Advanced Feasibility Cuts in Decoupled Cooperative Optimization of Power Flow

Tim Varelmann[1], Adrian W. Lipow[1], Michael Baldea[2,3], and Alexander Mitsos[4,1,5,*]

[1]Process Systems Engineering (AVT.SVT), RWTH Aachen University, 52074 Aachen, Germany

[2]McKetta Department of Chemical Engineering, The University of Texas at Austin, Austin, TX 78712, USA

[3]Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, Texas 78712, USA

[4]JARA-CSD, 52056 Aachen, Germany

[5]Institute of Energy and Climate Research, Energy Systems Engineering (IEK-10), Forschungszentrum Jülich GmbH, 52425 Jülich, Germany

[*]corresponding author

Abstract

Solving the optimal power flow problem in a cooperative way between power generators and industrial users can reduce electricity costs more effectively than price-based demand side management. In previous work, we recently developed a decomposition-based algorithm based on Benders-type cuts to solve cooperative optimal power flow

problems, which allows to protect sensitive load information. However, the algorithm suffers from unfavorable scaling behavior when the number of cooperating load entities increases. Herein, we improve the quality of the cutting planes that cooperating electricity users generate for the grid in lieu of sharing their dynamic process models. After comparing different cutting strategies and combinations thereof, we develop a tailored cutting strategy. This strategy improves the scaling behavior of our decomposition-based algorithm with multiple cooperating electricity user locations drastically. We obtain speedups factors up to almost 60 compared to our algorithm with the initial cutting strategy.

keywords: demand-side management; cooperative optimal power flow; congestion management; cutting plane algorithm

1 Introduction

The operation of the power grid is nowadays subject to uncertainty from both the demand and generation sides, the latter driven by a rapidly increasing contribution of (inherently time variable) renewable resources (such as wind and solar photovoltaics) to the power generation mix. Increasing the flexibility of the demand side of the grid becomes essential in ensuring that power supply and demand are balanced at all times. The umbrella term demand side management (DSM) refers to all activities that alter (most likely, temporally shift) electrical power demand with the goal of aligning supply and demand, particularly at times when the latter is at its peak (Baldea, 2017; Zhang and Grossmann, 2016).

Industrial electricity users are well-suited for engaging in such initiatives. Their loads are large, localized, and often flexible. A broad variety of manufacturing processes are considered applicable for DSM (Palensky and Dietrich, 2011): chlor-alkali processes (Hoffmann et al., 2020; Roh et al., 2019; Otashu and Baldea, 2019) – which we also consider in this article –, air separation (Ierapetritou et al., 2002; Zhang et al., 2015; Caspari et al., 2019; Tsay et al., 2019), aluminum electrolysis (Todd et al., 2008; Zhang and Hug, 2015), cement milling

(Summerbell et al., 2017; Spangenberg et al., 2015), steel production (Hadera et al., 2015; Castro et al., 2013, 2020), and glass production (Seo et al., 2020) are some examples.

However, the vast majority of DSM literature assumes electricity prices as given and known with reasonable accuracy, and the proposed approaches focus on modulating the operation of the industrial entity to exploit patterns in these prices. Assuming that users are purely price takers does not constitute a true cooperation between grid operators and energy-intensive processes that can have beneficial effects for both of them. In particular, electricity grid operators have to make assumptions about user demand in order to plan their generation. If the electricity prices result from such an assumption-based plan, beneficial effects of cooperation can not influence the prices. To deal with imprecisions in the assumed demand, grid operators have to allocate balancing reserves. An example for an undesired second-order effect of price-based DSM is the formation of rebound peaks in electricity demand (McAuliffe and Rosenfeld, 2004; Li et al., 2012).

With the aim of exploring the benefits of managing electricity generation, transmission and demand in close cooperation between grid operators and (large) electricity users (particularly, industrial facilities), we recently presented the concept of cooperative optimal power flow (OPF) (Otashu et al., 2021). Broadly speaking, it consists of ceding operational control of the facility to the grid operators. The latter are allowed to determine the production schedule of flexible industrial processes in a way that minimizes operating cost for the grid. In our initial effort (Otashu et al., 2021), letting the grid decide the production schedule required the cooperating industrial user to reveal a dynamic model of its operation to the grid, which raised a confidentiality issue. With follow-up work (Varelmann et al., 2021), inspired by the concept of Benders Decomposition (Benders, 1962) (BD), we proposed replacing the exchange of such confidential information between grid and cooperating processes with the exchange of temporal power demand profile definitions via cutting planes that only reflect information in the space of (time dependent) power profiles. In this way, we were able to reproduce our previous results (Otashu et al., 2021) by iteratively refining an outer approx-

imation of the region of power profiles that allow feasible operation of a chlor-alkali process in the power grid.

A remaining challenge with our decomposition-based approach is the unfavorable scaling behavior of the algorithm with the number of buses with cooperating loads (Varelmann et al., 2021). The significant increase in computational effort reported in our article (Varelmann et al., 2021) made the algorithm impractical for real-time use in large system instances. In this article, we revisit our strategy to compute the feasibility cuts that refine the aforementioned outer approximation of the region of power profiles. As a consequence of a more effective cutting strategy, the scaling behavior of the algorithm improves as well, such that it does not impede on practical implementations.

The broad applicability and success of BD has led to a plethora of improvements to virtually all aspects of the original algorithm (Benders, 1962). An early improvement to the cutting plane generation process was presented by Magnanti and Wong (1981). They introduced the notion of Pareto-optimal cuts and exploited useful properties of core-points of the feasible region of LPs. Papadakos (2008) simplified their approach later on by showing that just one optimization problem has to be solved in the subproblem stage to generate a Pareto-optimal cut. Gleeson and Ryan (1990) have shown that an alternative polyhedron can be used to select cuts corresponding to a minimal infeasible subsystem. This increases the effectiveness of feasibility cuts and thus reduces the number of required cuts. A unified treatment of feasibility cuts and optimality cuts is presented by Fischetti et al. (Fischetti et al., 2010).

Although we have not yet considered mixed-integer cooperative OPF problems, mixed-integer programming was the driving field for numerous contributions regarding cutting plane selection strategies. Balas (1998, 1979) was among the first to study the implications of geometric properties of polyhedra in disjunctive programming. McDaniel and Devine (1977) set the ground for the development of BD as branch-and-cut algorithm for mixed-integer problems. Many more recent articles emphasize the importance of high-density cuts (Sa-

haridis et al., 2010; Sherali and Lunday, 2013; Tang et al., 2013; Azad et al., 2013; Saharidis and Ierapetritou, 2013) to generate more effective cuts and thus reduce computational effort. Another recent development is special cuts that are useful in the root node (Rahmaniani et al., 2020) to the point where no branch-and-bound tree is required because the root node converges to the optimal integer solution. In particular, for feasibility cuts, normalizations of the unbounded dual problem corresponding to an infeasible primal problem occupy a key role (Balas and Perregaard, 2002; Fischetti et al., 2011). In the field of disjunctive programming, Cornuéjols and Lemaréchal (2006) have theoretically found a way to compute facet-defining cuts, an efficient algorithm to do so was recently presented by Conforti and Wolsey (2019). They have developed a method that generates feasibility cuts that almost surely define a facet or an improper face of the true feasible region that is approximated in the master problem (Conforti and Wolsey, 2019). The dissertation of Stursberg (2019) puts this method into the context of the work of Fischetti et al. (2010) and Cornuéjols and Lemaréchal (2006). Conforti and Wolsey's method was also adopted in the CPLEX BD feature (Bonami et al., 2020) and will be central in this article as well.

Applications of BD in contexts similar to ours can also be found in the literature. One example are security-constrained OPF problems (Monticelli et al., 1987; Wang et al., 2016). Another example is expansion location planning while considering the effects of DSM options (Jenabi et al., 2015).

