Fast Subproblem Solving in Benders Decomposition for Affine Potential-Based Flow Problems with Topology Switching and Scenarios

Oliver Gaul*
Chair of Operations Research
RWTH Aachen University
gaul@or.rwth-aachen.de
ORCID: 0000-0002-2131-1911

Tim Donkiewicz
Chair of Operations Research
RWTH Aachen University
donkiewicz@or.rwth-aachen.de
ORCID: 0000-0002-5721-3563

Marco Lübbecke
Chair of Operations Research
RWTH Aachen University
luebbecke@or.rwth-aachen.de
ORCID: 0000-0002-2635-0522

November 11, 2026

Abstract

Potential-based flow problems arise for a range of practical problems, like the operation of lossless power networks. The distribution of flows through the network is governed by the potential equation, which relates arc flows and node potentials following arc potential functions. We consider a general variant with affine potential functions, topology switching, and scenarios. It can be solved using a Benders decomposition algorithm, with each scenario corresponding to a subproblem. Subproblem solving time can then significantly contribute to overall running time. We present a fast algorithm to obtain Benders optimality and feasibility cuts from fixed master solutions. Problems with affine potential functions are reduced to equivalent problems where every potential function is the identity function. Optimal primal and dual solutions as well as unbounded dual rays are determined by systems of linear equations (SLE), resulting in a constant number of SLEs solved per subproblem. SLEs are solved using LU-factorizations, with the similarity between primal and dual, scenarios, and master solutions exploited by reusing factorizations via low-rank updates to significantly reduce theoretical complexity. We demonstrate the method on security-constrained optimal transmission switching instances, where subproblem solve time reductions of 97.98 percent compared to a standard LP solver are observed.

Keywords: Benders decomposition, potential-based flows, security-constrained optimal transmission switching, system of linear equations, rank-1 updates.

1 Introduction

Infrastructure networks are an important part of modern supply systems. They allow for the transportation of critical goods, such as water, electricity or gas from producers to household or industrial consumers. Due to the scale of operations, efficient planning can result in significant cost savings. This has become more pronounced in recent years, as the ongoing green energy transition and the resulting transitory systems prove particularly challenging in certain areas. For example, the German transmission system is experiencing a shift from relatively few, centralized power plants to many decentralized renewable energy producers (BDEW, 2023), as well as an imbalanced distribution of large-scale consumers and producers (BDEW, 2023; BNetzA, 2023). These challenges necessitate the expansion and adaptation of existing grid infrastructure, with the current expansion efforts lagging behind the necessary capacities (BDEW, 2023; BNetzA, 2023). This has caused an increase in costly operational measures such as the so-called

*Corresponding author.

re-dispatch, where already planned generation is changed in relatively short time periods, thus incurring higher costs, to redirect flows within the network (BNetzA, 2023). In Germany alone, re-dispatch cost has increased from €113.3 million in 2013 to €2,689.2 million in 2022 (BDEW, 2023).

As such, there is significant potential for the application of mathematical optimization in this domain. Classical flow problem formulations, where each arc of a network carries a flow to satisfy node demands, assume that the flow can be directly controlled, for example via trucks. However, oftentimes flows can only passively be controlled by adjusting network and demand parameters. Distribution through the network then depends on arc parameters and the relation between the "pressure" of adjacent nodes. The behavior of these flows can be modeled through the *potential-based flow* framework (e.g., Groß et al., 2019). There, a flow problem is extended by variable node *potentials*, measuring the "pressure" at each node of the network, as well as constant arc *conductance* and *potential function* to capture the physical properties of the flow. Flows and potentials then need to satisfy the *potential equation*, which links the incident potentials with conductance and flow following the potential function. Next to these so-called *line arcs*, *switch arcs* freely carry flow like arcs in a classical flow problem, but can be switched on or off to disrupt the synchronization of potentials of incident nodes. Based on the choice of potential function, this framework unifies different applications including lossless electrical power systems, and gas or water networks (Hendrickson and Janson, 1984; Pfetsch et al., 2026; Rockafellar, 1998).

An important additional consideration is the robustness of the network against certain contingency scenarios, like the possible outage of a set of lines. An explicit analysis might even be legally required (BNetzA, 2016). This requires the simultaneous analysis of the effects of shared decisions on the individual flows and potentials of all scenarios. Structurally, this naturally lends itself to Benders decomposition (Benders, 1962): the master problem determines the shared switching and demand variables, while subproblems verify flow feasibility and solution quality for each scenario. The application of Benders decomposition to potential-based flow problems has been thoroughly studied in the literature (Binato et al., 2001; Humpola, 2017; Khodaei and Shahidehpour, 2010; Ruiz and Conejo, 2015; Wu and Shahidehpour, 2010; Yang et al., 2017). However, these works usually focus on improvements to the master problem, for example via cuts, to reduce master problem linear relaxation solving time and branch-and-bound tree size. The efficiency of the subproblem solution algorithm remains a relatively underexplored subject in the Benders context, even though it can contribute significantly to overall running time.

2 Related Work

Early works explicitly discussing potential-based flow problems include those by Birkhoff and Diaz (1956), who already show the uniqueness of flows, and Rockafellar (1998). Recently, the problems have found increased attention. For example, Börner et al. (2025) discuss valid inequalities based on potential bounds, Groß et al. (2019) study the hardness of certain related algorithmic problems, and Pfetsch et al. (2026) provide an overview over methods and results used in the context of mathematical optimization, including also application-specific work. These applications include lossless DC power flow (Fisher et al., 2008; Hedman et al., 2009), which can be modeled with affine potential functions, as well as gas (Correa-Posada and Sánchez-Martín, 2014; Koch et al., 2015; Schmidt and Thürauf, 2024) and water networks (Burgschweiger et al., 2009; Mozafari and Zabihi, 2020). As mentioned by Birkhoff and Diaz (1956), related problems include, for example, traffic assignment (Dafermos and Sparrow, 1969; LeBlanc et al., 1975), where arcs are less attractive the more they are used.

Switch arcs correspond to different mechanisms, like valve switching and compressor activation control network configuration in gas networks (Correa-Posada and Sánchez-Martín, 2014), the opening and closing of switches in electrical systems (Fisher et al., 2008; Heidarifar and

Ghasemi, 2016), or configuring valves to en- or disable and control the direction of flow in water networks (Burgschweiger et al., 2009).

Scenarios occur in the context of robust optimization and uncertainty: a set of decisions have to be taken "here-and-now" before one of possibly infinitely many scenarios realizes, after which the "wait-and-see" decisions can be adjusted. They can, for example, cover changes in loads (Mozafari and Zabihi, 2020; Thürauf et al., 2026) or topology (Hedman et al., 2009; Li et al., 2020). Both, an aggregate view over the scenarios (Hedman et al., 2009) and a worst-case view (Li et al., 2020; Mozafari and Zabihi, 2020; Thürauf et al., 2026) can be considered.

The Benders decomposition algorithm (Benders, 1962) is a widely used exact algorithm where complicating variables are temporarily fixed to reduce overall computational burden. Rahmaniani et al. (2017) provide a survey of modern approaches. For affine (linear) potential-based flow problems, this is a commonly used method, with research focusing on formulation or cut design. A common application are linear power flow problems. For example, Binato et al. (2001) solve a power transmission network design problem with Benders decomposition, using big-M tightening and Gomory cuts. Khodaei and Shahidehpour (2010) determine generation schedules over multiple security scenarios for a security-constrained unit commitment problem. Wu and Shahidehpour (2010) propose strengthened Benders cuts for a network-constrained unit commitment problem. Ruiz and Conejo (2015) solve a three-level robust transmission expansion planning problem, using a column-and-constraint generation method with a specialized cut computation procedure.

For nonlinear problems, Benders decomposition is mostly applied to gas networks. Here, the generalized Benders decomposition algorithm of Geoffrion (1972) is used, which allows for solving nonlinear optimization problems where the subproblem is convex. For example, Humpola (2017) discusses multiple valid inequalities for passive and active gas network optimization problems in a Benders context. They also propose domain relaxation to reduce the time spent on the nonlinear subproblems. Yang et al. (2017) use a Benders decomposition approach to solve a gas flow problem, solving the nonlinear subproblems via successive linear programming, with the Benders cuts based on Lagrange multipliers on the nodal balance equations.

Overall, we observe that Benders decomposition is regularly used to solve potential-based flow problems, in particular affine ones. Most works there focus either on the general design of problem or decomposition, or on cut design. However, subproblem solving can be a significant contributor to overall running time, particularly when many scenarios need to be considered simultaneously. While stronger cuts can lead to fewer Benders iterations, and thus fewer subproblems to be solved, a specialized subproblem algorithm may still reduce overall running time.

Contribution

In this paper, we present a specialized subproblem solution algorithm as part of a Benders decomposition approach for potential-based flow problems with affine potential functions, switchable arcs and arc-based contingency scenarios. It is similar in spirit to, e.g., the knapsack algorithm used by Fischetti et al. (2017) or the semi-assignment algorithm by Contreras et al. (2011). In our case, the algorithm can be seen as a specialized simplex algorithm, where we obtain (a sufficiently large part of) an optimal basis and corresponding primal and dual solutions or dual proof of infeasibility in one step (cf. Vanderbei, 1998). It is split up into two parts.

First, a primal solution is determined to decide if and what kind of cut needs to be generated. This is based on the observation that the flow of a passive, i.e., switch arc free, affine potential-based flow problem is unique (Birkhoff and Diaz, 1956; Koch et al., 2015), together with a procedure to convert active potential-based flow problems with switch arcs fixed by the master solution into equivalent passive ones. The flow is then determined with a system of linear equations, which has the weighted Laplacian of the network as coefficient matrix.

In the second step, the primal solution is used to construct a dual linear system of equations

(SLE), which then provides the dual solution. Using a primal basis, another linear system is constructed. It shares its coefficient matrix with that of the first step. Again, this requires special handling of switch arcs. For optimality cuts, only the right-hand side differs between the two steps, and we show that the resulting dual solution is optimal, and partially unique for the chosen basis. For feasibility cuts, the matrix is extended by an additional row and column, and the right-hand side changes. Different feasibility cuts are constructed based on the component "causing" the infeasibility.

Similarly to simplex algorithms, our algorithm is extended by a factorization update scheme, where potential similarities between subproblems, rather than bases of the same problem, are exploited via low-rank updates. These occur between different master solutions and scenarios, but also between the primal and dual systems. We discuss the theoretical implications of the presented algorithm, concluding that the update scheme results in a significant reduction in theoretical complexity. We note that factorization update schemes, in particular also involving weighted Laplacian matrices such as the one we use, have been studied in other works as well, see, e.g., Alsac et al. (1983).

We demonstrate our approach on instances of the lossless DC security-constrained optimal transmission switching problem (Hedman et al., 2009), where transmission system operators minimize operating costs while considering single-line contingency scenarios. As we show, any affine potential-based flow problem instance can be transformed into such an instance, so these instances are representative for the larger class. Computational experiments on 174 instances demonstrate total running time reductions of more than 71.21 % in the geometric mean over the standard solving method, which further reduces to 97.98 % when considering only the subproblem solve time.

The paper is organized as follows: Section 3 provides a detailed problem description of the core potential-based flow problem as well as the additional aspects considered here. We then outline a branch-and-Benders-cut approach in Section 4, where each subproblem takes the form of a specific, simpler potential-based flow problem. For individual subproblems, this structure is exploited in Section 5 to arrive at an SLE-based solving algorithm. The factorization update scheme is detailed in Section 6, tying together the different subproblems solved as part of the branch-and-Benders-cut algorithm. Therein, we also briefly discuss the theoretical complexity of the proposed subproblem solving algorithm. Section 7 then provides a computational evaluation.

3 Potential-Based Flow Problems

Potential-based flows, as discussed by Groß et al. (2019), describe flow problems that often model real-life physical networks. We consider directed graphs $(V, A_L \cup A_S)$ with *nodes* V and *line* $A_L \subseteq V \times V$ and *switch arcs* $A_S \subseteq V \times V, A_L \cap A_S = \emptyset$. The set of all arcs is given by $A = A_L \cup A_S$. Each node $u \in V$ has a variable potential $\pi_u \in \mathbb{R}$ with lower and upper bounds $\underline{\pi}_u \in \mathbb{R}_{\leq 0}$ and $\bar{\pi}_u \in \mathbb{R}_{\geq 0}$, respectively. Every arc $a = (u, v) \in A_L$ has a variable *flow* $x_{(u,v)} \in \mathbb{R}$ and a *conductance* $\mu_{(u,v)} > 0$ which describes its ability to carry flow. The *potential network* (G, μ) comprises the graph G and arc conductances $\mu \in \mathbb{R}^{A_L}$. For each line arc $(u, v) \in A_L$, the interdependence between flow $x_{(u,v)}$ and potentials π_u, π_v is given by the *potential equation*

$$\pi_u - \pi_v = \psi_{(u,v)}(x_{(u,v)})\mu^{-1}_{(u,v)},$$

where $\psi_{(u,v)} : \mathbb{R} \to \mathbb{R}$ is a *potential function*. Potential functions are continuous and strictly increasing. Rearranging yields the flow as a function of the potentials:

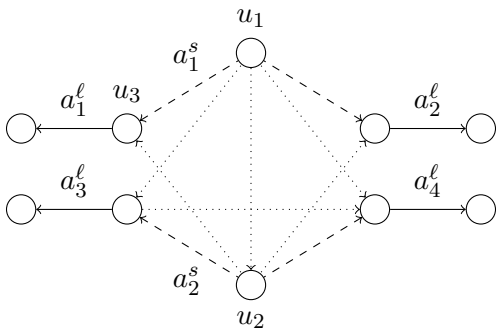
$$x_{(u,v)} = \psi^{-1}_{(u,v)}((\pi_u - \pi_v)\mu_{(u,v)}).$$

The flow is limited by lower and upper bounds $\underline{x}_{(u,v)} \in \mathbb{R}_{\leq 0}$ and $\bar{x}_{(u,v)} \in \mathbb{R}_{\geq 0}$, respectively.

Switch arcs $(u, v) \in A_S$ can be switched on or off to connect or disconnect nodes in the graph, denoted by switching state $z_{(u,v)} \in \{0, 1\}$. If a switch arc $(u, v) \in A_S$ is switched "on" ($z_{(u,v)} = 1$), the incident node potentials must be equal, $\pi_u = \pi_v$. If switched "off" ($z_{(u,v)} = 0$), the arc flow must be zero, $x_{(u,v)} = 0$, and the potentials are no longer directly linked. In the latter case, the difference of the respective potentials is confined to lower and upper limits $\underline{\pi}_{(u,v)} \in \mathbb{R}$ and $\bar{\pi}_{(u,v)} \in \mathbb{R}$, respectively, which are selected such that they can never restrict the choice of potentials.

We refer to switched on and off switch arcs as "active" and "inactive," respectively. We require that there are no two nodes in the same switch arc-connected component in the graph (V, A_S) that are incident to the same line arc. Additionally, for any two such components, there can be at most one line arc that is incident to a node in each component. Figure 1 shows an example graph with both line and switch arcs.

Figure 1: Network topology.



Note. Line arcs are represented by solid lines, active and inactive switch arcs by dashed and dotted lines, respectively. The active switch arcs and incident nodes form switch arc components.

