster. <https://www.anleitungen.rze.fau.de/hpc/woody-cluster/> (visited on 08/03/2020)
- Siddiqui S, Gabriel SA (2013) An SOS1-based approach for solving MPECs with a natural gas market application. *Netw Spat Econ* 13(2):205–227. <https://doi.org/10.1007/s11067-012-9178-y>
- Tang Y, Richard J-PP, Smith JC (2016) A class of algorithms for mixed-integer bilevel min-max optimization. *J Global Optim* 66(2):225–262. <https://doi.org/10.1007/s10898-015-0274-7>
- Wogrin S, Barquin J, Centeno E (2013) Capacity expansion equilibria in liberalized electricity markets: an EPEC approach. *IEEE Trans Power Syst* 28(2):1531–1539. <https://doi.org/10.1109/TPWRS.2012.2217510>
- Wogrin S, Centeno E, Barquin J (2011) Generation capacity expansion in liberalized electricity markets: a stochastic MPEC approach. *IEEE Trans Power Syst* 26(4):2526–2532. <https://doi.org/10.1109/TPWRS.2011.2138728>

Xu P, Wang L (2014) An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. *Comput Oper Res* 41:309–318. <https://doi.org/10.1016/j.cor.2013.07.016>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Thomas Kleinert^{1,2}  · Martin Schmidt³ 

Thomas Kleinert
thomas.kleinert@fau.de

¹ Friedrich-Alexander-Universität Erlangen-Nürnberg, Discrete Optimization, Cauerstr. 11, 91058 Erlangen, Germany

² Energie Campus Nürnberg, Fürther Str. 250, 90429 Nürnberg, Germany

³ Department of Mathematics, Trier University, Universitätsring 15, 54296 Trier, Germany