2 Cutting Planes for Cooperative Optimal Power Flow Calculations

In our previous work (Varelmann et al., 2021), we introduced the use of BD to compute a cooperative OPF solution without the need to share detailed process models between plant operators and the grid. This section briefly describes some relevant components of our algorithm; for more details, we refer to our previous article (Varelmann et al., 2021).

We treat the grid as the master problem and cooperating processes as subproblems, whose models should not be shared with (and are unknown to) the grid for confidentiality reasons. Initially, the only constraints for the power demand of each cooperating process in each time interval that are present in the grid are box constraints. We refer to the power demand at all time intervals of a given process as its power profile. The box constraints for the power profiles of cooperating electricity users are a crude outer approximation of the feasible region of power profiles for a cooperating electricity user. In our algorithm, the master problem solves optimal power flow problems, treating the power profiles of cooperating processes as decision variables subject to the current outer approximation of their feasible region.

The power profiles computed this way are then sent to the subproblems as a suggestion for a power profile. The suggestions are likely infeasible, because they only obey constraints representing an outer approximation of the true feasible region of power profiles. Each subproblem that received an infeasible power profile suggestion returns a cut that separates the suggested power profile from the outer approximation of the feasible region of power profiles in the master problem. Thus, each cut refines this outer approximation.

The master problem that represents the electricity grid can be written as (Varelmann

et al., 2021):

$$\min_{\mathbf{p}^{gen},\mathbf{T},\boldsymbol{\theta},\mathbf{p}^{load,\ \text{coop}}} \sum_{i\in\Omega_{Bus}} \sum_{t=1}^{H} F_i(p_{i,t}^{gen}) \tag{1a}$$

$$s.t.$$

$$p_{i,t}^{gen} - \left(p_{i,t}^{load,\ \text{fix}} + p_{i,t}^{load,\ \text{coop}}\right) = \sum_{j\in\Omega_l^i} T_{(i,j),t},\ \ \forall\ i \in \Omega_{Bus}, t \in \{1,\cdots,H\}, \tag{1b}$$

$$T_{(i,j),t} = \beta \cdot S_{i,j}(\theta_{j,t} - \theta_{i,t}),\ \ \forall (i,j) \in \Omega_l, t \in \{1,\cdots,H\}, \tag{1c}$$

$$T_{(i,j)}^{\min} \leq T_{(i,j),t} \leq T_{(i,j)}^{\max},\ \ \forall (i,j) \in \Omega_l, t \in \{1,\cdots,H\}, \tag{1d}$$

$$\theta_i^{\min} \leq \theta_{i,t} \leq \theta_i^{\max},\ \ \forall i \in \Omega_{Bus} \setminus \{\text{slack bus}\}, t \in \{1,\cdots,H\}, \tag{1e}$$

$$\theta_{\text{slack bus},t} = 0\ \ \forall t \in \{1,\cdots,H\}, \tag{1f}$$

$$\mathbf{p}^{gen,\ \min} \leq \mathbf{p}_t^{gen} \leq \mathbf{p}^{gen,\ \max}\ \ \forall t \in \{1,\cdots,H\}, \tag{1g}$$

$$\mathbf{p}_t^{gen} - \mathbf{p}_{t-1}^{gen} \leq \mathbf{p}^{gen,\ \text{ramp up}} \cdot \Delta t\ \ \forall t \in \{1,\cdots,H\}, \tag{1h}$$

$$\mathbf{p}_t^{gen} - \mathbf{p}_{t-1}^{gen} \geq -\mathbf{p}^{gen,\ \text{ramp down}} \cdot \Delta t\ \ \forall t \in \{1,\cdots,H\}, \tag{1i}$$

$$p_{i,t}^{load,\ \text{coop}} \in [p^{lb}, p^{ub}],\ \ \forall i \in \Omega_{Bus},\ \ \forall t \in \{1,\cdots,H\}, \tag{1j}$$

$$\sum_{t=1}^{H} \lambda_t^{c,coup} \cdot p_{i,t}^{load,\ \text{coop}} \geq \boldsymbol{\lambda}^{c\intercal}\mathbf{r}\ \ \forall c \in \{1,\cdots,\#\ \text{Cuts}\}. \tag{1k}$$

where we consider a one day horizon with a 15-minute-interval discretization, so $H = 96$ (Varelmann et al., 2021). We index the set of buses Ω_{Bus} with i. On each bus, the total load comprises fixed load $p_{i,t}^{load,\ \text{fix}}$ and cooperating load $p_{i,t}^{load,\text{coop}}$ (Varelmann et al., 2021). Ω_l^i is the set of buses connected to bus i via a transmission line, Ω_l is the set of transmission lines, and t denotes a time slot of duration Δt (Varelmann et al., 2021). The decision variables are the generation target levels of all generators in the grid, $\mathbf{p}_t^{gen} \in \mathbb{R}^{H\times|\Omega_{Bus}|}$; the power flow between connected pairs of buses i and j, $\mathbf{T} \in \mathbb{R}^{H\times|\Omega_l|}$; and the bus angles with respect to the slack bus in radians, $\boldsymbol{\theta} \in \mathbb{R}^{H\times|\Omega_{Bus}|}$ (Varelmann et al., 2021). The $F_i(\cdot)$ are linear operating cost functions of the generators, whose sum is minimized with the objective (1a) (Varelmann

et al., 2021). Each $p_{i,t}^{load}$ is the power demand on a bus i in time slot t (Varelmann et al., 2021). S is the circuit susceptance matrix with entries $S_{i,j}$ in pu (per unit on a base $\beta = 100$ MW) (Varelmann et al., 2021).

Should the master problem suggest feasible power profiles for all cooperating plants in an iteration, the algorithm immediately terminates, as a feasible solution that is optimal for the grid is found. This reflects our goal to minimize the cost for electricity generation and transmission for the overall grid. Usually, this also reduces electricity costs for electricity users, however, we can not guarantee reduced electricity costs on each bus, i.e., for each electricity user. Also, we do not consider other potentially relevant kinds of operating costs of cooperating processes, such as heating, cooling, or raw material costs in the current format of the problem definition. As a consequence, the subproblems in our algorithm only return feasibility cuts to react to infeasible power profile suggestions. It seems realistic that a cooperation compensation mechanism could be implemented by extending the algorithm to also use optimality cuts, but this has not yet been our focus, and will not be our focus in this article.

In order to compute a cut from the solution of the subproblems in response to an infeasible suggestion, we use a linear dynamic process model of the cooperating process. We use a coupling equation to set the electrical power demand of the process to equal the power profile suggested by the master problem, but allow for slacks in the assignment to the master suggestion. The sum of all slacks is minimized as the objective of the subproblem, as is commonly done in the literature (Birge and Louveaux, 2011). The complete subproblem can

be formulated as (Varelmann et al., 2021):

$$\min_{\mathbf{x},\mathbf{u},\mathbf{p},\mathbf{s}^+,\mathbf{s}^-} \sum_{t=1}^{H} s_t^+ + s_t^-, \tag{2a}$$

$$s.t.$$

$$\mathbf{A}\begin{bmatrix}\mathbf{x}\\ \mathbf{u}\\ \mathbf{p}\end{bmatrix} \geq \mathbf{b}, \tag{2b}$$

$$\mathbf{s}^+, \mathbf{s}^- \geq \mathbf{0}, \tag{2c}$$

$$p_t = p_{i,t}^{load,\ \mathrm{coop}} + s_t^+ - s_t^-, \ \ \forall t \in \{1, \cdots, H\}. \tag{2d}$$