Each node has an adjustable *demand* $b_u \in [\underline{b}_u, \bar{b}_u]$, $\underline{b}_u, \bar{b}_u \in \mathbb{R}$, $\underline{b}_u \leq \bar{b}_u$, with per-unit cost $F_u \in \mathbb{R}$. The demands are always balanced, i.e.,

$$\sum_{u \in V} b_u = 0,$$

due to the flow conservation constraints. If demands are fixed to a value and there are no switch arcs, $A_S = \emptyset$, the network is called *passive*. Otherwise, it is *active*.

A potential-based flow (x, π, z, b) then comprises arc flows $x \in \mathbb{R}^A$, node potentials $\pi \in \mathbb{R}^V$, switching states $z \in \{0, 1\}^{A_S}$ and demands $b \in \mathbb{R}^V$. It is considered feasible if it fulfills (1a)–(1h):

$$z_{(u,v)} \underline{x}_{(u,v)} \leq x_{(u,v)} \leq z_{(u,v)} \bar{x}_{(u,v)} \qquad \forall (u, v) \in A_S \tag{1a}$$

$$(1 - z_{(u,v)}) \underline{\pi}_{(u,v)} \leq \pi_u - \pi_v \leq (1 - z_{(u,v)}) \bar{\pi}_{(u,v)} \qquad \forall (u, v) \in A_S \tag{1b}$$

$$\sum_{(v,u) \in \delta^-(u)} x_{(v,u)} - \sum_{(u,v) \in \delta^+(u)} x_{(u,v)} = b_u \qquad \forall u \in V \tag{1c}$$

$$\psi_{(u,v)}^{-1}((\pi_u - \pi_v)\mu_{(u,v)}) = x_{(u,v)} \qquad \forall (u, v) \in A_L \tag{1d}$$

$$b_u \in [\underline{b}_u, \bar{b}_u] \qquad \forall u \in V \tag{1e}$$

$$z_{(u,v)} \in \{0, 1\} \qquad \forall (u, v) \in A_S \tag{1f}$$

$$x_{(u,v)} \in [\underline{x}_{(u,v)}, \bar{x}_{(u,v)}] \qquad \forall (u, v) \in A \tag{1g}$$

$$\pi_u \in [\underline{\pi}_u, \bar{\pi}_u] \qquad \forall u \in V \tag{1h}$$

Constraints (1a) and (1b) ensure that switch arcs carry no flow when off and have equal incident potentials when on. Here, for all switch arcs $(u, v) \in A_S$, the "big-M" parameters $\underline{x}_{(u,v)}, \bar{x}_{(u,v)} \in \mathbb{R}$ are chosen such that the bound can never be met while the arc is on, and analogously for $\underline{\pi}_{(u,v)}, \bar{\pi}_{(u,v)} \in \mathbb{R}$. Constraint (1c) enforces flow conservation: the net flow at each node meets its demand. Constraint (1d) is the potential equation relating flows and potentials. Constraints (1e)–(1h) specify the variables' domains.

Extended potential-based flow problems. In addition to this basic potential-based flow problem, we consider scenario-based robustness, line arc overloading, and switching limits. Robustness is enforced via a set of *contingencies* $\mathcal{C} \subseteq 2^A$, where each contingency $C \in \mathcal{C}$ represents a scenario in which all contained arcs fail. The methods discussed in this work extend naturally to arbitrary contingencies with individual bounds, (fixed) demands or conductances, however we focus on (single) arc contingencies for the sake of simplicity. The set $\mathcal{C}$ may include the empty set $\emptyset$, representing the base case without failures. Switching and demand decisions are made independently of which contingency occurs, while flows and potentials adapt to each contingency. We denote by $A_L^C \coloneqq A_L \setminus C$ and $A_S^C \coloneqq A_S \setminus C$, respectively, the sets of line and switch arcs unaffected by contingency C, and by $A^C \coloneqq A_L^C \cup A_S^C$ the set of all such arcs.

In each contingency $C \in \mathcal{C}$, each line arc $(u, v) \in A_L$ can be overloaded by $o_{(u,v)}^C \in \mathbb{R}_{\geq 0}$ beyond its soft bounds $\underline{x}_{(u,v)}, \tilde{x}_{(u,v)} \in \mathbb{R}$ at cost $F^o \in \mathbb{R}_{\geq 0}$ per unit, within the already introduced hard bounds $\underline{x}_{(u,v)}$ and $\bar{x}_{(u,v)}$, i.e., $\underline{x}_{(u,v)} \leq \underline{x}_{(u,v)} \leq \tilde{x}_{(u,v)} \leq \bar{x}_{(u,v)}$. Overloaded arcs contribute to the objective function for each contingency individually. Alternatively, ways to aggregate subproblem costs, e.g., the maximum over all subproblems, could also be considered.

Each switch arc $(u, v) \in A_S$ has a default state. We collect arcs switched on by default in set $A_S^1 \subseteq A_S$, and in set $A_S^0 = A_S \setminus A_S^1$ those which are inactive in the default state. At most $Z \in \mathbb{N}$ switch arcs may be switched away from their defaults. The graph has to be "islanding-free" in all contingencies. That is, all nodes $u \in V$ where demands are not fixed to zero, i.e., $\underline{b}_u \neq 0$ or $\bar{b}_u \neq 0$, have to be part of the same connected component induced by the active arcs (arcs that are switched on and not part of the contingency). This is a looser requirement than full connectedness in that it allows for parts of the graph to be disconnected by switching or contingencies, as long as only one of them is relevant to the distribution of demands.

All these parameters define a potential-based flow problem $\mathcal{P} = (G, \mu, \psi, \mathcal{C}, H)$, where

$$H = (\underline{b}, \bar{b}, \underline{x}, \bar{x}, \underline{x}, \tilde{x}, \underline{\pi}, \bar{\pi}, A_S^0, Z, F, F^o)$$

collects bounds, default state, and costs. Should the contingencies be irrelevant in a statement, we omit them from the description of $\mathcal{P}$.

We are ready to state the complete extended model (omitting the connectedness constraints) in Constraints (2a)–(2n). Arc flows and node potentials now materialize in every scenario. The corresponding variables are thus superscripted with the respective contingency.

$$\min \quad \sum_{u \in V} F_u b_u + \sum_{C \in \mathcal{C}} \sum_{(u,v) \in A_L^C} F^o o^C_{(u,v)} \tag{2a}$$

s.t. $\forall C \in \mathcal{C}:$

$$\sum_{(u,v) \in A_S^1} \big(1 - z_{(u,v)}\big) + \sum_{(u,v) \in A_S^0} z_{(u,v)} \leq Z \tag{2b}$$

$$z_{(u,v)} \underline{x}_{(u,v)} \leq x^C_{(u,v)} \leq z_{(u,v)} \bar{x}_{(u,v)} \qquad \forall (u,v) \in A_S^C \tag{2c}$$

$$(1 - z_{(u,v)}) \underline{\pi}_{(u,v)} \leq \pi^C_u - \pi^C_v \leq (1 - z_{(u,v)}) \bar{\pi}_{(u,v)} \qquad \forall (u,v) \in A_S^C \tag{2d}$$

$$\sum_{(v,u) \in \delta^-(u) \setminus C} x^C_{(v,u)} - \sum_{(u,v) \in \delta^+(u) \setminus C} x^C_{(u,v)} = b_u \qquad \forall u \in V \tag{2e}$$

$$\psi^{-1}_{(u,v)}((\pi^C_u - \pi^C_v)\mu_{(u,v)}) = x^C_{(u,v)} \qquad \forall (u,v) \in A_L^C \tag{2f}$$

$$\underset{\sim}{x}_{(u,v)} - o^C_{(u,v)} \leq x^C_{(u,v)} \leq \tilde{x}_{(u,v)} + o^C_{(u,v)} \qquad \forall (u,v) \in A_L^C \tag{2g}$$

$$\underline{x}_{(u,v)} \leq x^C_{(u,v)} \leq \bar{x}_{(u,v)} \qquad \forall (u,v) \in A_L^C \tag{2h}$$

$$\underline{\pi}_u \leq \pi^C_u \leq \bar{\pi}_u \qquad \forall u \in V \tag{2i}$$

$$b_u \in [\underline{b}_u, \bar{b}_u] \qquad \forall u \in V \tag{2j}$$

$$z_{(u,v)} \in \{0,1\} \qquad \forall (u,v) \in A_S \tag{2k}$$

$$x^C_{(u,v)} \in \mathbb{R} \qquad \forall (u,v) \in A^C \tag{2l}$$

$$\pi^C_u \in \mathbb{R} \qquad \forall u \in V \tag{2m}$$

$$o^C_{(u,v)} \geq 0 \qquad \forall (u,v) \in A_L^C \tag{2n}$$

The objective function minimizes total demand and overload costs across all contingencies. Constraint (2b) enforces the switching limit on deviations from default settings. The core potential-based flow Constraints (1a)–(1d) are replicated for each contingency via Constraints (2c)–(2f). Constraint (2g) tracks line arc overloading beyond soft bounds. Constraints (2h) and (2i) enforce the bounds on flows and potentials for each contingency. Constraints (2j)–(2n) specify bounds and domains of the variables.

Affine Potential Functions. In this work, we focus on instances with affine potential functions with positive slope:

$$\psi_{(u,v)}(x) = r_{(u,v)} x + s_{(u,v)}, \, \forall x \in \mathbb{R}, (u,v) \in A_L,$$

where $r_{(u,v)} \in \mathbb{R}_{>0}$ and $s_{(u,v)} \in \mathbb{R}$. By standard transformation, any such function can be replaced by the identity function:

Proposition 1. *Let $\mathcal{P} = (G, \mu, \psi, H)$ be a potential-based flow problem with affine potential functions ψ. Then, there exists an equivalent potential-based flow problem $\mathcal{P}' = (G, \mu', \psi', H')$ where every potential function is the identity function $id : \mathbb{R} \to \mathbb{R}$, $id(x) = x$ such that any feasible solution (x', π, z, b') to $\mathcal{P}'$ can be transformed into a feasible solution (x, π, z, b) to $\mathcal{P}$ with the same objective function value, and vice versa.*

Proof. We transform the original instance via arc parameter rescaling and demand adjustment.

For each $(u,v) \in A_L$, inverting the potential function in Constraint (1d) yields

$$\begin{aligned} x_{(u,v)} &= \psi_{(u,v)}^{-1}((\pi_u - \pi_v)\mu_{(u,v)}) \\ &= \frac{(\pi_u - \pi_v)\mu_{(u,v)} - s_{(u,v)}}{r_{(u,v)}} \\ &= (\pi_u - \pi_v)\frac{\mu_{(u,v)}}{r_{(u,v)}} - \frac{s_{(u,v)}}{r_{(u,v)}}. \end{aligned}$$

The rescaled conductance is

$$\mu'_{(u,v)} \coloneqq \frac{\mu_{(u,v)}}{r_{(u,v)}},$$

yielding

$$\begin{aligned} x'_{(u,v)} \coloneqq& (\pi_u - \pi_v)\mu'_{(u,v)} \\ =& (\pi_u - \pi_v)\frac{\mu_{(u,v)}}{r_{(u,v)}} - \frac{s_{(u,v)}}{r_{(u,v)}} + \frac{s_{(u,v)}}{r_{(u,v)}} \\ =& x_{(u,v)} + \frac{s_{(u,v)}}{r_{(u,v)}}. \end{aligned}$$

The shift $s_{(u,v)}/r_{(u,v)}$ is absorbed into all flow bounds, for instance, the transformed lower bound becomes $\underline{x}'_{(u,v)} = \underline{x}_{(u,v)} + s_{(u,v)}/r_{(u,v)}$. These shifts also affect flow conservation Constraints (1c). Substituting

$$x_{(u,v)} = x'_{(u,v)} - \frac{s_{(u,v)}}{r_{(u,v)}} \tag{3}$$

gives

$$\begin{aligned} b_u &= \sum_{(v,u)\in\delta^-(u)} x_{(v,u)} - \sum_{(u,v)\in\delta^+(u)} x_{(u,v)} \\ &= \sum_{(v,u)\in\delta^-(u)} x'_{(v,u)} - \sum_{(u,v)\in\delta^+(u)} x'_{(u,v)} \\ &\quad - \sum_{(v,u)\in\delta^-(u)} \frac{s_{(v,u)}}{r_{(v,u)}} + \sum_{(u,v)\in\delta^+(u)} \frac{s_{(u,v)}}{r_{(u,v)}}. \end{aligned}$$

Rearranging yields

$$\sum_{(v,u)\in\delta^-(u)} x'_{(v,u)} - \sum_{(u,v)\in\delta^+(u)} x'_{(u,v)} = b'_u,$$

with transformed demand

$$b'_u \coloneqq b_u + \sum_{(v,u)\in\delta^-(u)} \frac{s_{(v,u)}}{r_{(v,u)}} - \sum_{(u,v)\in\delta^+(u)} \frac{s_{(u,v)}}{r_{(u,v)}} \tag{4}$$

for all $u \in V$. We shift the demand bounds accordingly

$$\underline{b}'_u = \underline{b}_u + \sum_{(u,v)\in\delta^-(u)} \frac{s_{(u,v)}}{r_{(u,v)}} - \sum_{(u,v)\in\delta^+(u)} \frac{s_{(u,v)}}{r_{(u,v)}},$$

analogously for $\bar{b}'_u$, and add the term

$$-F_u \cdot \left(\sum_{(v,u)\in\delta^-(u)} \frac{s_{(v,u)}}{r_{(v,u)}} - \sum_{(u,v)\in\delta^+(u)} \frac{s_{(u,v)}}{r_{(u,v)}} \right)$$

to the objective function.

To summarize, the transformed instance is obtained from the original instance by, for all line arcs $(u, v) \in A_L$, replacing the conductance $\mu_{(u,v)}$ by the rescaled $\mu'_{(u,v)}$, shifting flow bounds $\underline{x}_{(u,v)}$, $\bar{x}_{(u,v)}$, $\underline{x}_{(u,v)}$, and $\tilde{x}_{(u,v)}$ by $s_{(u,v)}/r_{(u,v)}$, and replacing its potential function with the identity potential function

$$x'_{(u,v)} = \mu'_{(u,v)}(\pi_u - \pi_v) \quad \forall (u, v) \in A_L;$$

and for all nodes $u \in V$, shifting its demand bounds $\underline{b}_u, \bar{b}_u$ and adding a term to the objective to match the adjusted demand b_u. The transformed instance is equivalent to the original instance: following Equations (3) and (4), any feasible solution to the transformed instance can be transformed into a feasible solution to the original instance with the same objective function value, and vice versa. □

In the following, we can thus assume all potential functions to be the identity function. As such, we can substitute all flow variables on line arcs by a term over potential variables following the potential equation. For ease of notation, we still refer to flows where helpful, but line arc flows and potential equations are not explicitly part of the following models.

Problem (1a)–(1h) is NP-hard if potential functions are the identity function (Lehmann et al., 2014), and thus following Proposition 1 also if they are affine.

4 A Benders Decomposition Approach

We decompose the problem Constraints (2a)–(2n) into a master problem (MP) and one subproblem (SP) per contingency $C \in \mathcal{C}$. The MP initially consists only of demand b and switching z variables as well as the constraint on switching actions (2b) and constraint

$$\sum_{u \in V} b_u = 0$$

to enforce global demand balance. Additionally, it has one Benders recourse variable $\eta_C \geq 0$ for each contingency $C \in \mathcal{C}$ that is added to the objective function, reflecting a lower bound estimate of that contingency's objective function contribution for any master solution.

Subproblems contain the remaining constraints and variables, with master variables as fixed parameters. Let $(\hat{b}, \hat{z}, \hat{\eta}_C)$, $\hat{b} \in \mathbb{R}^V$, $\hat{z} \in \{0, 1\}^{A_S}$, $\hat{\eta}_C \in \mathbb{R}$ denote a master solution with values for demands, topology, and the Benders variable of contingency $C \in \mathcal{C}$, respectively. We denote the corresponding C-th subproblem as $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$, and its optimal objective function value, if feasible, as $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^* \in \mathbb{R}$. The subproblem $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ is a potential-based flow problem where contingency arcs $(u, v) \in C$ are removed, and the master problem decisions are fixed. We call these networks with fixed demands and switch arc states *semi-passive*. The subproblem model for $C \in \mathcal{C}$ is given in Model (5a)–(5j). For reference, we indicate the dual variables (or "Farkas multipliers," respectively) corresponding to the respective constraints in square brackets.

$$
\begin{aligned}
& S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^* = \\
\min \quad & \sum_{(u,v)\in A_L^C} F^o o^C_{(u,v)} && && && (5a)\\
\text{s.t.} \quad & \hat{z}_{(u,v)}\underline{x}_{(u,v)} \le x^C_{(u,v)} \le \hat{z}_{(u,v)}\bar{x}_{(u,v)} && [\underline{\tau}^C, \bar{\tau}^C] && \forall (u,v)\in A_S^C && (5b)\\
& (1-\hat{z}_{(u,v)})\underline{\pi}_{(u,v)} \le \pi^C_u - \pi^C_v \le (1-\hat{z}_{(u,v)})\bar{\pi}_{(u,v)} && [\underline{\alpha}^C, \bar{\alpha}^C] && \forall (u,v)\in A_S^C && (5c)\\
& \sum_{(v,u)\in\delta^-(u)\cap A_L^C} (\pi^C_v - \pi^C_u)\mu_{(v,u)} \\
& - \sum_{(u,v)\in\delta^+(u)\cap A_L^C} (\pi^C_u - \pi^C_v)\mu_{(u,v)} \\
& \qquad + \sum_{(v,u)\in\delta^-(u)\cap A_S^C} x^C_{(v,u)} \\
& \qquad - \sum_{(u,v)\in\delta^+(u)\cap A_S^C} x^C_{(u,v)} = \hat{b}_u && [\varphi^C] && \forall u\in V && (5d)\\
& \underset{\sim}{x}_{(u,v)} - o^C_{(u,v)} \le (\pi^C_u - \pi^C_v)\mu_{(u,v)} \le \tilde{x}_{(u,v)} + o^C_{(u,v)} && [\underline{\sigma}^C, \bar{\sigma}^C] && \forall (u,v)\in A_L^C && (5e)\\
& \underline{x}_{(u,v)} \le (\pi^C_u - \pi^C_v)\mu_{(u,v)} \le \bar{x}_{(u,v)} && [\underline{\gamma}^C, \bar{\gamma}^C] && \forall (u,v)\in A_L^C && (5f)\\
& \underline{\pi}_u \le \pi^C_u \le \bar{\pi}_u && [\underline{\xi}^C, \bar{\xi}^C] && \forall u\in V && (5g)\\
& x^C_{(u,v)} \in \mathbb{R} && && \forall (u,v)\in A_S^C && (5h)\\
& \pi^C_u \in \mathbb{R} && && \forall u\in V && (5i)\\
& o^C_{(u,v)} \in \mathbb{R}_{\ge 0} && && \forall (u,v)\in A_L^C && (5j)
\end{aligned}
$$

The decomposed problem is solved using a branch-and-Benders-cut algorithm (cf. Rahmaniani et al., 2017). The MP is solved as usual using a branch-and-bound algorithm. Every time an integer feasible MP solution is obtained during the algorithm, it needs to be verified with all subproblems, and is only valid if it is "accepted" by all of them. For this, each subproblem is solved with the master solution's values as parameters. If an SP is infeasible, a feasibility cut is added to the MP. This feasibility cut is based on the values of a dual direction of unboundedness ("Farkas multipliers"), and cuts off at least the master solution causing the infeasibility. With $\underline{\tau}^{C*}$ and $\bar{\tau}^{C*}$ denoting dual direction values of the lower and upper bound, respectively, of Constraint (5b), and analogously for the other constraints, a feasibility cut is

$$
\begin{aligned}
& \sum_{(u,v)\in A_S^C} \Big(z_{(u,v)}\bar{x}_{(u,v)}\bar{\tau}^{C*}_{(u,v)} - z_{(u,v)}\underline{x}_{(u,v)}\underline{\tau}^{C*}_{(u,v)} \\
& \qquad + (1-z_{(u,v)})\bar{\pi}_{(u,v)}\bar{\alpha}^{C*}_{(u,v)} - (1-z_{(u,v)})\underline{\pi}_{(u,v)}\underline{\alpha}^{C*}_{(u,v)} \Big) \\
& + \sum_{(u,v)\in A_L^C} \Big(\tilde{x}_{(u,v)}\bar{\sigma}^{C*}_{(u,v)} - \underset{\sim}{x}_{(u,v)}\underline{\sigma}^{C*}_{(u,v)} + \bar{x}_{(u,v)}\bar{\gamma}^{C*}_{(u,v)} - \underline{x}_{(u,v)}\underline{\gamma}^{C*}_{(u,v)} \Big) \\
& + \sum_{u\in V} \big(b_u\varphi^{C*}_u + \underline{\pi}_u\underline{\xi}^{C*}_u + \bar{\pi}_u\bar{\xi}^{C*}_u \big) \le 0.
\end{aligned} \tag{6}
$$

Otherwise, if an SP is feasible, and the optimal objective function value of $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ is strictly larger than the master solution value for η_C, an optimality cut is added to the MP. This cut is based on the values of an optimal dual solution. Denoting these the same way as for the feasibility cut, the cut inequality only differs in the right-hand side, here η_C, i.e., the Benders recourse variable. By adding these cuts, changes in switching actions and demands in the MP

reducing line arc overloads in contingencies are thereby incentivized. If neither cut is added in any subproblem, the master solution is accepted as a valid solution in the current node of the branch-and-bound tree.

5 Solving Subproblems as Systems of Linear Equations

In this section, we focus on solving a single subproblem $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ after obtaining the master solution, i.e., fixed values for demand b, topology z and η_C. We discuss how a constant number of systems of linear equations (SLE) are sufficient to solve subproblems, i.e., check for and compute optimality and feasibility cuts. The next section then integrates this as part of the larger branch-and-Benders-cut algorithm, where multiple subproblems are solved in succession.

We assume that the graph is islanding-free. That is, all nodes with demands not fixed to zero by their bounds are connected in the graph induced by the active arcs, i.e., those switched-on and not part of a given contingency. This can be enforced through, e.g., a separate separation procedure. We verify connectedness by computing a minimum cut on the contracted graph (see upcoming section) with line arcs not in the contingency contracted as well. Additionally, all nodes that are not part of the connected component containing these demands are removed with their incident arcs, as we can assume that they do not carry flow (or potentials) in an optimal solution.

The subproblem solving algorithm is motivated by the comparatively high computational effort of the simplex algorithm—it requires solving a potentially very high number of linear equation systems, whereas our algorithm requires solving only one or two of equal or smaller size. It is roughly divided into two parts: first we solve a reduced version of the primal subproblem, free of soft and hard variable bounds, the objective function, and switch arcs. From this, we obtain flow and potential variable values for all arcs and nodes. The potential differences as well as relevant flow values of a switch arc-free network are unique for a given master solution (Birkhoff and Diaz, 1956). With these, we can thus check whether the subproblem is feasible, and, if so, compute its objective function value. This determines whether the subproblem yields an optimality or a feasibility cut, or neither. If it does yield a cut, we solve a reformulated version of the dual subproblem as an SLE to obtain all necessary dual variable values to construct the respective cut. While similar, this dual SLE differs between optimality and feasibility cuts, as detailed later.

The remainder of this section is structured as follows: in Subsection 5.1, we discuss an arc contraction procedure for fixed switch arcs to obtain a switch arc-free graph yielding an equivalent line arc flow. Subsection 5.2 discusses the first part of the algorithm, i.e., how to construct the primal SLE from a given master solution. Subsections 5.3 and 5.4 then explain the second part of the algorithm, i.e., how to construct the dual SLE for an optimality or feasibility cut, respectively, from a given master and primal SLE solution. The resulting improvements in computational complexity are discussed jointly in the next section.

5.1 Switch Arc Contraction

With all switching variables fixed in the subproblem, and thus a semi-passive network, we can simplify the underlying network by substituting all switch arcs in a way that preserves the flow on line arcs. This is done by "contracting" switch arcs iteratively, eventually resulting in a graph with only line arcs. The uniqueness of flows with regard to a passive network was observed by, e.g., Birkhoff and Diaz (1956), so the contraction procedure can be seen as a precondition for the uniqueness of flows in our subproblems. Similarly, contracted graphs are used for constructing the dual SLEs. For both primal and dual solutions, we first determine most variable values on the contracted graph, and then any remaining values as necessary, as discussed in the following sections.

We first show that we can disregard inactive switch arcs. For ease of readability, we omit any notational reference to the contingency. Then, we describe the contraction procedure that eliminates active switch arcs.

Proposition 2. *Let $\mathcal{P} = ((V, A_L \cup A_S), \mu, \psi, H)$ be a potential-based flow problem with a semi-passive network fixed by $\hat{b} \in \mathbb{R}^V$ and $\hat{z} \in A_S$. Let $A_S^{\hat{z}} = \{(u,v) \in A_S \mid \hat{z}_{(u,v)} = 1\}$ be the set of active switch arcs. Any optimal solution to the identical problem without inactive switch arcs $\mathcal{P}' = ((V, A_L \cup A_S^{\hat{z}}), \mu, \psi, H)$ can be transformed into an optimal solution to $\mathcal{P}$.*

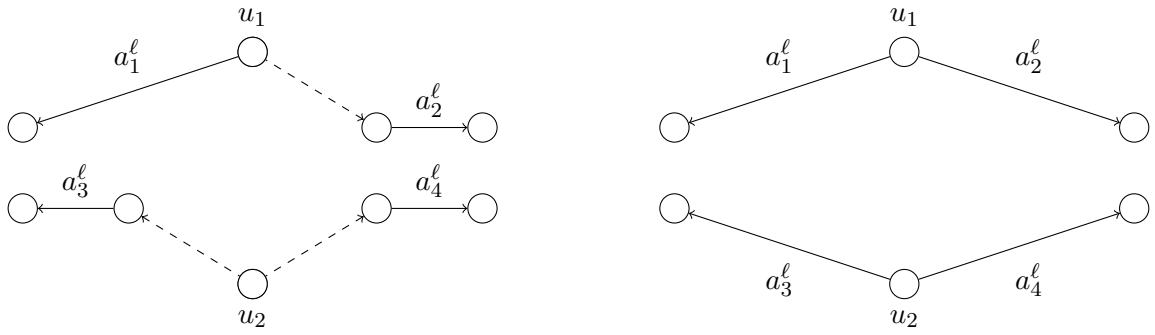
Proof. Let $(\pi^*, x^{*\prime})$, $\pi \in \mathbb{R}^V$, $x^{*\prime} \in \mathbb{R}^{A_S^{\hat{z}}}$, be an optimal solution to $\mathcal{P}'$. Then, (π^*, x^*), where

$$x^*_{(u,v)} = \begin{cases} x^{*\prime}_{(u,v)}, & (u,v) \in A_S^{\hat{z}} \\ 0, & \text{else} \end{cases}$$

is a feasible solution to $\mathcal{P}$: all constraints except for Constraints (5b) and (5c) for all $(u,v) \notin A_S^{\hat{z}}$ as well as Constraints (5d) of the incident nodes are unaffected and thus trivially fulfilled. Let $(u,v) \notin A_S^{\hat{z}}$ be an inactive arc. Since it is inactive, i.e., $x^*_{(u,v)} = 0$, Constraints (5b) are fulfilled with equality, and Constraints (5c) are always fulfilled by construction. Constraints (5d) for the incident nodes $u, v \in V$ sum over the additional variable $x_{(u,v)}$, but since $x^*_{(u,v)} = 0$, the result does not change.

The solution (π^*, x^*) is optimal for $\mathcal{P}$, otherwise, there must be a solution $(\pi^{*\prime\prime}, x^{*\prime\prime})$ for $\mathcal{P}$ with improved objective function value and $x^{*\prime\prime}_{(u,v)} = 0$ for all $(u,v) \notin A_S^{\hat{z}}$. But then, $(\pi^{*\prime\prime}, x^{*\prime\prime})$ reduced to arcs in $A_S^{\hat{z}}$ is clearly feasible for $\mathcal{P}'$, so $(\pi^*, x^{*\prime})$ can not be optimal for $\mathcal{P}'$. □

Figure 2: Switch arc component contraction.



(a) After contracting arc a_1^s. (b) Full switch arc contraction.

Note. Contraction of switch arc components from Figure 1 after removing inactive switch arcs. Solid lines represent line arcs, dashed lines active switch arcs. In (a), the arc a_1^ℓ is now directly connected to u_1. In (b), the components have fully been merged into the representative nodes u_1 and u_2.

For a specific contingency C denote by $A_S^{\hat{z},C} \subseteq \{(u,v) \in A_S^C \mid \hat{z}_{(u,v)} = 1\}$ the set of remaining, active switch arcs in the network. Let $\mathcal{W}^{\hat{z},C} \subseteq 2^V$ denote the set of connected components in the subgraph $(V, A_S^{\hat{z},C})$, the *active switch arc components*. By Constraints (5c), the nodes of any such component share the same potential. To determine the potential of all nodes, it thus suffices to consider each connected component in the graph as a single representative node, the potential of which represents the potential of all members of the component. We iteratively replace each arc $(u,v) \in A_S^{\hat{z},C}$ and its incident nodes in the component $W \in \mathcal{W}^{\hat{z},C}$ with a single contracted node: first, we remove (u,v) from $A_S^{\hat{z},C}$. Then, we replace all arcs $(u',v), (v,v') \in A_L^C \cup A_S^{\hat{z},C}$ that are incident to v with (u',u) and (u,v'), respectively (while keeping their other properties the same). If a node is adjacent to both u and v, i.e., replacing the arc incident to v would result

in two parallel arcs, we do not add the second arc. As no line arc can be incident to two nodes of the same switch arc-connected component, such an arc can only be a switch arc. Lastly, we remove v from V and set the demand of the contracted node to $\hat{b}_u + \hat{b}_v$.

We denote the remaining node of component W by $u(W)$. The set of (line) arcs in the contracted graph is denoted by $A_L^{\hat{z},C}$. Refer to Figure 2 for examples.

Solving the contracted problem is sufficient to obtain an optimal primal solution to the original problem. As before, our notation does not refer to the contingency in the result and proof.