where $\mathbf{s}^+$ and $\mathbf{s}^-$ are the vectors of slack variables s_t^+ and s_t^- in each time step (Varelmann et al., 2021). Similarly, the process states, process inputs, and the power demands p_t of the process in all time intervals are gathered in the vectors $\mathbf{x}$, $\mathbf{u}$, and $\mathbf{p}$ (Varelmann et al., 2021). In (2b), $\mathbf{A}$ and $\mathbf{b}$ form a reformulated representation of the state-space dynamic process model used by Otashu et al. (2021) and Varelmann et al. (2021). The coupling equations are (2d) (Varelmann et al., 2021). If any power profile for which the process operation is feasible (i.e., the subproblem is feasible) exists, then problem (2) is feasible. From the dual information of the solution of (2), a cut can be generated that cuts off the suggestion defined by the $p_{i,t}^{load,\ \mathrm{coop}}$ in the master problem. As this formulation minimizes the distance between a feasible power profile and the suggestion from the master in the L1-norm, we will refer to it as the *L1-normalized subproblem*, or short *L1 subproblem*. Also, we refer to the cuts obtained from such subproblems as *L1 cuts* and we refer to the feasible power profiles $\mathbf{p}$, which is the sum of master suggestion and slacks, as *L1 feasible points*.

Finally, our algorithm relies on a heuristic termination criterion. As the master problem has an outer approximation of the feasible region of power profiles for cooperating processes, every solution of a master problem defines a lower bound on the optimal objective of the cooperative OPF problem. Since we only refine the outer approximation of the feasible region of power profiles for cooperating processes by adding cuts, i.e., restricting the feasible

region of the overall master problem, this lower bound is monotonically increasing. On the other hand, each subproblem solution defines a feasible power profile for the cooperating process represented by the subproblem. This feasible power profile can be used to solve a non-cooperative OPF problem, where the demand of the process is fixed to the feasible power profile. The objective of such a non-cooperative OPF problem is an upper bound to the optimal objective of the cooperative OPF problem (Varelmann et al., 2021). Thus, we could compute an optimality gap in each iteration and terminate the algorithm once a user-defined optimality gap is reached. However, computing the upper bound is a non-negligible computational effort, as it involves the solution of a non-cooperative OPF problem.

Therefore, our heuristic is to only compute the upper bound when the subproblem solutions have small objectives, which signals potential for an acceptable optimality gap. We consider the subproblem solutions promising if the "feasibility-distance" specified by the sum of the objectives of all subproblems is below a threshold that is changed adaptively over the course of the iterations (Varelmann et al., 2021). Without this termination criterion, the algorithm would have to converge the BD up to solver tolerance. This requires orders of magnitude more iterations, is prone to numerical issues, and provides no significant further cost improvements, as we described in our previous work (Varelmann et al., 2021). With this criterion, we terminate the algorithm with a rigorously proven optimality gap, which the user of the algorithm can select. In our previous work (Varelmann et al., 2021), we recommended an optimality gap of 0.01%.

3 An Alternative Normalization of Infeasible Subproblems to Generate Advanced Cuts

While the L1-normalization is common practice to compute feasibility cuts, it only guarantees that the suggested solution from the master problem is cut off in the next iteration. There are no farther-reaching guarantees for the overall quality of feasibility cuts obtained this

way. As an alternative, Conforti and Wolsey (2019) have developed a method to efficiently compute feasibility cuts that come with stronger quality properties from linear cut-generating problems. An important property of linear programs (LPs) is that their feasible region is a convex polytope, i.e., an n-dimensional body bounded by $n-1$-dimensional facets. The method of Conforti and Wolsey (2019) computes cuts that almost surely define facets or improper faces of the feasible region of the cut-generating linear problem with just one LP. The basic idea is to specify a ray going from the master suggestion to a core point of the feasible region of the cut-generating problem and finding the point where the ray crosses the boundary of the feasible region. In the context of cooperative OPF problems, we adapt their method to compute strong cuts to describe the feasible region of power profiles of cooperating processes. As a natural core point, we use the power profile of nominal process operation, where the power demand is constant over the entire time horizon.

The formulation of our subproblems inspired by the method of Conforti and Wolsey is:

$$\min_{\mathbf{x},\mathbf{u},\mathbf{p},\lambda} \lambda, \tag{3a}$$

$$s.t.$$

$$\mathbf{A}\begin{bmatrix}\mathbf{x}\\ \mathbf{u}\\ \mathbf{p}\end{bmatrix} \geq \mathbf{b}, \tag{3b}$$

$$0 \leq \lambda \leq 1, \tag{3c}$$

$$p_t = (1-\lambda)\cdot p_{i,t}^{load,\ \text{coop}} + \lambda \cdot p^0, \quad \forall t \in \{1,\cdots,H\}. \tag{3d}$$

where λ describes how far the master suggestion has to be moved towards the nominal power demand p^0. To distinguish this subproblem, the cuts and feasible power profiles generated from it from their counterpart generated with formulation (2), we refer to this formulation as *CW-normalized subproblem* or *CW subproblem*, we refer to the cuts generated form such subproblems as *CW cuts*, and we refer to feasible power profiles $\mathbf{p}$ as *CW feasible points*. A CW feasible point is the convex combination of master suggestion and core point that is

feasible and closest to the master suggestion.

4 Case Studies

4.1 Cooperative OPF with Concentrated and Distributed Cooperating Load

We compare the performance of our decoupled cooperative OPF algorithm with L1 subproblems and CW subproblems with the case studies known from our previous works (Otashu et al., 2021; Varelmann et al., 2021). As grid model, we use the IEEE reliability test system 24-bus network model (Soroudi, 2017). We consider two grid states: in a normal grid, the full generation capacity is available, while in a congested grid, we removed the largest generator and drastically reduced the generation capacity of the second-largest generator, which corresponds to a total loss of generation capacity of 24%. We examine the cutting strategies with 71 MW concentrated on bus 5 and distributed along buses 5, 7, and 8, both in a normal and in a congested grid. In Figure 1, we visualize the topology of the grid; the locations of buses 5, 7, and 8; and the locations of the generators failing in the congested grid.

As cooperating process, we choose a chlor-alkali process, which is energy intensive but has temporal flexibility in its production. Therefore, it is well-suited for DSM. In both (2b) and (3b), we use a linearized version of the rigorous model presented by Otashu and Baldea (Otashu and Baldea, 2019), which represents the process behavior sufficiently well for the purposes of cooperative OPF (Otashu et al., 2021).

4.2 Examination of the Scaling Behavior with Multiple Cooperating Loads

Later, we also study the scaling behavior with the number of buses with cooperating load of a tailored cutting strategy in the IEEE 24-bus reliability test grid. In setup A, we place

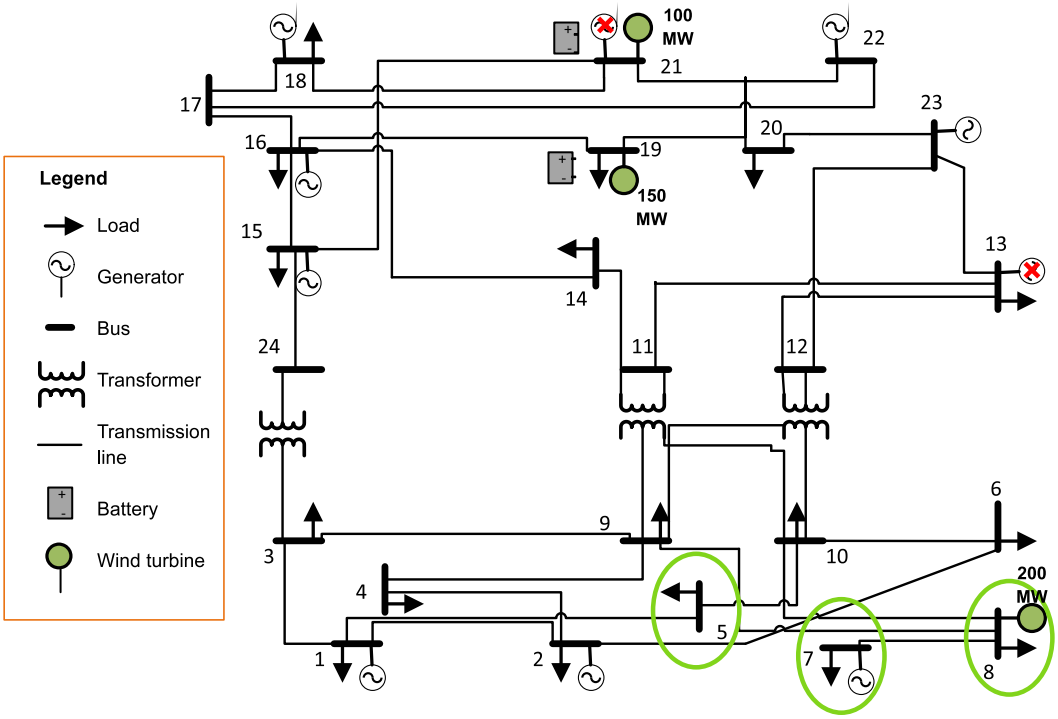

Figure 1: Modified IEEE 24 bus reliability test grid (Soroudi, 2017). We place concentrated cooperating load of 71 MW on bus 5, and distributed the 71 MW of cooperating load along buses 5, 7, and 8. Generators that are removed or reduced in a congested grid are marked with a red cross. The total generation capacity reduction in a congested grid is 24% of the normal generation capacity.

cooperative loads on buses 5, 10, 9, and 19; in setup B we place cooperative loads on buses 7, 6, 10, and 16. These are the same scaling study setups we presented in our previous article (Varelmann et al., 2021). Like we did there, we solve the cooperative OPF problem with our decoupled algorithm with cooperating load only on the first bus of each setup, followed by the first two buses, the first three buses, and all four buses. Each cooperating load corresponds to half of the total load on the respective bus, as we did previously as well (Varelmann et al., 2021). We study the scaling setups A and B in a normal and a congested grid.

4.3 Computational Infrastructure

Our source code is written in C++ and uses Gurobi 9.5.0 as the LP-solver on a Linux Machine with a AMD Ryzen Threadripper 2990WX 32-Core CPU with 2.20 GHz for each core and 64 GB RAM. Our code and the obtained result data is available open-source under the EPL 2.0 license on http://permalink.avt.rwth-aachen.de/?id=597385. To mimic a distributed solution process as well as possible, we solve all subproblems in parallel threads using modern C++ concurrency features.

5 The Benefits of CW-Normalization

5.1 Algorithmic Performance using L1-Normalization, CW-Normalization and Hybrids

For our first analysis, we compare the performance of the following cutting strategies: only using L1-normalized subproblems (which recomputes the results from our previous article (Varelmann et al., 2021)), only using CW-normalized subproblems, using cuts from both subproblems and feasible points from L1-subproblems to compute the optimality gap, and using cuts from both subproblems and feasible points from CW-subproblems to compute the

Table 1: Iterations Until Acceptable Relative Optimality Gap with Different Cutting Strategies

Case Study Setup	only L1 subproblems	only CW subproblems	L1 and CW cuts, L1 feasible points	L1 and CW cuts, CW feasible points
5n	143	54	41	70
5c	166	67	51	74
578n	286	79	74	108
578c	492	149	96	152

optimality gap. We examine the performance of these strategies in terms of iterations and runtime until an acceptable relative optimality gap of 0.01% can be guaranteed with the concentrated cooperating load on bus 5 and the distributed cooperating load along buses 5, 7 and 8 for a grid in normal and congested state. As a short notation, we refer to the concentrated setups as 5n and 5c, and to the distributed setups as 578n and 578c. The iterations until termination with an acceptable relative optimality gap are given in Table 1, the runtime until termination is visualized in Figure 2.

An analysis of Table 1 and Figure 2 allows several important conclusions: First, and most striking is that CW cuts are substantially more effective than L1 cuts. Second, L1 subproblems also have their merit, as a comparison of the two strategies that use both cuts reveals: The strategy using L1 feasible points to determine the current relative optimality gap terminates perceptibly faster than the strategy using CW feasible points for this purpose, although both methods use the same cuts. This is not surprising, because L1 feasible points can be located in any direction from the master suggestion and will minimize the L1-distance to the master suggestion, whereas CW feasible points have to lie on a specific ray from the master suggestion towards the core point (which corresponds to nominal production here).

Third, a rather unexpected finding is that the strategy only using CW subproblems outperforms the strategy using both kinds of cuts and CW feasible points. These two strategies only differ in the absence or utilization of L1 cuts. In terms of both iterations and runtime, the strategy only using CW subproblems consistently outperforms the strategy that additionally uses L1 cuts. Our interpretation of the evidence that using L1 cuts leads to larger computational effort is that L1 cuts can sometimes cut off regions from the master

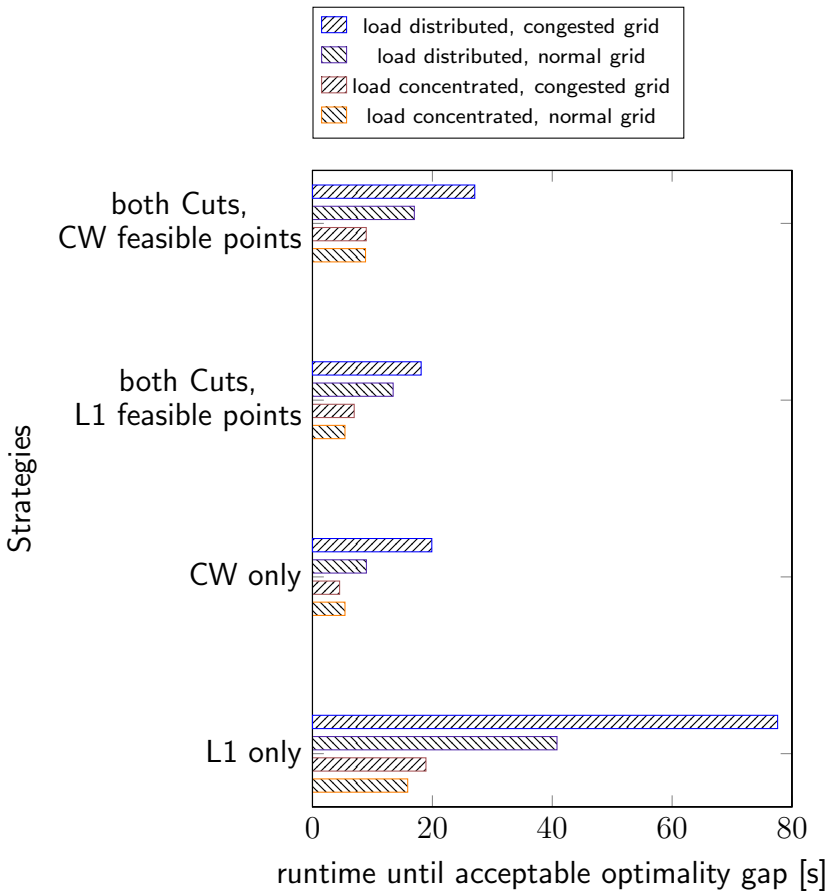

Figure 2: Runtimes of our four case studies using different subproblem formulations. The fastet strategy is 'CW only', followed by 'both Cuts, L1 feasible points'. The 'L1 only' strategy used in (Varelmann et al., 2021) is clearly outperformed by all other strategies.

problem that – when suggested to a CW subproblem – would yield particularly useful CW cuts. We conclude that – in the presence of CW cuts – L1 cuts have only minimal utility and can in fact be harmful for the performance. Broadly translated, our findings indicate cuts should exclusively be generated from CW subproblems, while L1 feasible points have a smaller optimality gap than CW feasible points, so the former are preferred over the latter.