Theorem 1. *Let $\mathcal{P} = ((V, A_L \cup A_S), \mu, \psi, H)$ be a potential-based flow problem with a semi-passive network fixed by $\hat{b} \in \mathbb{R}^V$ and $\hat{z} \in A_S$. Let $\mathcal{P}' = ((V', A'_L \cup A'_S), \mu', \psi', H')$, $A'_S = \emptyset$, be the potential-based flow problem on the corresponding passive contracted network. Any optimal solution $(\pi^{*\prime}, x^{*\prime})$ for $\mathcal{P}'$ can be transformed directly into an optimal solution (π^*, x^*) for $\mathcal{P}$, and vice versa.*

Proof. Following Proposition 2, we can disregard all inactive switch arcs of $\mathcal{P}$. For the transformed solution, for all switch arc components $W \in \mathcal{W}^{\hat{z}}$, we set the potential of all contained nodes to that of the representative node,

$$\pi^*_u = \pi^{*\prime}_{u(W)} \quad \forall u \in W, W \in \mathcal{W}^{\hat{z}}.$$

The switch arc flows x^* need to be assigned values such that the flow conservation Constraints (5d) are fulfilled. Let again $A_S^{\hat{z}} = \{(u,v) \in A_S \mid \hat{z}_{(u,v)} = 1\}$ denote the set of active switch arcs. We determine a spanning tree on the subgraph $(W, A_S^{\hat{z}} \cap (W \times W))$ for each component $W \in \mathcal{W}^{\hat{z}}$. All arcs of that subgraph not in the spanning tree are assigned a flow of zero, in analogy to non-basic variables in a network simplex algorithm. Then, starting from the leaves of the spanning tree until all variables are determined, there always is a node in W where the flow conservation equation has only one undetermined variable, the flow of an incident switch arc. It is easy to see that this solution is feasible and has the same objective function value in $\mathcal{P}$, the flows on line arcs remain unchanged.

Conversely, we show that for any feasible solution of $\mathcal{P}$, the contraction procedure preserves the flow on line arcs, i.e., that a solution with its variables adapted as described remains feasible in $\mathcal{P}'$. Let $x^* \in \mathbb{R}^{A_L \cup A_S^{\hat{z}}}$ denote an optimal flow solution, with line arc flows obtained following the potential equations. Let $(u,v) \in A_S^{\hat{z}}$ be the first arc we contract (into node u). By Constraints (5d), it holds that

$$\sum_{(v,u) \in \delta^-(u)} x^*_{(v,u)} - \sum_{(u,v) \in \delta^+(u)} x^*_{(u,v)} = \hat{b}_u \tag{7}$$

and

$$\sum_{(u,v) \in \delta^-(v)} x^*_{(u,v)} - \sum_{(v,u) \in \delta^+(v)} x^*_{(v,u)} = \hat{b}_v. \tag{8}$$

The adapted Constraint (5d) at the contracted node u is

$$\begin{aligned} &\sum_{(v,u) \in \delta^-(u)} x_{(v,u)} - \sum_{(u,v) \in \delta^+(u)} x_{(u,v)} \\ &+ \sum_{(u,v) \in \delta^-(v)} x_{(u,v)} - \sum_{(v,u) \in \delta^+(v)} x_{(v,u)} \\ &\qquad = \hat{b}_u + \hat{b}_v, \end{aligned}$$

which is the sum of Equations (7) and (8), and thus the solution fulfills the new constraint. It is clear that all other constraints remain fulfilled, and that the adapted solution is still optimal. By induction, this holds after repeating the procedure on the remaining arcs in $A_S^{\hat{z}}$ as well. □

Any semi-passive network can thus be simplified to a passive network, and the results by Birkhoff and Diaz (1956) apply.

5.2 Primal Solving

In this subsection, we construct the "primal" SLE, and show how it is used to obtain a full primal solution. We first show how this SLE is constructed from a master solution, and provide a proof showing that it always has a unique solution (Birkhoff and Diaz, 1956). This unique solution then allows us to determine if and which cut needs to be generated. Additionally, certain of the resulting primal variable solution values are needed to construct the dual SLEs in the following sections.

Following Proposition 2 and Theorem 1, we are able to omit all switch arcs. For ease of notation, we assume without loss of generality that the graph is already switch arc free. Then, we can simplify the formulation given in Constraints (5a)–(5j) to the one in (9a)–(9g) by removing Constraints (5b) and (5c) and variables $x^C_{(u,v)}$ for $(u,v) \in A^C_S$.

$$
\begin{aligned}
& S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^* = \\
\min \quad & \sum_{(u,v)\in A^C_L} F^o o^C_{(u,v)} && && && (9a)\\
\text{s. t.} \quad & \sum_{(v,u)\in\delta^-(u)\setminus C} (\pi^C_v - \pi^C_u)\mu_{(v,u)} \\
& - \sum_{(u,v)\in\delta^+(u)\setminus C} (\pi^C_u - \pi^C_v)\mu_{(u,v)} = \hat{b}_u && [\varphi^C] && \forall u \in V && (9b)\\
& \underline{x}_{(u,v)} - o^C_{(u,v)} \le (\pi^C_u - \pi^C_v)\mu_{(u,v)} \le \tilde{x}_{(u,v)} + o^C_{(u,v)} && [\underline{\sigma}^C, \bar{\sigma}^C] && \forall (u,v) \in A^C_L && (9c)\\
& \underline{x}_{(u,v)} \le (\pi^C_u - \pi^C_v)\mu_{(u,v)} \le \bar{x}_{(u,v)} && [\underline{\gamma}^C, \bar{\gamma}^C] && \forall (u,v) \in A^C_L && (9d)\\
& \underline{\pi}_u \le \pi^C_u \le \bar{\pi}_u && [\underline{\xi}^C, \bar{\xi}^C] && \forall u \in V && (9e)\\
& \pi^C_u \in \mathbb{R} && && \forall u \in V && (9f)\\
& o^C_{(u,v)} \in \mathbb{R}_{\ge 0} && && \forall (u,v) \in A^C_L && (9g)
\end{aligned}
$$

Constraints (9b) of the simplified model then yield a system of linear equations on the potential variables. We call it the *primal* SLE and denote it by $L^{\hat{z},C,p}$. With $\delta(u) \setminus C$ as the set of in- and outgoing arcs of node $u \in V$, the coefficient matrix of the primal SLE is given by the (negated) weighted Laplacian matrix of the graph, i.e.,

$$
D^{\hat{z},C}_{u,v} = \begin{cases} -\sum_{(u,v)\in\delta(u)\setminus C} \mu_{(u,v)}, & u = v, \\ \mu_{(u,v)}, & (u,v) \in A^C_L, \\ \mu_{(v,u)}, & (v,u) \in A^C_L, \\ 0, & \text{else}, \end{cases} \tag{10}
$$

for all $u, v \in V$: for Constraints (9b) for node $u \in V$, variable π_u occurs in the first sum for each incoming arc $(u,v) \in \delta^-(u) \setminus C$ with coefficient $-\mu_{(u,v)}$, and in the second sum for each outgoing arc $(u,v) \in \delta^+(u) \setminus C$ with coefficient $\mu_{(u,v)}$, but the sum itself has a negative coefficient. Similarly, each node $v \in V$ adjacent to u occurs exactly once with the conductance of the connecting arc as positive coefficient.

For passive potential-based flow problems such as this one, it is known that the flow, obtained from the solution of the SLE following the potential equation, is unique (Birkhoff and Diaz, 1956). As stated by Pfetsch et al. (2026), we can thus first obtain a primal solution to $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ by solving the SLE, and then check its adherence to the bounds afterward. This covers the soft bounds given by Constraints (9c) that are penalized in the objective function, and the hard bounds given by Constraints (9d) and (9e). We show a proof for the uniqueness here as we give similar proofs later on, and as it is integral to our work—otherwise we might derive a suboptimal solution or falsely conclude that the subproblem is infeasible.

Theorem 2. *Any solution to the primal SLE $D^{\hat{z},C}\pi^C = \hat{b}$ yields the same flow.*

Proof. We select as *reference node* an arbitrary node, say node $w \in V$, such that removing it does not split the graph into disconnected components. Such a node always exists, for example any leaf of a spanning tree of the graph. We then set its potential π_w^C to some arbitrary value $\pi^{\text{ref}} \in \mathbb{R}$. This value can indeed be chosen arbitrarily at first, as, due to the linear potential function, the flows are unaffected by linear shifts in potentials, i.e., $\pi_1 - \pi_2 = (\pi_1 + \varepsilon) - (\pi_2 + \varepsilon)$ for any $\pi_1, \pi_2, \varepsilon \in \mathbb{R}$.

We then remove the entries corresponding to the reference node in the SLE to obtain the remaining coefficient matrix $D^{\hat{z},C,\text{ref}}$, with

$$D_{u,v}^{\hat{z},C,\text{ref}} = D_{u,v}^{\hat{z},C} \qquad \forall u, v \in V \setminus \{w\}.$$

The matrix is identical to the original matrix, but without the row and column corresponding to the reference node. In particular, the remaining diagonal entries remain unchanged.

From standard linear algebra, we know that any SLE $Dx = b$, $D \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, m \in \mathbb{N}$ with $n \in \mathbb{N}$ variables has a unique solution exactly if $\text{rank}(D) = n$. In our remaining SLE, $D^{\hat{z},C,\text{ref}}$ needs to have rank $|V| - 1$. This is the case if

1. it is weakly diagonally dominant, i.e.,
$$\left|D_{u,u}^{\hat{z},C,\text{ref}}\right| \geq \sum_{v \in V \setminus \{u,w\}} \left|D_{u,v}^{\hat{z},C,\text{ref}}\right| \quad \forall u \in V \setminus \{w\}, \tag{11}$$
2. Equation (11) holds with strict inequality for at least one $u \in V \setminus \{w\}$, and
3. $D^{\hat{z},C,\text{ref}}$ is irreducible (Johnson et al., 2023).

All of these hold:

1. This is true due to the matrix structure, and since all conductances are positive. For all $u, v \in V \setminus \{w\}$:
$$D_{u,v}^{\hat{z},C,\text{ref}} = \begin{cases} -\sum_{(u,v) \in \delta(u) \setminus C} \mu_{(u,v)}, & u = v, \\ \mu_{(u,v)}, & (u,v) \in A_L^C, \\ \mu_{(v,u)}, & (v,u) \in A_L^C, \\ 0, & \text{else.} \end{cases}$$
Then for all $u \in V \setminus \{w\}$:
$$\begin{aligned} \left|D_{u,u}^{\hat{z},C,\text{ref}}\right| &= \left| - \sum_{(u,v) \in \delta(u) \setminus C} \mu_{(u,v)} \right| \\ &= \sum_{(u,v) \in \delta(u) \setminus C} \left|\mu_{(u,v)}\right| \\ &\geq \sum_{\substack{(u,v) \in \delta(u) \setminus C \\ \{u,v\} \not\ni w}} \left|\mu_{(u,v)}\right| \\ &= \sum_{v \in V \setminus \{w,u\}} \left|D_{u,v}^{\hat{z},C,\text{ref}}\right|. \end{aligned} \tag{12}$$
2. Since the graph is connected, there must be at least one node $u \in V$ that is connected to the reference node, so $(u,w) \in A_L^C$ or $(w,u) \in A_L^C$. For each such u,
$$(\delta(u) \setminus C) \supsetneq (\delta(u) \setminus C) \setminus \{(u,w),(w,u)\}.$$
Thus, since all conductances are positive, the inequality in Equation (12) holds strictly.

3. The matrix is reducible exactly if there exists a partition U, U' of V such that $D^{\hat{z},C,\text{ref}}_{u,u'} = 0$ for all $u \in U$, $u' \in U'$. But then the graph would be disconnected (or it is only connected at node w, but w was chosen such that removing it does not split the graph).

□

Since any solution to $L^{\hat{z},C,p}$ determines a unique flow and the objective function solely depends on the flow, this flow must coincide with the optimal flow in $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$, provided the solution is feasible. Consider a solution $\pi^{C*} \in \mathbb{R}^V$ to $L^{\hat{z},C,p}$. From Constraints (9c), we have that for any optimal solution,

$$o^{C*}_{(u,v)} = \max\{ -(\pi^{C*}_u - \pi^{C*}_v)\mu_{(u,v)} + \underline{x}_{(u,v)}, \\ (\pi^{C*}_u - \pi^{C*}_v)\mu_{(u,v)} - \tilde{x}_{(u,v)}, 0\}$$

for all $(u,v) \in A^C_L$, i.e., if overloaded, the overloading variables match the violation of the soft bounds. The optimal objective function value to $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ is given by

$$S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^* = \sum_{(u,v) \in A^C_L} F^o o^{C*}_{(u,v)}.$$

Adherence to Constraints (9d) can be immediately determined from the solution. For Constraints (9e), it is based on the choice of reference potential. They are necessarily violated exactly if

$$\begin{aligned} &\max\{0, \max\{\pi^{C*}_u - \bar{\pi}_u \mid u \in V\}\} \\ >&\max\{0, \min\{\pi^{C*}_u - \underline{\pi}_u \mid u \in V\}\}, \text{ or} \\ &\max\{0, \min\{\bar{\pi}_u - \pi^{C*}_u \mid u \in V\}\} \\ <&\max\{0, \max\{\underline{\pi}_u - \pi^{C*}_u \mid u \in V\}\}, \end{aligned}$$

i.e., if potentials can not be shifted enough to satisfy the largest violation of an upper bound without violating any lower bound, or vice versa.

This is sufficient to decide on the type of cut we construct: If $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^* > \hat{\eta}_C$, and the subproblem is feasible, we construct an optimality cut. The subproblem is infeasible if any of Constraints (9d) or (9e) are violated, in which case we construct a feasibility cut. Otherwise, if the subproblem is feasible, and $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^* \leq \hat{\eta}_C$, the master solution is accepted by this subproblem. Note that, unlike when using a simplex algorithm, we do not have a dual solution to the subproblem. This comes next.

5.3 Optimality Cuts

In this subsection, we construct a separate SLE, the solution of which allows us to easily construct an optimality cut. We consider the uncontracted subproblem with switch arcs in this section. We first determine an optimal primal basis based on our optimal primal solution, and then use that basis to construct the "dual" SLE and show that it yields an optimal dual solution.

Optimal primal basis. The primal solution potentially induces multiple optimal primal bases for $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ in augmented form, as the potential values can be additively shifted, as discussed, and the flow on cyclic switch arc components is not necessarily uniquely determined. We deterministically select one of them as follows, and then use it to compute an optimal dual solution. The basis needs to contain $m \in \mathbb{N}$ variables, where m is the number of constraints of $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$, including all variables that are non-zero. This includes the additional slack variables that we need to introduce to convert the inequality constraints to equality constraints. The columns of the selected variables need to be linearly independent. Following the proof of Theorem 2,

we know that this does not hold directly for the potential variables. As in the proof, we thus substitute the potential variable π^C_w of the reference node $w \in V$ by the reference value $\pi^{\text{ref}} \in \mathbb{R}$ and remove the corresponding constraint. With this, there are $m = 4|A^C_S| + 3|V| - 1 + 4|A^C_L|$ constraints.

For the basis B, we select one variable per constraint, always selecting the ones that are non-zero for a solution derived from π^{C*}:

- We select both slack variables of Constraints (5b).
- For Constraints (5c) for arc $(u, v) \in A^C_S$, both slack variables are zero if the arc is on, else both are potentially non-zero. Of all the switch arcs that are on, we choose a maximal subset $A^{\hat{z},C\prime}_S \subseteq A^{\hat{z},C}_S$ such that the underlying undirected subgraph $(V, \{\{u, v\} \mid (u, v) \in A^{\hat{z},C\prime}_S\})$ is a forest (i.e., for each switch arc component, we select only arcs corresponding to an arbitrary undirected spanning tree), and select the corresponding flow variables x^C for the basis. We select the upper bound slack variables of the remaining switch arcs, as well as all lower bound slack variables.
- Up to $|V| - 1$ π^C variables are non-zero as a result of $L^{\hat{z},C,p}$ and Constraints (5d). We select all π^C variables for the basis (excluding that of the reference node).
- For Constraints (5e) for arc $(u, v) \in A^C_L$, either $o^{C*}_{(u,v)} > 0$, in which case at least one of the slack variables is zero, or $o^{C*}_{(u,v)} = 0$. In the latter case, we select both slack variables, in the former the non-zero one and $o^C_{(u,v)}$.
- We select both slack variables of (5f) and (5g), they can be zero or non-zero.