5.2 Tailored Single- and Multi-CW-Cut Strategies

Motivated by these findings, we propose two more strategies that are tailored to combine the best properties from both kinds of subproblems: First, a single-cut strategy that only uses cuts from CW subproblems, but solves an L1 subproblem to use an L1 feasible point

in the non-cooperative OPF problem when the termination heuristic needs to compute an upper bound. Second, we extend this strategy by a second CW cut in each iteration to get a multi-cut strategy. To generate a second cut that can differ from the first in each iteration, we solve the CW subproblem with a different core point. The only requirement on a core point for the resulting CW-cut to be valid is that the core point must be a feasible power profile. In particular, every convex combination between nominal production and the feasible points obtained from a subproblem solution is a valid core point, because the feasible region of power profiles is a convex region.

As we visualize in Figure 3, to generate a cut that differs from the first cut, we can not use a convex combination of the feasible point from the first subproblem and the standard core point. The reason is that the ray defined by the master suggestion and such a convex combination would by construction be the same ray as in the first subproblem. Therefore, we store the feasible point obtained from the first subproblem of the *previous iteration* and select a convex combination between this feasible point and the point corresponding to nominal production:

$$\mathbf{p}^0_{i,(sec)} = \mu \cdot \mathbf{p}_{i-1} + (1-\mu) \cdot \mathbf{p}^0 \tag{4}$$

where $\mathbf{p}^0_{i,(sec)}$ is the core point of the second CW subproblem in iteration i, $\mathbf{p}_{i-1}$ is the feasible point from the first CW subproblem in the previous iteration, and $\mathbf{p}^0$ is the standard core point corresponding to nominal production.

The convex combination factor μ can in principle be chosen freely between 0 and 1. For $\mu \geq 0.995$, we sometimes faced numerical infeasibilities of the resulting second CW subproblem. Again, Figure 3 can explain why this happens: With values for μ close to 1, we might aim at a tip of the true feasible set. Then, the part of the ray defined by master suggestion and core point that corresponds to feasible power profiles vanishes numerically. For $\mu \leq 0.995$, we did not have any such issues and the performance of the tailored multi-cut strategy was largely insensitive to the value of μ, with best results for $\mu \in [0.95, 0.99]$. We therefore choose $\mu = 0.97$ for this case study. Note that the second cut might yield a new

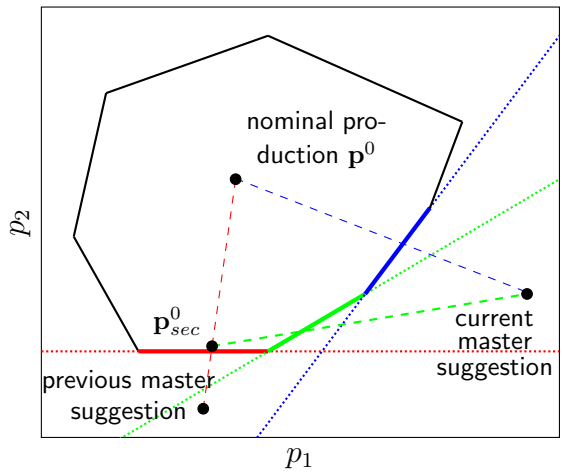

Figure 3: Geometry of core point selection of second CW subproblem of an iteration. We construct a second core point $\mathbf{p}^0_{sec}$ by a linear combination of $\mathbf{p}^0$ and the feasible point found in the previous iteration on the solid red line. The secondary subproblem aims to find a facet between the current master suggestion and $\mathbf{p}^0_{sec}$. In this case, the solid green facet is identified without the need for an additional master iteration. However, the secondary cut might in general also be a duplication of the primary cut in blue.

cut, but is not guaranteed to do so in every iteration.

In Table 2 and Figure 4, we present the required number of iterations and the runtime from the two new strategies until an acceptable optimality gap is obtained. For an easier comparison, we also repeat the results from the two best strategies of the previous section, namely 'CW only' and 'both cuts, L1 feasible points' therein. We can see that our tailored single-cut strategy takes comparably many iterations as the strategy using both cuts and L1 feasible points. The strategy only using CW subproblems requires more iterations than any other strategy shown in Table 2, as it is the only strategy using CW feasible points to compute the optimality gap. The tailored multi-cut strategy reduced the number of iterations required by around a third compared to the tailored single-cut strategy. This is a good result and clearly the lowest number of required iterations any strategy has given. Nevertheless, it also shows that the entirety of second cuts is less effective than the first cuts, because for equally effective second cuts, we would have expected the number of iterations to decrease by around 50 % rather than only around 33 %. However, we did expect that sometimes, second cuts just duplicate the first cut, so we were not surprised by this finding.

Table 2: Iterations Until Acceptable Relative Optimality Gap with Tailored Cutting Strategies

Case Study Setup	only CW subproblems	L1 and CW cuts, L1 feasible points	CW cuts, L1 feasible points	2 CW cuts/iteration, L1 feasible points
5n	54	41	52	36
5c	67	51	50	35
578n	79	74	65	48
578c	149	96	105	63

The lowest runtimes are observed using our tailored strategies in all cases; sometimes the 'CW only' strategy is competitive. The tailored multi-cut strategy requires fewer iterations than the tailored single-cut strategy. The former solves twice as many subproblems per iteration. This additional computational effort only has a minor effect on the runtime of each iteration, because we solve all subproblems in parallel. The only drawback of adding two cuts per iteration is that the simplex basis we use to warm-start the master problem has to be updated with respect to two cuts. Therefore, the multi-cut strategy often, but not always, terminates earlier than the single-cut strategy in terms of runtime. We conclude that the tailored multi-cut strategy works best in our case studies, but the single-cut strategy and the CW-only strategy sometimes find an acceptable solution just as quickly.

5.3 Cooperative OPF Solutions of Tailored Strategies

The cooperative OPF solutions computed by our algorithm using the tailored cutting strategies are similar to the solutions we found using L1 subproblems in our previous work (Varelmann et al., 2021). Using these three strategies, we compare the generation and transmission costs for the grid in Table 3. It shows that the generation and transmission cost for the grid is independent of the chosen strategy up to some tens of dollars. The same is true for the electricity cost of the cooperating plant, which we report for the setups with concentrated load in Table 4. We do not show the electricity cost for cooperating plants in the setups with distributed loads, as they are independent of the chosen strategy as well.