These variables form a valid basis for the Model (5a)–(5j), derived from our optimal solution. We select exactly m variables, one for each constraint. This includes all non-zero variables, as only zero valued variables are discarded. The corresponding columns are also linearly independent: the columns of all slack variables, i.e., those for Constraints (5b), (5f), and (5g); and those we selected for Constraints (5c) and (5e) are unit vectors. This also holds for any selected overload variable $o^C_{(u,v)}, (u, v) \in A_L$. These unit vectors are trivially linearly independent and can be used to remove any non-zero entries in the rows corresponding to the respective constraints. Then, only the coefficients of Constraints (5c) where the arc is switched on and Constraints (5d) remain. Together, they form a $(|A^{\hat{z},C}_S| + |V| - 1) \times (|A^{\hat{z},C}_S| + |V| - 1)$ coefficient matrix of some SLE with the right-hand side according to that of the constraints. Its variables are the potential variables of all nodes except for the reference node, as well as the flow variables of all selected switch arcs. As discussed in Subsection 5.2, the solution of the contracted graph is uniquely determined, and immediately gives us the potentials of all nodes of the original graph. The potential of the nodes that were contracted is equal to that of their representative node. The flows on all selected switch arcs are then uniquely determined following the nodal balance equations (5d), as discussed for the proof of Theorem 1: since the underlying undirected graph of each component is always a spanning tree, the nodal balance equation at each leaf node of such a tree always only has at most one undetermined variable, the flow on an incident switch arc. The same then holds for at least one additional node, until eventually all flow variables are uniquely determined. In total, the SLE has a unique solution, and thus the columns of its coefficient matrix must be linearly independent, as desired. Hence, our selection represents a valid primal basis for our optimal solution.

Dual system of linear equations. We use the basis to construct a system of linear equations to obtain an optimal dual solution for Model (5a)–(5j). For this, we first dualize the primal subproblem containing only the variables in our basis. For brevity, we do not explicitly show the full form of this dual subproblem here, as it is not of further interest. Instead, we discuss

how it can be simplified to obtain the system of linear equations shown in (13a)–(13h), which then is reduced to the SLE (16), the SLE that is solved during the algorithm. We also keep the reference node's variable and constraints for ease of notation, removing them is only needed for uniqueness arguments.

First, all dual constraints of non-slack primal variables can be treated as equality constraints, since we want to obtain an optimal solution: x^C and π^C are unrestricted, so their dual constraints are equality constraints immediately from dualization. The variable o^C is only chosen if positive, so its dual constraint must be an equality constraint following complementary slackness.

Second, we can disregard most of the dual variables, as they are equal to 0 or not relevant to the remaining solution. We keep the dual variables for flow conservation (5d) φ_u^C for all nodes $u \in V$, switch arc potential linking (5c) $\alpha_{(u,v)}^C$ for all active switch arcs $(u,v) \in A_S^{\hat{z},C}$ (including those that are not in $A_S^{\hat{z},C\prime}$), and overloading constraints (5e) for all overloaded line arcs $\underline{\sigma}_{(u,v)}^C$, $(u,v) \in A_L^C$, $x_{(u,v)}^{C*} < \underline{x}_{(u,v)}$, as well as $\bar{\sigma}_{(u,v)}^C$, $(u,v) \in A_L^C$, $x_{(u,v)}^{C*} > \tilde{x}_{(u,v)}$.

All other dual variables are assumed to be equal to zero: from complementary slackness, we know that in an optimal solution, this must be the case for any dual variable where the corresponding slack variable is positive. For Constraints (5b), if the arc is off, the dual variable has objective function coefficient 0 and does not appear in any other constraints (the flow variable is not in the basis), so it can be chosen arbitrarily. If the arc is on, the slack variables are positive by construction. For Constraints (5c), if the arc is off, the slack variables are positive by construction. For the remaining dual variables of Constraints (5e), (5f), and (5g), some constraints can be fulfilled with equality. We assume for now that the corresponding dual variables can always be chosen as zero in an optimal solution, which we show in the proof of Theorem 3.

For ease of notation, we denote for all line arcs $(u,v) \in A_L$

$$\begin{aligned}
\underline{\omega}_{(u,v)}^C &\coloneqq (\varphi_u^C - \varphi_v^C - \underline{\sigma}_{(u,v)}^C)\mu_{(u,v)}, \\
\bar{\omega}_{(u,v)}^C &\coloneqq (\varphi_u^C - \varphi_v^C + \bar{\sigma}_{(u,v)}^C)\mu_{(u,v)}, \\
\omega_{(u,v)}^C &\coloneqq (\varphi_u^C - \varphi_v^C)\mu_{(u,v)},
\end{aligned}$$

and

$$\hat{\omega}_{(u,v)}^C \coloneqq \begin{cases} \underline{\omega}_{(u,v)}^C, & x_{(v,u)}^{C*} < \underline{x}_{(v,u)} \\ \bar{\omega}_{(u,v)}^C, & x_{(v,u)}^{C*} > \tilde{x}_{(v,u)} \\ \omega_{(u,v)}^C, & \text{else.} \end{cases}$$

The resulting system of linear equations can then be seen in (13a)–(13h).

$$\sum_{(v,u)\in\delta^-(u)\setminus C} \hat{\omega}^C_{(v,u)} - \sum_{(u,v)\in\delta^+(u)\setminus C} \hat{\omega}^C_{(u,v)} + \sum_{(v,u)\in\delta^-(u)\cap A_S^{\hat{z},C}} \bar{\alpha}^C_{(v,u)} - \sum_{(u,v)\in\delta^+(u)\cap A_S^{\hat{z},C}} \bar{\alpha}^C_{(u,v)} = 0 \qquad [\pi^C] \qquad \forall u \in V \tag{13a}$$

$$-\underline{\sigma}^C_{(u,v)} = F^o \qquad [o^C] \qquad \forall (u,v) \in A_L^C, x^{C*}_{(u,v)} < \underline{x}_{(u,v)} \tag{13b}$$

$$-\bar{\sigma}^C_{(u,v)} = F^o \qquad [o^C] \qquad \forall (u,v) \in A_L^C, x^{C*}_{(u,v)} > \tilde{x}_{(u,v)} \tag{13c}$$

$$-\varphi^C_u + \varphi^C_v = 0 \qquad [x^C] \qquad \forall (u,v) \in A_S^{\hat{z},C\prime} \tag{13d}$$

$$\bar{\alpha}^C_{(u,v)} \in \mathbb{R} \qquad \forall (u,v) \in A_S^{\hat{z},C} \tag{13e}$$

$$\underline{\sigma}^C_{(u,v)} \in \mathbb{R} \qquad \forall (u,v) \in A_L^C, x^{C*}_{(u,v)} < \underline{x}_{(u,v)} \tag{13f}$$

$$\bar{\sigma}^C_{(u,v)} \in \mathbb{R} \qquad \forall (u,v) \in A_L^C, x^{C*}_{(u,v)} > \tilde{x}_{(u,v)} \tag{13g}$$

$$\varphi^C_u \in \mathbb{R} \qquad \forall u \in V \tag{13h}$$

This SLE has $|A_S^{\hat{z},C}| - |A_S^{\hat{z},C\prime}|$ more variables than equations, so we can not always immediately obtain a unique solution. Similarly to the last section, we avoid this by first calculating a solution without switch arcs. We first describe the procedure for this and then show that it always provides an optimal solution.

From Constraints (13b) and (13c), the values of $\underline{\sigma}^C_{(u,v)}$ and $\bar{\sigma}^C_{(u,v)}$ are directly determined for all line arcs:

$$\underline{\sigma}^{C*}_{(u,v)} = \begin{cases} 0, & x^{C*}_{(u,v)} \geq \underline{x}_{(u,v)} \\ -F^o, & x^{C*}_{(u,v)} < \underline{x}_{(u,v)}, \end{cases}$$

$$\bar{\sigma}^{C*}_{(u,v)} = \begin{cases} 0, & x^{C*}_{(u,v)} \leq \tilde{x}_{(u,v)} \\ -F^o, & x^{C*}_{(u,v)} > \tilde{x}_{(u,v)} \end{cases}$$

for all $(u,v) \in A_L^C$. We unify this as

$$\sigma^{C*}_{(u,v)} = \begin{cases} 0, & \underline{x}_{(u,v)} \leq x^{C*}_{(u,v)} \leq \tilde{x}_{(u,v)} \\ -F^o, & x^{C*}_{(u,v)} < \underline{x}_{(u,v)} \\ F^o, & x^{C*}_{(u,v)} > \tilde{x}_{(u,v)} \end{cases}$$

for all $(u,v) \in A_L^C$. Note that this term is fixed based on the primal solution. Observe that the remaining dual problem is almost a semi-passive potential-based flow with identity potential functions itself—potentials φ^C, conductances μ, switch flows $\bar{\alpha}^C$, and demands σ^{C*} with μ. However, the objective function does not fit with our definition of a potential-based flow problem.

Nevertheless, from Theorem 1, we know that the contraction procedure preserves "flows" for both line $\omega^C = (\varphi^C_u - \varphi^C_v)\mu_{(u,v)}$, $(u,v) \in A_L^C$, and switch arcs $\bar{\alpha}^C_{(u,v)}$, $(u,v) \in A_S^{\hat{z},C}$. As before, let $W \subseteq V$ be a set of all the nodes that are contracted to a common node, i.e., the nodes of a connected component in $(V, A_S^{\hat{z},C})$, and let $\mathcal{W}^{\hat{z},C}$ denote the set of all such sets. Summing up Equation (13a) for all $u \in W$ allows us to remove the $\bar{\alpha}^C$ variables, it yields

$$\sum_{u\in W} \left(\sum_{(v,u)\in\delta^-(u)\setminus C} \omega^C_{(v,u)} - \sigma^{C*}_{(v,u)}\mu_{(v,u)} - \sum_{(u,v)\in\delta^+(u)\setminus C} \omega^C_{(u,v)} - \sigma^{C*}_{(u,v)}\mu_{(u,v)} \right) = 0$$

(each switch arc in $A_S^{\hat{z},C} \cap (W \times W)$, i.e., that is part of the component, occurs exactly once as incoming and outgoing arc each). We can move the fixed terms containing σ^{C*} to the right-hand side

$$\sum_{u \in W} \left(\sum_{(v,u) \in \delta^-(u) \setminus C} \omega^C_{(v,u)} - \sum_{(u,v) \in \delta^+(u) \setminus C} \omega^C_{(u,v)} \right)$$
$$= \sum_{u \in W} \left(\sum_{(v,u) \in \delta^-(u) \setminus C} \sigma^{C*}_{(v,u)} \mu_{(v,u)} - \sum_{(u,v) \in \delta^+(u) \setminus C} \sigma^{C*}_{(u,v)} \mu_{(u,v)} \right).$$

We denote the right-hand side term as

$$\sigma^{C*}_W := \sum_{u \in W} \left(\sum_{(v,u) \in \delta^-(u) \setminus C} \sigma^{C*}_{(v,u)} \mu_{(v,u)} - \sum_{(u,v) \in \delta^+(u) \setminus C} \sigma^{C*}_{(u,v)} \mu_{(u,v)} \right). \tag{14}$$

This yields the SLE

$$\sum_{u \in W} \left(\sum_{(v,u) \in \delta^-(u) \setminus C} \omega^C_{(v,u)} - \sum_{(u,v) \in \delta^+(u) \setminus C} \omega^C_{(u,v)} \right) = \sigma^{C*}_W \quad \forall W \in \mathcal{W}^{\hat{z},C}. \tag{15}$$

This SLE may still have more variables than equations. We mitigate this by merging the φ^C variables of each component onto its representative node in the contracted graph. This preserves the optimality of our computed dual solution, as discussed for Theorem 3. We denote the representative node in the contracted graph of component $W \in \mathcal{W}^{\hat{z},C}$ by $u(W)$. The resulting dual SLE, denoted $L^{\hat{z},C,d}$ is

$$\sum_{(u(U),u(W)) \in \delta^-(u(W))} \omega^C_{(u(U),u(W))} - \sum_{(u(W),u(U)) \in \delta^+(u(W))} \omega^C_{(u(W),u(U))} = \sigma^{C*}_W \quad \forall W \in \mathcal{W}^{\hat{z},C}. \tag{16}$$

The sets δ^- and δ^+ refer to the contracted graph here. Observe that, for any graph, only the right-hand side changes between dual SLE $L^{\hat{z},C,d}$ for the optimality cut and the primal SLE $L^{\hat{z},C,p}$ (cf. Equation (10)).

For switch arcs, it remains to calculate values for the $\bar{\alpha}^C$ variables following Equation (13a). This is a separate flow problem for each component, with nodal surpluses derived from the dual line flows given by φ^{C*} and μ. As discussed for the primal switch arc flows, we solve it by constructing a spanning tree for each of the components of a given substation, and then moving the surpluses "up" starting from the leaves. If an (undirected) component contains cycles, the solution values for these dual variables are not unique. However, their objective function coefficient is 0, so they do not directly influence the objective function value.

It remains to show that the computed dual solution is optimal.

Theorem 3. *Any solution to the SLE $L^{\hat{z},C,d}$, extended by the tree-based recovery of the dual variables $\bar{\alpha}^C$ and setting the dual variables corresponding to the remaining bounds to zero, yields an optimal dual solution to the original $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$.*

Proof. We need to show that the system produces a unique solution for the dual "flows" ω^C, and that reference value, contraction procedure, and disregarding part of the slack variables do not lead to a suboptimal solution.

From Theorem 2, we know that the SLE yields a unique flow $\omega^{C*} \in A_L^{\hat{z},C}$ on the contracted graph, with unique values for $\varphi^{C*} \in \mathcal{W}^{\hat{z},C}$ given a chosen reference value and node. Following Constraints (13d), it holds that $\varphi^{C*}_u = \varphi^{C*}_{u(W)}$ for all $W \in \mathcal{W}^{\hat{z},C}, u \in W$, as required by the contraction procedure.

However, unlike the potential variables, the dual variable for node $u \in V$, including the reference node, appears in the objective function of the dual subproblem with potentially non-zero coefficient $\hat{b}_u \in \mathbb{R}$. We must hence show that the choice of reference value has no impact on the objective function value. We required that the demand in the network is balanced, i.e., that its sum over all nodes is equal to 0. Let $\varepsilon \in \mathbb{R}$. Then,

$$\begin{aligned}\sum_{u \in V} \varphi_u^{C*} \cdot \hat{b}_u &= \sum_{u \in V} \varphi_u^{C*} \cdot \hat{b}_u + \varepsilon \cdot 0 \\ &= \sum_{u \in V} \varphi_u^{C*} \cdot \hat{b}_u + \varepsilon \cdot \sum_{u \in V} \hat{b}_u \\ &= \sum_{u \in V} (\varphi_u^{C*} + \varepsilon) \cdot \hat{b}_u,\end{aligned}$$

as desired.

Lastly, assume, for a contradiction, that it is invalid to disregard the dual variables of Constraints (5e), (5f), and (5g). That is, there is some constraint where the flow or potential is exactly equal to a (soft or hard) bound, and the dual variable of that constraint cannot be equal to zero in any optimal solution. Consider a relaxation that differs from the original LP (5a)–(5j) only in loosening exactly this bound by some $\varepsilon \in \mathbb{R}_{>0}$ (i.e., increasing the upper bound or decreasing the lower bound, as applicable). As discussed, the potentials of the primal problem are uniquely determined independently of this bound; thus, relaxing it impacts neither the feasibility nor the objective function value of the primal solution. Since the bound is now no longer active, the dual variable vanishes by complementary slackness. This dual solution is feasible for the dual of the original LP, because changing a right-hand side of the primal does not affect the dual feasible region. Since the changed dual variable is at value zero, its modified objective function coefficient (the loosened bound) does not affect the dual objective function value. Therefore, by weak duality, our dual solution (of the same objective function value as the primal) must also be optimal. This contradicts our assumption that there is no such solution. We conclude that we can set all remaining dual variables corresponding to Constraints (5e), (5f), and (5g) to zero. □

5.4 Feasibility Cuts

In this section, we adapt $L^{\hat{z},C,d}$ for the case that $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ is infeasible, and we thus need to construct a feasibility cut. To do so, we need to find a dual direction of unboundedness, i.e., a direction with positive objective function value, of which we can add any positive multiple to a feasible dual point without violating any constraints. There are two cases in which the subproblem is infeasible: an arc flow (resulting from the bound-free primal subproblem) violates its hard bounds in Constraints (9d), or a node potential violates its hard bounds in Constraints (9e). Both cases can occur simultaneously, also for multiple arcs or nodes. We add exactly one cut, prioritizing flows over potentials.

Flow bound violation. If a flow violates some hard bound, we know that there must be a line arc $(w, u) \in A_L^C$ between nodes $w, u \in V$ and components $W', U' \in \mathcal{W}^{\hat{z},C}$, $w \in W', u \in U'$, with $(\pi_w^C - \pi_u^C)\mu_{(w,u)} < \underline{x}_{(w,u)}$ or $(\pi_w^C - \pi_u^C)\mu_{(w,u)} > \bar{x}_{(w,u)}$. We discuss only the latter case, the former is analogous. The dual variable of the violated constraint for the contracted graph is $\bar{\gamma}^C_{(u(W'),u(U'))}$ with objective function coefficient $\bar{x}_{(u(W'),u(U'))}/\mu_{(u(W'),u(U'))}$. Increasing the solution value of $\bar{\gamma}^C_{(u(W'),u(U'))}$ increases the objective function value of the dual solution. However, we need to do so in a way that does not impact the feasibility of the solution. The variable occurs only in

Constraints (13a), for both endpoints of the arc. We modify the SLE in Equation (15) to

$$\sum_{(u(W),u(U))\in\delta^-(u(W))} \omega^C_{(u(W),u(U))} - \sum_{(u(U),u(W))\in\delta^+(u(W))} \omega^C_{(u(U),u(W))} = 0 \;\; \forall W \in \mathcal{W}^{\hat{z},C} \setminus \{W', U'\},$$
$$\bar{\gamma}^C_{(u(W'),u(U'))} + \sum_{(u(W'),u(U))\in\delta^-(u(W'))} \omega^C_{(u(W'),u(U))} - \sum_{(u(U),u(W'))\in\delta^+(u(W'))} \omega^C_{(u(U),u(W'))} = 0,$$
$$-\bar{\gamma}^C_{(u(W'),u(U'))} + \sum_{(u(U'),u(U))\in\delta^-(u(U'))} \omega^C_{(u(U'),u(U))} - \sum_{(u(U),u(U'))\in\delta^+(u(U'))} \omega^C_{(u(U),u(U'))} = 0. \quad (17)$$

Any solution that satisfies this SLE can be added onto a dual feasible point without leaving the dual feasible region. It thus is a feasible direction for the dual subproblem. The direction also needs to have a positive objective function value (here, e.g., 1), so we add this equation to (17)

$$\sum_{W\in\mathcal{W}^{\hat{z},C}} \hat{b}_{u(W)} \cdot \varphi^C_{u(W)} + \sum_{(u(W),u(U))\in A_L^{\hat{z},C}} \left((\underline{\sigma}^{C*}_{(u(W),u(U))} + \bar{\sigma}^{C*}_{(u(W),u(U))}) \cdot \tilde{x}_{(u(W),u(U))} \right)$$
$$+ \frac{\bar{x}_{(u(W'),u(U'))}}{\mu_{(u(W'),u(U'))}} \cdot \bar{\gamma}^C_{(u(W'),u(U'))} = 1.$$

Since the primal is infeasible and the dual is feasible, a dual direction of unboundedness must exist. As our SLE is constructed to find exactly such a direction, it is guaranteed to be solvable.

Potential bound violation. If Constraint (9e) is violated, let $\tilde{w} \in V$, part of component $W' \in \mathcal{W}^{\hat{z},C}$ be a node with the largest violation (or closest to violation) of its upper bound, i.e.,

$$\tilde{w} \in \arg\max\{\pi^{C*}_u - \bar{\pi}_u \mid u \in V\}.$$

We do not need to consider the lower bound case, as we can always shift the potentials such that only upper bounds are violated. Analogously to the previous discussion, we modify the SLE in Equation (15) to

$$\sum_{(u(W),u(U))\in\delta^-(u(W))} \omega^C_{(u(W),u(U))} - \sum_{(u(U),u(W))\in\delta^+(u(W))} \omega^C_{(u(U),u(W))} = 0 \;\; \forall W \in \mathcal{W}^{\hat{z},C} \setminus \{W'\},$$
$$\bar{\xi}^C_{u(W')} + \sum_{(u(W'),u(U))\in\delta^-(u(W'))} \omega^C_{(u(W'),u(U))} - \sum_{(u(U),u(W'))\in\delta^+(u(W'))} \omega^C_{(u(U),u(W'))} = 0, \quad (18)$$

and add the objective function via equation

$$\sum_{W\in\mathcal{W}^{\hat{z},C}} \hat{b}_{u(W)} \cdot \varphi^C_{u(W)} + \sum_{(u(W),u(U))\in A_L^{\hat{z},C}} \left((\underline{\sigma}^{C*}_{(u(W),u(U))} + \bar{\sigma}^{C*}_{(u(W),u(U))}) \cdot \tilde{x}_{(u(W),u(U))} \right)$$
$$+ \bar{\pi}_{u(W')} \cdot \bar{\xi}^C_{u(W')} = 1.$$

As for optimality cuts, solving the respective SLEs yields values for all dual variables, and the desired cut.

6 Factorization Scheme & Complexity

In this section, we discuss the implications of our proposed subproblem solving regarding the overall branch-and-Benders-cut algorithm. We first discuss how to leverage the similarity of the SLEs within and between subproblems. Afterward, we analyze the theoretical complexity of the resulting algorithm.

Subproblem similarity. During the branch-and-Benders-cut algorithm, the set of active arcs changes between master solutions and between contingencies and their corresponding subproblems. Each subproblem $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ is solved with a primal and possibly a dual SLE. Each SLE $Dx = b$ consists of a subproblem-dependent matrix $D \in \mathbb{R}^{n \times n}$ and a right-hand side $b \in \mathbb{R}^n$ representing currently active arcs (cf. Constraints (9b) and Equation (15)), where $n = |\mathcal{W}^{\hat{z},C}|$ for primal and optimality cut subproblems, and $n = |\mathcal{W}^{\hat{z},C}| + 1$ in case of a feasibility cut. We solve these linear equation systems using an LU-factorization: first, D is decomposed into a lower triangular matrix L and an upper triangular matrix U, with $L, U \in \mathbb{R}^{|\mathcal{W}^{\hat{z},C}| \times |\mathcal{W}^{\hat{z},C}|}$, such that $LU = D$. Afterward, we solve $Ly = b$ for y and then $Ux = y$ for x to obtain a solution to the original SLE. Computing the factorization from scratch has as running time complexity of $\mathcal{O}(|\mathcal{W}^{\hat{z},C}|^3)$ (Golub and Van Loan, 2013). This dominates the running time complexity when the matrix is triangular, e.g., for solving an SLE using an existing factorization, $\mathcal{O}(|\mathcal{W}^{\hat{z},C}|^2)$ (Golub and Van Loan, 2013). As discussed below, we can convert the factorization of a matrix into that of a sufficiently similar one in $\mathcal{O}(|\mathcal{W}^{\hat{z},C}|^2)$ as well. We thus want to avoid complete factorizations by exploiting the similarity between the matrices of the subproblems that we solve.

The similarities occur between Benders iterations, between contingencies, and between the primal and dual SLEs of the same subproblem. As observed before, the coefficient matrices of the SLEs for primal $L^{\hat{z},C,p}$ and optimality cuts $L^{\hat{z},C,d}$ are equal. The similarity of contingencies depends on the contingency size, in our evaluation instances, for example, it is small and independent of the network size. The switching limit parameter Z restricts the number of differing switch arcs between master solutions.

Updating individual arcs. Subproblems within or between Benders iterations differ in the underlying contracted graph. Switch arc states determine the components of the graph, line arc states determine how components are connected. In the simplest case, two subproblems differ in a single line arc that is operational or failed due to a contingency. A matrix $D^{-(u',v')}$ with arc $(u', v') \in A_L$ disconnected differs from the same matrix $D^{+(u',v')}$ with that arc connected only slightly (cf. Equation (10)):

$$D^{-(u',v')} + D^{(u',v')} = D^{+(u',v')},$$

where

$$D^{(u',v')}_{(u,v)} = \begin{cases} -\mu_{(u',v')}, & \{u,v\} \subsetneq \{u',v'\} \\ \mu_{(u',v')}, & \{u,v\} = \{u',v'\} \\ 0, & \text{else} \end{cases}$$

for all $u, v \in V$. Let

$$q^{(u',v')}_u = \begin{cases} \mu_{(u',v')}, & u = u' \\ -\mu_{(u',v')}, & u = v' \\ 0, & \text{else,} \end{cases}$$

and

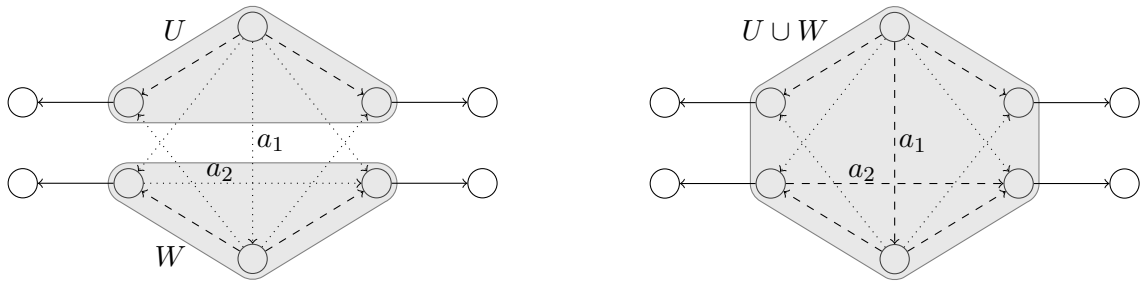
$$r^{(u',v')}_u = \begin{cases} -1, & u = u' \\ 1, & u = v' \\ 0, & \text{else,} \end{cases}$$

for all $u \in V$. Then,

$$D^{(u',v')} = q^{(u',v')} r^{(u',v')\top}.$$

Given an LU-factorization $L^{-(u',v')}U^{-(u',v')} = D^{-(u',v')}$, we can thus perform a rank-1 update to arrive at an LU-factorization $L^{+(u',v')}U^{+(u',v')} = D^{+(u',v')}$. This update can be performed in $\mathcal{O}(|V|^2)$ (Stange et al., 2007).

Figure 3: Component reconfiguration.



(a) U and V split. (b) U and V merged.

Note. Reconfiguration line arcs are represented by solid lines, active and inactive switch arcs by dashed and dotted lines, respectively. Activating arc a_1 merges or splits components U and W (case 1 and 2). Changing the state of arc a_2 has no immediate effect, as its incident nodes are part of the same component regardless (case 3).

Differing switch arc states can lead to a change in the components represented in the matrix. Recall that the matrix considers only active components, i.e., components that contain at least one node with non-zero demand. There are three distinct outcomes regarding switch arc components after changing the state of a single arc:

- After an arc is activated, two previously separated components are now connected, resulting in one merged component. If at least one of them is active, we connect it (in the matrix, via rank-1 updates) to the line arcs of the other component. If the other component is also active, we disconnect all its line arcs and then delete the corresponding row and column in the matrix.

- The reverse: after deactivating an arc, a component is split into two now distinct components. If at least one of them is active, we disconnect all line arcs of the other component, the matrix entries now represent this first component. If the second component is also active, we add it as a new component to the matrix by adding a row and column to represent the new component in the matrix and then connecting its line arcs.

- No change in active components. If the arc is activated, this is because the incident nodes are already part of the same component, or because one of the incident nodes was previously not part of an active component. If the arc is deactivated, this is because the incident nodes are connected via other active switch arcs, or because nodes are removed from the component, but do not form an active component. The matrices stay the same, so we do not need to update the factorization.

Refer to Figure 3 for an example.

A special case occurs if the switch arc is paired with exactly one line arc, i.e., there are nodes $u, v, w \in V$, $(u, v) \in A_L$, $(v, w) \in A_S^C$, $\delta(v) = \{(u, v), (v, w)\}$. This can be seen as a combined line and switch arc (u, w), if switched on it has the properties of (u, v) and if switched off, the potentials u, w are disconnected. The potential at v is not explicitly relevant, it is equal to that of u if the arc is on and to that of w if it is off. We omit its row and column in the SLE matrices, and treat switching the arc analogously to a contingency including the line arc.

Factorization update algorithm. Multiple differing arcs can be resolved by an iterative chain of such updates. Algorithm 1 is called after a valid integer feasible solution is found during the branch-and-Benders-cut algorithm. It describes how to perform one subproblem iteration of the Benders algorithm, i.e., solve a subproblem for each contingency and return optimality or feasibility cuts if applicable. We depart from the factorization of the previous master topology $\hat{z}' \in \{0, 1\}^{|A_S|}$, $L^{\hat{z}'}U^{\hat{z}'} = D^{\hat{z}'}$. For the first call, this is the "base" factorization

Algorithm 1 Subproblem Solving with Factorization Updates.

input network and contingency data $(G, \mu, \psi, \mathcal{C}, H)$; previous factorization $L^{\hat{z}'}, U^{\hat{z}'}$; master solution $\hat{z}$, $\hat{b}$, $\hat{\eta}$; previous master topology $\hat{z}'$
$C' \leftarrow \emptyset$
$L^{\hat{z},C'}, U^{\hat{z},C'} \leftarrow$ rank-1 updates to $L^{\hat{z}'}, U^{\hat{z}'}$ according to $\hat{z}$

- disconnect all $(u,v) \in A_L^{\hat{z}',\emptyset} \setminus A_L^{\hat{z},\emptyset}$
- update component representation for all $(u,v) \in A_S$, $\hat{z}_{(u,v)} \neq \hat{z}'_{(u,v)}$
- reconnect all $(u,v) \in A_L^{\hat{z},\emptyset} \setminus A_L^{\hat{z}',\emptyset}$

for all $C \in \mathcal{C}$ **do**
    $L^{\hat{z},C}, U^{\hat{z},C} \leftarrow$ rank-1 updates to $L^{\hat{z},C'}, U^{\hat{z},C'}$

- disconnect all $(u,v) \in C \cap A_L^{\hat{z},C'}$
- update component representation for all $(u,v) \in ((C \cup C') \setminus (C \cap C')) \setminus A_S^{\bar{z}}$
- reconnect all $(u,v) \in C' \cap A_L^{\hat{z},C}$

    solve primal SLE with $L^{\hat{z},C}, U^{\hat{z},C}$ (cf. Subsection 5.2)
    if $S_C(\hat{b}, \hat{z}, \hat{\eta}_C)$ infeasible **then**
        feasibility cut: expand $L^{\hat{z},C}, U^{\hat{z},C}$ and solve feasibility cut SLE (cf. Subsection 5.4)

- for $\varphi^C, \bar{\gamma}^C_{(\tilde{u},\tilde{v})}$, where $(\tilde{u},\tilde{v}) \in A_L^C$ is a violating arc, or
- for $\varphi^C, \bar{\xi}^C_{\tilde{u}}$, where $\tilde{u} \in V$ is a violating node

    else
        $\eta_C \leftarrow S_C(\hat{b}, \hat{z}, \hat{\eta}_C)^*$
        if $\eta_C > \hat{\eta}_C$ **then**
            optimality cut: solve optimality cut SLE with $L^{\hat{z},C}, U^{\hat{z},C}$ (cf. Subsection 5.3)
        end if
    end if
    $C' \leftarrow C$
end for
$L^{\hat{z},C'}, U^{\hat{z},C'} \leftarrow$ rank-1 updates to $L^{\hat{z},C'}, U^{\hat{z},C'}$

- update component representation for all $(u,v) \in C \cap A_S^{\hat{z},C}$
- reconnect all line arcs $(u,v) \in C'$

return all found cuts

$LU = D$ that corresponds to the equation system with all arcs switched according to the default configuration of the network, i.e., $\hat{z}'_{(u,v)} = i$ for all $(u,v) \in A_S^i$, $i \in \{0,1\}$.