As an example, we also visualize the final power profiles of the cooperating plant with the resulting price profile for the setup with a concentrated cooperating load on bus 5 in a

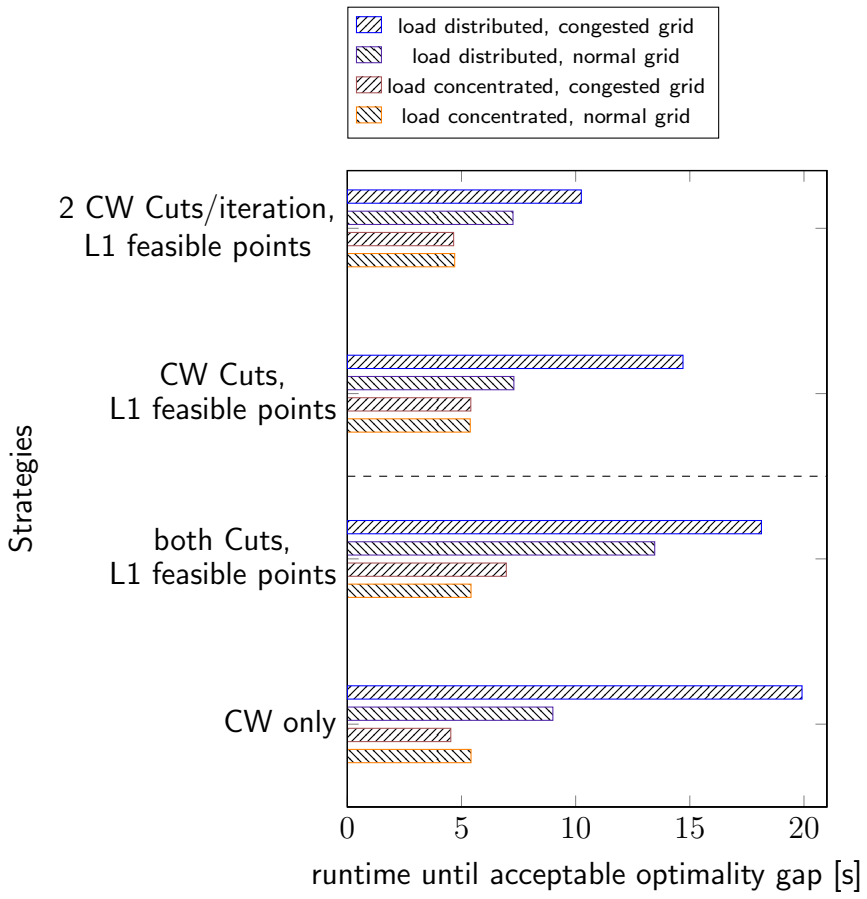

Figure 4: Runtime comparison of the two tailored CW strategies (at the top) to two promising strategies from before (at the bottom). We obtain the best performance with the tailored multi-cut strategy. In some isolated setups, the performance of the tailored single-cut strategy or the 'CW only' strategy are comparable.

normal grid. In Figure 5, we show that the choice of L1 subproblems, or our tailored single-, or multi-cut strategies has (as it should) only minimal influence on the power profiles and price profiles. In other words, the choice of cutting strategy has no impact on the economic or dispatch results – neither cost for grid or plants, nor power power profiles or price profiles –, it only affects how efficiently the results are computed.

Table 3: Cooperative Optimal Power Flow Costs With Different Strategies

Setup	concentrated load, normal grid	concentrated load, congested grid	distributed load, normal grid	distributed load, congested grid
optimal L1 objective [$]	1,804,476	1,816,380	1,804,489	1,818,665
optimal single-cut objective [$]	1,804,472	1,816,325	1,804,468	1,818,627
optimal multi-cut objective [$]	1,804,435	1,816,351	1,804,460	1,818,629

Table 4: Electricity Costs for Cooperating Plants With Different Strategies

Setup	concentrated load, normal grid	concentrated load, congested grid
L1 plant cost [$]	16,035	25,453
single-cut plant cost [$]	16,031	25,420
multi-cut plant cost [$]	16,054	25,404

5.4 Scaling Behavior of Decoupled Cooperative OPF Algorithm with CW-Normalization

We repeat the analysis of the scaling behavior of our decoupled algorithm for cooperative OPF problems with respect to multiple buses with cooperating loads from our previous article (Varelmann et al., 2021) as described in Section 4. We use our tailored multi-cut strategy as it performed best in the previous section. In Figure 6, we show the absolute runtime until an acceptable solution is found, and also include the speedup with respect to the strategy using only L1-normalized subproblems that we used previously (Varelmann et al., 2021).

The new cutting strategy leads to a major improvement in the scaling behavior of our algorithm. In our previous article (Varelmann et al., 2021), the trend was a rapidly increasing runtime. Now, we can still see an increase in the runtime with additional buses with cooperating loads, but this increase reflects the overall larger problem much better than it used to. We can solve all problems with four buses with cooperating loads in less than twenty seconds. Even more important for the scaling behavior than an overall reduction in runtime is that the speedup relative to an L1 subproblem strategy is increasing as more buses have cooperating load. This clearly shows that we have improved the scaling behavior and not only reduced the overall runtime. For four buses with cooperating load, we achieve

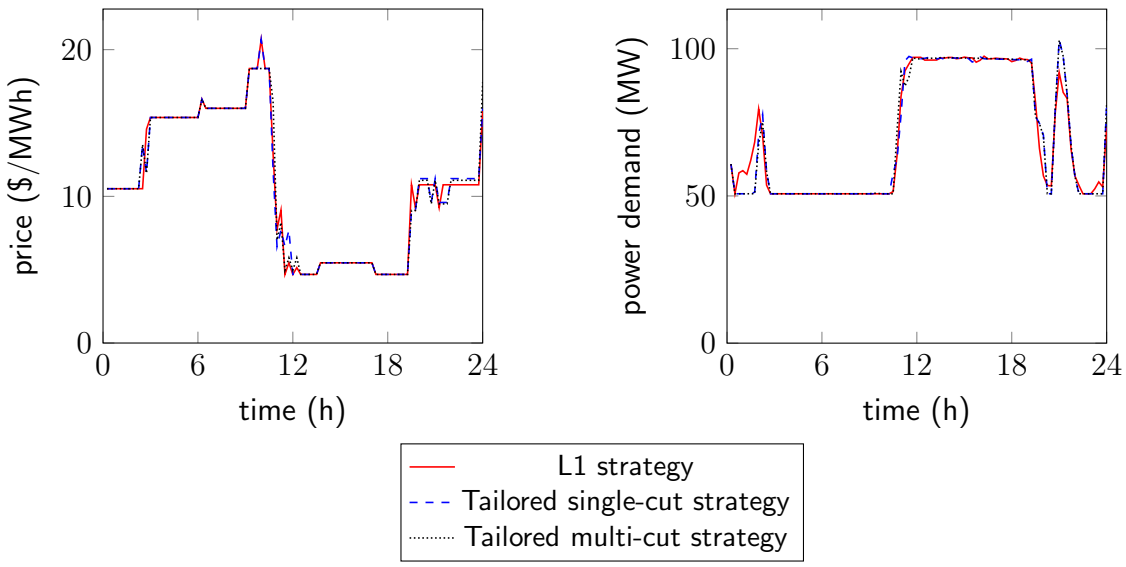

Figure 5: Comparison of price profiles and power profiles computed with different cutting strategies. As expected, both tailored strategies reproduce the results from the L1 strategy closely.

a speedup of almost a factor of 60. Thus, computing a cooperative OPF solution becomes practically tractable for a larger number of cooperating load locations.

Encouraged by these results, we also tried to solve cooperative OPF problems with even more buses with cooperating load. We solved problems where 6, 10, and all 17 load-buses that are available in the IEEE 24-bus reliability test grid had cooperating loads corresponding to half of the total load on that bus. For the setup with 6 buses with cooperating load, we used buses 1, 2, 6, 7, 10, and 16; and for the setup with 10 buses with cooperating load, we used buses 1, 2, 6, 7, 9, 10, 13, 16, 18, and 20. We ran all these problems in a normal and a congested grid. The number of required iterations, and the runtime until an acceptable solution with a relative optimality gap of 0.01 % was found are listed in Table 5. The results increase our confidence that we have developed a practical method.

While studying problems with many buses with cooperating loads, we made another interesting observation: The integrated cooperative OPF problem formulation that used confidential dynamic process models (Otashu et al., 2021) is numerically difficult to solve as soon as 4 or more buses with cooperating load exist. Even with Gurobi features such as the

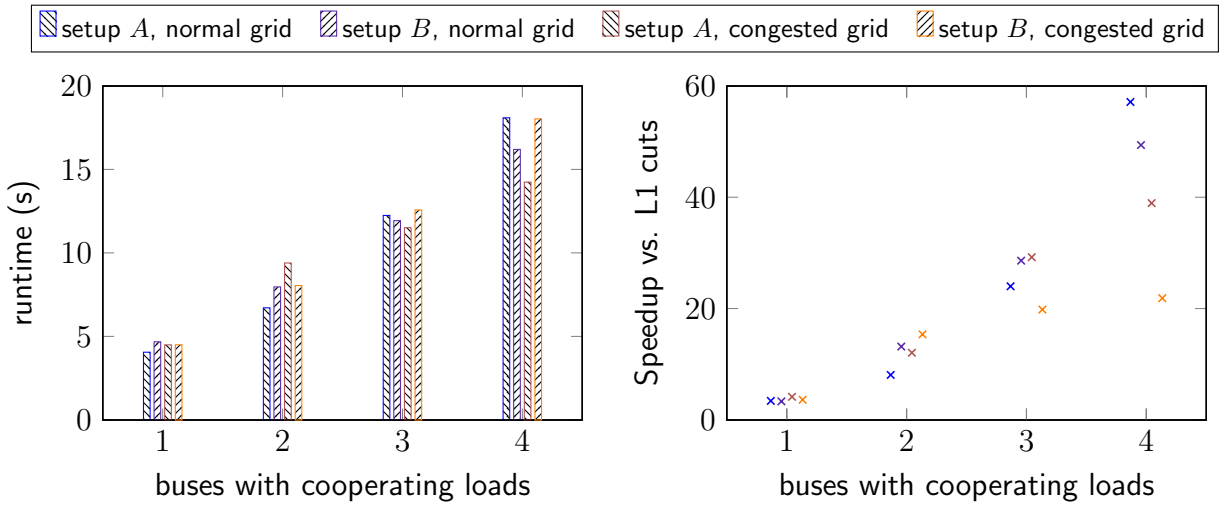

Figure 6: Improved scaling behavior of our tailored multi-cut strategy with respect to co-operating end-users on multiple buses. The left plot visualizes the total runtime with our tailored multi-cut strategy and the right plot shows the speedup of this strategy compared to only using L1 cuts as we did in our previous work (Varelmann et al., 2021).

'NumericalFocus' parameter set to the maximum value, we could not solve the integrated problem for some setups with 4 buses with cooperating load and all setups with 6 buses with cooperating load, we did not try for 10 or 17 buses. In contrast, our decomposition-based algorithm shows numerical issues for the first time when using 17 buses with cooperating load in a congested grid. They could be circumvented by setting the 'NumericalFocus' parameter to the maximum value. The range of matrix coefficients in master and subproblems is several orders of magnitude smaller than the range of coefficients in the integrated problem after

Table 5: Performance of the multi-cut strategy for a large number of buses with cooperating load

Setup	Iterations	Runtime [s]
6 buses, normal grid	137	35
6 buses, normal grid	91	24
10 buses, normal grid	127	99
10 buses, normal grid	295	458
17 (all) buses, normal grid	230	758
17 (all) buses, normal grid	231	2879*

∗ With cooperating load on all buses, we had to set Gurobi's 'NumericalFocus' option to 3 because of numerical issues. This lets Gurobi use quadruple precision and sets other settings to particularly careful values – at the expense of maximum performance.

decomposing the problem to grid and cooperating processes.

6 Conclusions

We have applied the method to generate facet-defining cuts presented by Conforti and Wolsey (2019) to our BD-based algorithm to compute cooperative OPF solutions and compared several combinations of subproblem usages in our algorithm. The comparison results motivated the development of tailored strategies using CW-normalized subproblems to generate one or more cuts per iteration and L1-normalized subproblems to find feasible points that provide a good upper bound. In our case studies, we could confirm that our tailored strategies perform better than all the strategies contained in our initial comparison. Finally, we examined the scaling behavior of our tailored multi-cut strategy with respect to a growing number of buses with cooperating load in the grid. We showed that with the multi-cut strategy we developed, the computational effort only increases moderately with the number of buses with cooperating load. In combination with the overall better performance, our decoupled algorithm to solve cooperative OPF problems exhibits a performance that enables practical usage. A remaining challenge for the concept of cooperative optimal power flow computations remains that it is unclear how to guarantee a fair compensation for all cooperating participants – either as reduced operating cost or monetary compensation.

Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 333849990/ GRK2379 (IRTG Modern Inverse Problems). Partial financial support for MB from the National Science Foundation (NSF) through the CAREER Award 1454433 is acknowledged with gratitude.

References

N. Azad, G. K. Saharidis, H. Davoudpour, H. Malekly, and S. A. Yektamaram. Strategies for protecting supply chain networks against facility and transportation disruptions: an improved Benders decomposition approach. *Annals of Operations Research*, 210(1):125–163, 2013. doi: 10.1007/s10479-012-1146-x.

E. Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–51, 1979.

E. Balas. Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Applied Mathematics*, 89(1-3):3–44, 1998. doi: 10.1016/s0166-218x(98)00136-x.

E. Balas and M. Perregaard. Lift-and-project for mixed 0–1 programming: recent progress. *Discrete Applied Mathematics*, 123(1-3):129–154, 2002. doi: 10.1016/s0166-218x(01)00340-7.

M. Baldea. *Employing Chemical Processes as Grid-Level Energy Storage Devices*, pages 247–271. Springer International Publishing, Cham, 2017. doi: 10.1007/978-3-319-42803-1_9.

J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962. doi: 10.1007/bf01386316.

J. R. Birge and F. Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, New York, 2011. doi: 10.1007/978-1-4614-0237-4.

P. Bonami, D. Salvagnin, and A. Tramontani. Implementing Automatic Benders Decomposition in a Modern MIP Solver. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 78–90. Springer, 2020. doi: 10.1007/978-3-030-45771-6_7.

A. Caspari, C. Offermanns, P. Schäfer, A. Mhamdi, and A. Mitsos. A flexible air separation process: 2. Optimal operation using economic model predictive control. *AIChE Journal*, 65(11):e16721, 2019. doi: 10.1002/aic.16721.

P. M. Castro, L. Sun, and I. Harjunkoski. Resource–task network formulations for industrial demand side management of a steel plant. *Industrial & Engineering Chemistry Research*, 52(36):13046–13058, 2013. doi: 10.1021/ie401044q.

P. M. Castro, G. Dalle Ave, S. Engell, I. E. Grossmann, and I. Harjunkoski. Industrial demand side management of a steel plant considering alternative power modes and electrode replacement. *Industrial & Engineering Chemistry Research*, 59(30):13642–13656, 2020. doi: 10.1021/acs.iecr.0c01714.

M. Conforti and L. A. Wolsey. "Facet" separation with one linear program. *Mathematical Programming*, 178(1):361–380, 2019. doi: 10.1007/s10107-018-1299-8.

G. Cornuéjols and C. Lemaréchal. A convex-analysis perspective on disjunctive cuts. *Mathematical Programming*, 106(3):567–586, 2006. doi: 10.1007/s10107-005-0670-8.

M. Fischetti, D. Salvagnin, and A. Zanette. A note on the selection of Benders' cuts. *Mathematical Programming*, 124(1-2):175–182, 2010. doi: 10.1007/s10107-010-0365-7.

M. Fischetti, A. Lodi, and A. Tramontani. On the separation of disjunctive cuts. *Mathematical Programming*, 128(1):205–230, 2011. doi: 10.1007/s10107-009-0300-y.