From this initial factorization, we want to compute the factorization $L^{\hat{z}}U^{\hat{z}} = D^{\hat{z}}$ of the current master topology $\hat{z} \in \{0,1\}^{A_S}$. For each arc $(u,v) \in A_S$, $\hat{z}_{(u,v)} \neq \hat{z}'_{(u,v)}$, that differs between the master solutions, the underlying contracted graph can change. As discussed before, this can lead to a change in the components represented in the matrix, which is resolved by sequentially activating or deactivating the line arcs incident to either component.

After updating the factorization to the master topology, a subproblem is solved for each contingency. To go from one contingency $C' \in \mathcal{C}$ to the next $C \in \mathcal{C}$, we need to disconnect each line arc in $C \cap A_L^{\hat{z},C'}$, and connect each line arc in $C' \cap A_L^{\hat{z},C}$. Each of these requires a rank-1 update, if the incident components are active. Switch arcs $(u,v) \in ((C \cup C') \setminus (C \cap C')) \setminus A_S^{\bar{z}}$ activated or deactivated between the contingencies can lead to a change in matrix components, handled as above.

We then solve the primal SLE, as discussed in Subsection 5.2. If the subproblem is feasible and its objective function value η_C is smaller than or equal to the "estimated" objective function value $\hat{\eta}$, no cut needs to be added, we proceed with the next subproblem.

If it is feasible and its objective function value is larger than $\hat{\eta}$, we compute an optimality cut. As detailed in Subsection 5.3, the matrix of the corresponding SLE is the same as for the primal subproblem, only the right hand side changes.

If it is infeasible, let $(\tilde{u},\tilde{v}) \in A_L^{\hat{z},C}$ be an arc violating the hard flow bounds, if one exists. As discussed in Subsection 5.4, we need to solve the SLE

$$\begin{pmatrix} D^{\hat{z},C} & s^{(\tilde{u},\tilde{v})} \\ \hat{b}^\top & \bar{x}_{(\tilde{u},\tilde{v})} \end{pmatrix} \begin{pmatrix} \varphi^C \\ \bar{\gamma}^C_{(\tilde{u},\tilde{v})} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

where $D^{\hat{z},C}$ is the matrix of the primal SLE, $\hat{b}$ is the dual objective function coefficient vector for φ^C, and

$$s^{(\tilde{u},\tilde{v})}_{u(W)} = \begin{cases} 1, & u(W) = \tilde{u} \\ -1, & u(W) = \tilde{v} \\ 0, & \text{else,} \end{cases} \quad \forall W \in \mathcal{W}^{\hat{z},C}$$

corresponds to the coefficients of $\bar{\gamma}^C_{(\tilde{u},\tilde{v})}$ in Equations (17). We already have the factorization for $D^{\hat{z},C} = L^{\hat{z},C}U^{\hat{z},C}$ and want to expand it to cover the remaining entries of the matrix. It must hold that

$$\begin{aligned} &\begin{pmatrix} D^{\hat{z},C} & s^{(\tilde{u},\tilde{v})} \\ \hat{b}^\top & \bar{x}_{(\tilde{u},\tilde{v})} \end{pmatrix} \\ &= \begin{pmatrix} L^{\hat{z},C} & 0 \\ \ell^{\hat{z},C\top} & 1 \end{pmatrix} \begin{pmatrix} U^{\hat{z},C} & u^{\hat{z},C} \\ 0^\top & v^{\hat{z},C} \end{pmatrix} \\ &= \begin{pmatrix} L^{\hat{z},C}U^{\hat{z},C} & L^{\hat{z},C}u^{\hat{z},C} \\ \ell^{\hat{z},C\top}U^{\hat{z},C} & \ell^{\hat{z},C\top}u^{\hat{z},C} + v^{\hat{z},C} \end{pmatrix}. \end{aligned}$$

We need to compute $\ell^{\hat{z},C}, u^{\hat{z},C} \in \mathbb{R}^{\mathcal{W}^{\hat{z},C}}$ and $v^{\hat{z},C} \in \mathbb{R}$. For this, we can solve the SLEs

$$U^{\hat{z},C\top}\ell^{\hat{z},C} = \hat{b}$$

and

$$L^{\hat{z},C}u^{\hat{z},C} = s^{(\tilde{u},\tilde{v})},$$

and then solve

$$v^{\hat{z},C} = \bar{x}_{(\tilde{u},\tilde{v})} - \ell^{\hat{z},C\top}u^{\hat{z},C}.$$

Note that $U^{\hat{z},C\top}$ and $L^{\hat{z},C}$ are already triangular, and these SLEs are thus solvable in $\mathcal{O}(|\mathcal{W}^{\hat{z},C}|^2)$.

If no violating arc exists, let instead $\tilde{u} \in V$ be a node with the highest or closest to violation, as discussed in Subsection 5.4. The SLE we need to solve is

$$\begin{pmatrix} D^{\hat{z},C} & s^{\tilde{u}} \\ \hat{b}^\top & \bar{\pi}_{\tilde{u}} \end{pmatrix} \begin{pmatrix} \varphi^C \\ \bar{\xi}^C_{\tilde{u}} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

where

$$s^{\tilde{u}}_{u(W)} = \begin{cases} 1, & u(W) = \tilde{u} \\ 0, & \text{else,} \end{cases} \quad \forall W \in \mathcal{W}^{\hat{z},C}.$$

We proceed analogously to update the factorization.

Finally, after all subproblems have been solved, we revert the changes to $L^{\hat{z},C}$ and $U^{\hat{z},C}$ caused by the contingencies, and return all cuts that were found.

Theoretical complexity. As a last point, we discuss the theoretical complexity of the proposed methods, as a complement to the empirical results in the next section. We propose a method with which the subproblems can be solved with systems of linear equation, and combine it with a factorization update scheme. We compare this to solving the subproblems as LPs with a simplex algorithm. The methods differ only in their way to obtain primal and dual solutions.

While a simplex algorithm can require up to 2^{n-1} iterations, in practice it empirically uses only $\mathcal{O}(m)$, with a small constant (Bertsimas and Tsitsiklis, 1997), where $n = |V| + |A^C_S|$ is the number of variables and $m = 4|A^C_S| + 3|V| + 4|A^C_L|$ is the number of constraints. Each of these iterations requires solving up to three SLEs of size m. For solving the SLEs, a factorization update scheme similar to ours can be used, which reduces their overall complexity to $\mathcal{O}(m^2)$ per simplex iteration. We solve $|\mathcal{C}|$ different subproblems in one Benders iteration. In total, we arrive at $\mathcal{O}(|\mathcal{C}| \cdot m \cdot m^2) = \mathcal{O}(|\mathcal{C}| \cdot (|V| + |A|)^3) = \mathcal{O}(|\mathcal{C}| \cdot |A|^3)$. When the number of contingencies is restrained to the number of line arcs by some constant factor (e.g., one contingency per line arc), this reduces to $\mathcal{O}(|A|^4)$.

To solve each subproblem with our method, we need to solve at most two SLEs. Additionally, these SLEs contain only at most one variable per node. Solving all of them without factorization updates thus has a total complexity of $\mathcal{O}(|\mathcal{C}| \cdot 2 \cdot |V|^3) \subseteq \mathcal{O}(|\mathcal{C}| \cdot |A|^3)$, or $\mathcal{O}(|A|^4)$ if the number of contingencies is restrained by the number of arcs. This can be reduced when using factorization updates: Let $\hat{C} \in \mathbb{Z}$ be the size of the largest contingency. Computing the initial factorization, or updating from a previous master solution, can be done in $\mathcal{O}(|V|^3)$. For each subproblem, we then need to revert up to $\hat{C}$ updates from the previous subproblem, and apply equally many of the current subproblem. Afterward, we solve the primal and dual SLE via forward and backward substitution. For feasibility cuts, we require another forward and backward substitution to update the factorization. We arrive at a total complexity of $\mathcal{O}(|V|^3 + |\mathcal{C}| \cdot (2 \cdot \hat{C} \cdot |V|^2 + 4 \cdot |V|^2)) = \mathcal{O}(|V|^3 + |\mathcal{C}| \cdot \hat{C} \cdot |V|^2) \subseteq \mathcal{O}(|A|^3 + |\mathcal{C}| \cdot \hat{C} \cdot |A|^2)$. If the number of contingencies is restrained by the number of arcs, and $\hat{C}$ is independent of the network size, this reduces to $\mathcal{O}(|A|^3) \subset \mathcal{O}(|A|^4)$. This is, e.g., the case for the so-called $N-1$ criterion, where we have at most one contingency per line arc, and each contingency contains at most one line arc. Note that our coarse analysis is overly conservative for our approach as in sparse networks where the number of arcs is linear in the number of nodes, whereas in general $|A| \in \mathcal{O}(|V|^2)$.

It should be remarked that our method can be seen as a specialized simplex algorithm, and many standard acceleration techniques apply. These are out of the scope of this work. We refer to, for example, Vanderbei (1998) and Bertsimas and Tsitsiklis (1997) for details.

7 Evaluation

In this section, we evaluate the performance of the proposed subproblem algorithms: "classical" with Gurobi `GRB`, with the systems of linear equations solver `SLE`, introduced in Sections 5.2–5.4, as well as the systems of linear equations solver with factorization updates `SLE-U` (Section 6).

All algorithms are implemented using Julia 1.12.6 (Bezanson et al., 2017). We use PowerModels 0.21.85 (Coffrin et al., 2018) with JuMP 1.30.0 (Lubin et al., 2023) to model the Benders master, as well as the Benders subproblem for `GRB`. They are solved using Gurobi 13.0.0 (Gurobi Optimization, LLC, 2026), with one model object modified between subproblems. The Benders functionality is implemented via Gurobi's callbacks. For `SLE` and `SLE-U`, subproblems are solved with the Julia package LinearSolve 3.75.0 (Kimmerer et al., 2026). The factorization update functionality for `SLE-U` is newly implemented, based on the work by Stange et al. (2007). To avoid numerical issues, factorizations are recomputed in regular intervals, among other things whenever the dimension of the coefficient matrix changes.

Evaluation setting. We evaluate the methods on instances of the DC security-constrained optimal transmission switching problem (SCOTS). For these instances, the graph represents a DC power network, with potentials and flows representing *voltage angles* and *power flows*, respectively. The potential function is affine, with $r_{(u,v)} = 1$ on all arcs $(u, v) \in A_L$. The constant $s_{(u,v)} \in \mathbb{R}$ can model, e.g., transformers. Most line arcs in the instances are switchable themselves, which can be modelled as the aforementioned pairing of two correspondingly parametrized line arcs and one switch arc in the potential-based flow setting. Switch arcs can also represent *substations*, i.e., components that are located so close to each other that their electrical properties can be disregarded. This corresponds to the switch arc components discussed in the previous sections. The set of contingencies aligns with the $N-1$-criterion, i.e., one contingency per line arc, as long as it does not split the network. As discussed, any affine potential based flow instance can be transformed into one with an identity function as potential equation, and thus also a SCOTS instance.

Benchmark dataset. We use a custom benchmark set based on the PGLib-OPF (Babaeinejadsarookolaee et al., 2019) set, instances of which are commonly used for testing power flow applications (Fisher et al., 2008; Hedman et al., 2009; Heidarifar and Ghasemi, 2016). The original benchmark set itself is insufficient since it contains too few instances that are solvable within sufficient time with our computational setup. Additionally, many of the instances are infeasible for the SCOTS. We compute new instances by joining parts of the existing instances onto each other. The resulting benchmark set contains 174 instances. The construction of the benchmark set will be detailed in an upcoming publication.

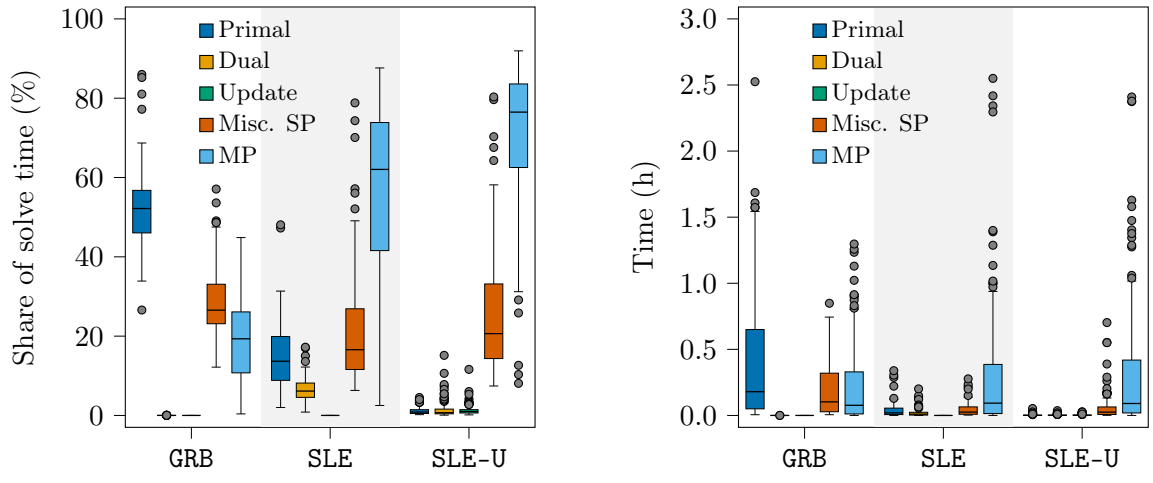
Computational Setup. We use machines with 2× Intel Xeon L5630 quad-core processors at 2.13 GHz (8 cores, 16 GB RAM) on Debian GNU/Linux 11 across all experiments. A three-hour time limit is imposed per instance.