J. Gleeson and J. Ryan. Identifying minimally infeasible subsystems of inequalities. *ORSA Journal on Computing*, 2(1):61–63, 1990. doi: 10.1287/ijoc.2.1.61.

H. Hadera, I. Harjunkoski, G. Sand, I. E. Grossmann, and S. Engell. Optimization of steel production scheduling with complex time-sensitive electricity cost. *Computers & Chemical Engineering*, 76:117–136, 2015. doi: 10.1016/j.compchemeng.2015.02.004.

C. Hoffmann, J. Weigert, E. Esche, and J.-U. Repke. Towards demand-side management of the chlor-alkali electrolysis: Dynamic, pressure-driven modeling and model validation of the 1, 2-dichloroethane synthesis. *Chemical Engineering Science*, 214:115358, 2020. doi: 10.1016/j.ces.2019.115358.

M. Ierapetritou, D. Wu, J. Vin, P. Sweeney, and M. Chigirinskiy. Cost minimization in an energy-intensive plant using mathematical programming approaches. *Industrial & Engineering Chemistry Research*, 41(21):5262–5277, 2002. doi: 10.1021/ie011012b.

M. Jenabi, S. F. Ghomi, S. A. Torabi, and S. H. Hosseinian. Acceleration strategies of Benders decomposition for the security constraints power system expansion planning. *Annals of Operations Research*, 235(1):337–369, 2015. doi: 10.1007/s10479-015-1983-5.

Y. Li, B. L. Ng, M. Trayer, and L. Liu. Automated residential demand response: Algorithmic implications of pricing models. *IEEE Transactions on Smart Grid*, 3(4):1712–1721, 2012. doi: 10.1109/tsg.2012.2218262.

T. L. Magnanti and R. T. Wong. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484, 1981. doi: 10.1287/opre.29.3.464.

P. McAuliffe and A. Rosenfeld. Response of residential customers to critical peak pricing and time-of-use rates during the summer of 2003. *California Energy Commission*, pages 1–14, 2004.

D. McDaniel and M. Devine. A modified Benders' partitioning algorithm for mixed integer programming. *Management Science*, 24(3):312–319, 1977. doi: 10.1287/mnsc.24.3.312.

A. Monticelli, M. Pereira, and S. Granville. Security-constrained optimal power flow with post-contingency corrective rescheduling. *IEEE Transactions on Power Systems*, 2(1): 175–180, 1987. doi: 10.1109/mper.1987.5527553.

J. I. Otashu and M. Baldea. Demand response-oriented dynamic modeling and operational optimization of membrane-based chlor-alkali plants. *Computers & Chemical Engineering*, 121:396–408, 2019. doi: 10.1016/j.compchemeng.2018.08.030.

J. I. Otashu, K. Seo, and M. Baldea. Cooperative optimal power flow with flexible chemical process loads. *AIChE Journal*, 67(4):e17159, 2021. doi: 10.1002/aic.17159.

P. Palensky and D. Dietrich. Demand side management: Demand response, intelligent energy systems, and smart loads. *IEEE Transactions on Industrial Informatics*, 7(3): 381–388, Aug 2011. ISSN 1551-3203. doi: 10.1109/tii.2011.2158841.

N. Papadakos. Practical enhancements to the Magnanti–Wong method. *Operations Research Letters*, 36(4):444–449, 2008. doi: 10.1016/j.orl.2008.01.005.

R. Rahmaniani, S. Ahmed, T. G. Crainic, M. Gendreau, and W. Rei. The Benders dual decomposition method. *Operations Research*, 68(3):878–895, 2020. doi: 10.1287/opre.2019.1892.

K. Roh, L. C. Brée, K. Perrey, A. Bulan, and A. Mitsos. Optimal oversizing and operation of the switchable chlor-alkali electrolyzer for demand side management. In *Computer Aided Chemical Engineering*, volume 46, pages 1771–1776. Elsevier, 2019. doi: 10.1016/b978-0-12-818634-3.50296-4.

G. K. Saharidis and M. G. Ierapetritou. Speed-up Benders decomposition using maximum density cut (MDC) generation. *Annals of Operations Research*, 210(1):101–123, 2013. doi: 10.1007/s10479-012-1237-8.

G. K. Saharidis, M. Minoux, and M. G. Ierapetritou. Accelerating Benders method using covering cut bundle generation. *International Transactions in Operational Research*, 17 (2):221–237, 2010. doi: 10.1111/j.1475-3995.2009.00706.x.

K. Seo, T. F. Edgar, and M. Baldea. Optimal demand response operation of electric boosting glass furnaces. *Applied Energy*, 269:115077, 2020.

H. D. Sherali and B. J. Lunday. On generating maximal nondominated Benders cuts. *Annals of Operations Research*, 210(1):57–72, 2013. doi: 10.1007/s10479-011-0883-6.

A. Soroudi. *Power system optimization modeling in GAMS*, volume 78. Springer, Cham, 2017. doi: 10.1007/978-3-319-62350-4_2.

J. P. Spangenberg, M. Kleingeld, and J. F. van Rensburg. Analysing the effect of DSM projects at South African cement factories. In *2015 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, pages 117–122. IEEE, 2015. doi: 10.1109/icue.2015.7280256.

P. M. Stursberg. *On the mathematics of energy system optimization*. PhD thesis, Technische Universität München, 2019.

D. L. Summerbell, D. Khripko, C. Barlow, and J. Hesselbach. Cost and carbon reductions from industrial demand-side management: Study of potential savings at a cement plant. *Applied Energy*, 197:100–113, 2017. doi: 10.1016/j.apenergy.2017.03.083.

L. Tang, W. Jiang, and G. K. Saharidis. An improved Benders decomposition algorithm for the logistics facility location problem with capacity expansions. *Annals of Operations Research*, 210(1):165–190, 2013. doi: 10.1007/s10479-011-1050-9.

D. Todd, M. Caufield, B. Helms, A. P. Generating, I. M. Starke, B. Kirby, and J. Kueck. Providing reliability services through demand response: A preliminary evaluation of the demand response capabilities of Alcoa Inc. *ORNL/TM*, 233, 2008.

C. Tsay, A. Kumar, J. Flores-Cerrillo, and M. Baldea. Optimal demand response scheduling of an industrial air separation unit using data-driven dynamic models. *Computers & Chemical Engineering*, 126:22–34, 2019. doi: 10.1016/j.compchemeng.2019.03.022.

T. Varelmann, J. I. Otashu, K. Seo, A. W. Lipow, A. Mitsos, and M. Baldea. A decoupling strategy for protecting sensitive information in cooperative optimization of power flow. *AIChE Journal*, 2021. doi: 10.1002/aic.17429. in press, 19th Aug 2021.

Q. Wang, J. D. McCalley, T. Zheng, and E. Litvinov. Solving corrective risk-based security-constrained optimal power flow with Lagrangian relaxation and Benders decomposition. *International Journal of Electrical Power & Energy Systems*, 75:255–264, 2016. doi: 10.1016/j.ijepes.2015.09.001.

Q. Zhang and I. E. Grossmann. Planning and scheduling for industrial demand side management: advances and challenges. In *Alternative Energy Sources and Technologies*, pages 383–414. Springer, 2016. doi: 10.1007/978-3-319-28752-2_14.

Q. Zhang, I. E. Grossmann, C. F. Heuberger, A. Sundaramoorthy, and J. M. Pinto. Air separation with cryogenic energy storage: optimal scheduling considering electric energy and reserve markets. *AIChE Journal*, 61(5):1547–1558, 2015. doi: 10.1002/aic.14730.

X. Zhang and G. Hug. Bidding strategy in energy and spinning reserve markets for aluminum smelters' demand response. In *2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5. IEEE, 2015. doi: 10.1109/ISGT.2015.7131854.