7.1 Results

We compare the three methods under scenarios with a reduced number of available switch arcs. For the remaining discussion in this section, we disregard all 25 instances that are infeasible or where all instances encountered the time limit, leaving a remaining 149 instances. Where primal solutions were found, the reported gaps for instances exceeding the time limit remain within 1.00 % for almost all instances, and differ only marginally between solution methods. We hence omit gap values from further discussion.

Time distribution. Since we exclusively change the method for solving subproblems while leaving the output unchanged, the largest difference between the methods can be expected to occur in the subproblem running time. Figure 4 shows the relative distribution of the different parts of the solving process for all three methods in box plots. We observe that for `GRB`, the mean subproblem running time dominates the overall running time at 72.49 %, with Gurobi's

Figure 4: Mean running time distribution.



(a) Relative. (b) Absolute.

Note. Legend in order from left to right for each method. "Primal" is the time spent exclusively solving the subproblem LP with Gurobi or the primal SLE, respectively. "Dual" is the equivalent for the dual SLE for `SLE` and `SLE-U`. "Update" is the factorization update time for `SLE-U`. "Misc. SP" is any remaining SP time, including connectedness separation and generation of the cut object. "MP" is the MP solving time. Each box spans the interquartile range, whiskers extend to 1.5× IQR, dots are outliers.

solving process taking up 49.46 % ("Primal", dual solutions are obtained without relevant additional effort). As such, noticeable improvements in that solving process also result in noticeable improvements in the overall solving process. While the absolute time spent on solving the MP remains relatively constant for all methods, its relative running time contribution increases to 75.25 % for `SLE` and 82.93 % for `SLE-U`. This is due to reductions in the time spent on solving the "core" of the SP, i.e., in the LP or SLE solver: for `SLE`, solving the primal SLEs uses 9.35 %, and solving the optimality and feasibility SLEs 4.75 % of running time. For `SLE-U`, these are further reduced to 0.70 % and 0.69 %, respectively, with an additional 0.77 % for factorization updates. The remaining time is taken up by miscellaneous subproblem computations, including cut construction and allocation, checking for disconnected components, updating the LP and SLE objects and checking the solutions against bounds. Despite effort spent to reduce the time spent on these computations, it remains a noticeable contributor. Further improvements here are likely possible, but would not influence the SP solving time.

MP and miscellaneous SP times can be seen as relatively static, as they are not directly influenced by the choice of SP solver. This can be observed particularly for `SLE` and `SLE-U`, where they match very closely (1099.73 s vs. 1142.31 s). `GRB` times differ slightly, this is due to a slightly different set of miscellaneous SP computations, as well as possible differences in dual solutions. While we often observed equal dual solutions, up to numerical differences, between Gurobi and our algorithms, the subproblems do not necessarily have unique solutions. In fact, as discussed, while our dual solutions are always feasible and where applicable optimal, a different choice of connected component spanning trees or violated flow or potential for feasibility cuts can yield different solutions for our algorithms as well. These differences seem to be mildly disadvantageous to our methods (95747.2581 solved SPs for `GRB`, 98962.1218 for `SLE`, 101760.1538 for `SLE-U` on average).While this may point to slightly weaker cuts that we produce, it can also be attributed to advanced internal solver mechanisms that are not publicly documented.

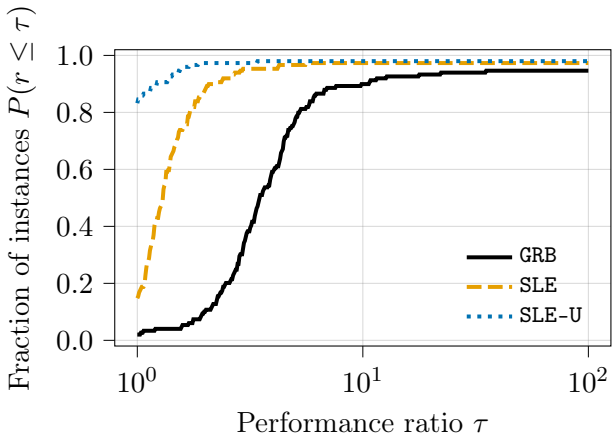
Running time comparison. Table 1 provides an aggregate view on the running time behavior of the methods. In every metric, there is a clear hierarchy, with `SLE-U` as the fastest solver, `SLE` in second, and `GRB` last.

Table 1: Aggregate running time statistics for GRB, SLE and SLE-U.

	Solved	Mean	Geo. Mean
GRB	141	2321.6646	958.7539
SLE	146	986.1129	367.8782
SLE-U	147	906.2038	276.0177

Note. "Mean" is the arithmetic mean, "geo. mean" is the shifted geometric mean with shift 10. Aggregate values cover only the 136 instances solved by all methods.

Figure 5: Performance profile for total solve time.



To investigate the methods as a part of the overall solving process as well as individually, the following time comparisons feature the total solve time (Figure 5), the total SP time (Figure 6a), which includes the miscellaneous SP time, and the SP solve time ("Primal," "Dual," and "Update" in Figure 6b). The figures depict performance profiles, where the x-axis shows the relative slowdown compared to the best performing method on a given instance, and the y-axis the fraction of instances where the slowdown is at most as large as the x-value. For example, in Figure 5 for 89.93 % of instances, GRB is at most 10 times slower than the best performing method.

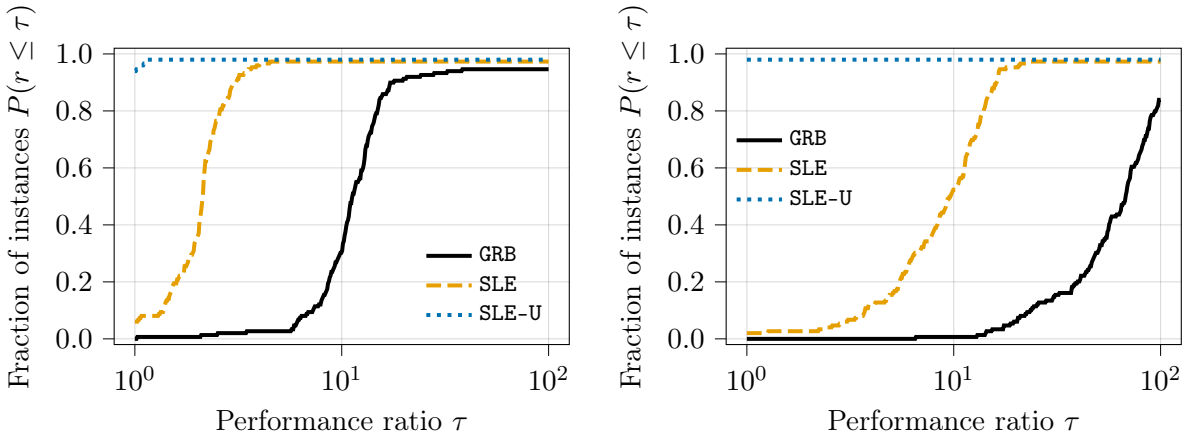
For total running time, Figure 5, SLE-U is the fastest solver for 83.22 % of instances, and less than 54.77 % slower for 95.00 %. SLE is the fastest method for 14.77 % of instances. It is at least 2 times slower for about 12.08 % of instances, and at least 3 times slower for about 4.70 % of instances. GRB is by far the slowest method. Only 10.07 % of instances are at most 2 times slower. Half of all instances are at least 3.4924 times slower, 10.07 % of instances are at least 10 times slower and 5.37 % of instances are at least 100 times slower.

The magnitude of the differences increases when observing only the SP time. Here, SLE is at least 2 times slower for 64.43 % of instances, and at least 3 times slower for about 10.07 % of instances. For GRB, 69.80 % of instances are at least 10 times slower. 5.37 % of instances remain at least 100 times slower.

The SP solve time, the focus of the methods, shows the largest improvements. SLE-U is the fastest method for all instances. SLE is at least 2 times slower for 97.32 % of instances, and at least 10 times slower for about 47.65 % of instances. For GRB, 86.58 % of instances are at least 30 times slower, and 16.11 % of instances are at least 100 times slower.

In Section 6, we discussed the theoretical complexity of the methods, and observed that, for single line arc contingencies, SLE has a running time complexity in $\mathcal{O}(|A| \cdot |V|^3)$ and SLE-U in $\mathcal{O}(|V|^3 + |A| \cdot |V|^2)$. While $|A|$ would be equal to $|V|^2$ for a fully connected graph, the instances in the benchmark set have an average of only 1.4683 arcs per vertex, i.e., the underlying graphs

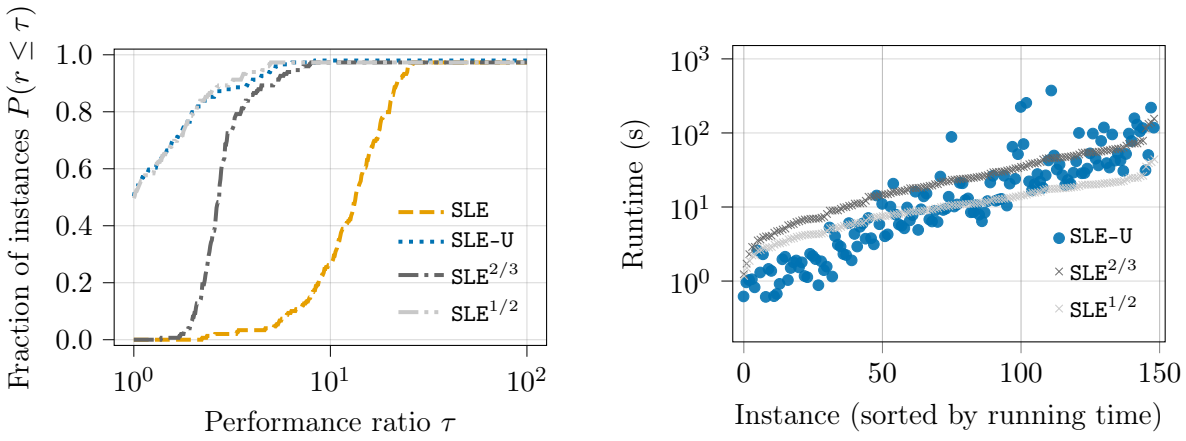
Figure 6: Performance profiles for SP time.



(a) Total time.

(b) Solve time (GRB/SLEs).

Figure 7: Synthesized data for SP solve times (SLEs).



(a) Performance profile.

(b) Scatter plot.

Note. Data for artificial methods $\texttt{SLE}^{2/3}$ and $\texttt{SLE}^{1/2}$ is synthesized from individual SLE data points.

are sparsely connected, and thus a relatively low number of subproblems are solved per iteration. Additionally, the SLE matrices corresponding to a graph with all switch arcs active and without contingencies have an average number of non-zero entries of 1.80 % over all instances. SLE solution algorithms specialized on sparse matrices exhibit near-linear running time based on the number of non-zero elements of the matrix (Schaub et al., 2017; Spielman and Teng, 2004).

Figures 7a and 7b visualize this relationship. Here, we added artificial methods $\texttt{SLE}^{2/3}$ and $\texttt{SLE}^{1/2}$, and compare them with SLE and SLE-U. For each instance, letting $t \in \mathbb{R}$ be the time in seconds needed by SLE, the time needed for $\texttt{SLE}^{2/3}$ and $\texttt{SLE}^{1/2}$ is set to be $t^{2/3}$ and $t^{1/2}$, respectively. This roughly emulates a hypothetical method in a complexity class correspondingly reduced compared to SLE. Note that this view ignores hidden constants in $\mathcal{O}$-notation and only serves as an intuition. In Figure 7a, SLE-U matches closely with $\texttt{SLE}^{1/2}$, suggesting that it indeed exhibits a near-linear running time and SLE a quadratic one. Note, however, that the y-axis of the performance denotes the fraction of instances instead of specific instances. An instance-by-instance comparison is provided by Figure 7b. SLE-U seems to be closer to $\texttt{SLE}^{1/2}$ for easier instances, and slows down to between $\texttt{SLE}^{1/2}$ and $\texttt{SLE}^{2/3}$ for harder instances.

8 Conclusion

Potential-based flow problems cover a range of different network flow related problems. Due to their application to several infrastructure domains, they are of significant practical importance. Affine potential functions cover, for example, applications in lossless DC power systems. Benders decomposition is a common exact algorithm in this area, particularly when considering different scenarios simultaneously. Even though the resulting subproblems are linear programs, typically solved with a simplex algorithm, they can account for a major share of the total running time.

Our contribution builds on the classical fact that in a passive, potential-based network with fixed topology and nodal demands, the resulting flow is unique (as observed by Birkhoff and Diaz, 1956). We conclude that in the subproblems "there is nothing to optimize," and thus the general simplex method is a considerable overhead. Our work can be interpreted as exposing this special problem structure to the simplex algorithm. Primal solutions are obtained from solving a single, carefully crafted system of linear equations (SLE). More importantly, dual solutions are required to construct Benders cuts. It is only natural that dual solutions arise from solving the same SLE ("transposed"), and we develop that this is indeed the case (and deriving a dual direction of unboundedness requires only a little more work). Overall, we need to solve at most two SLEs per subproblem. Just like in the simplex method, we accelerate solving the SLEs by factorizing the coefficient matrices. Over multiple subproblems, factorizations are updated, thereby reducing the theoretical complexity and practical running times in certain, natural cases. It should be noted that our SLEs have smaller coefficient matrices than those that would be solved in the simplex method. Overall, this gives us an efficient "simplex-free," but "simplex-inspired" subproblem solving algorithm for affine potential-based flow problems with switchable arcs and arc-based contingency scenarios.

Using our approach, we were able to obtain significant improvements over Gurobi as a baseline on instances synthesized from the PGLib-OPF (Babaeinejadsarookolaee et al., 2019). The shifted geometric mean of the overall running time is reduced by 71 %. The subproblem solving time share decreases from 49 % for Gurobi to a little more than 2 % for our approach. Empirical results also match theoretical results regarding factorization update complexity.

Finally, we would like to mention two potential further developments of our work, one easier, one more challenging:

- The Benders algorithm presented in this work, where we solve all subproblems in every iteration and obtain at most one cut per subproblem, is only one option for handling multiple scenarios. It might be worthwhile to compute additional cuts, for example by choosing multiple violated components for a feasibility cut or by solving additional, related subproblems in a given Benders iteration. Adding additional cuts can lead to a reduced number of iterations overall, and thus less time spent in the master problem. Our algorithm allows for exploring many additional subproblems at very low computational cost.

- A natural extension would be to consider flow problems with non-affine potential functions. However, our algorithm relies on the subproblems being linear to construct linear systems. Nevertheless, it suffices that potential functions ψ are continuous, strictly increasing and $\psi(0) = 0$ for the flow to be unique (Pfetsch et al., 2026), so there is room for improvement also in these cases.

Acknowledgments

The authors want to thank Alexander Helber for pointing out the uniqueness property for the primal subproblems. This research was supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK) under grant 03EI4056D within the InnOpTEM project (September 2022–August 2025).

References

O Alsac, B Stott, and WF Tinney. Sparsity-oriented compensation methods for modified network solutions. *IEEE Transactions on Power Apparatus and Systems*, PAS-102(5):1050–1060, 1983. doi: 10.1109/TPAS.1983.318045.

Sogol Babaeinejadsarookolaee, Adam Birchfield, Richard D Christie, Carleton Coffrin, Christopher DeMarco, Ruisheng Diao, Michael Ferris, Stephane Fliscounakis, Scott Greene, Renke Huang, et al. The power grid library for benchmarking AC optimal power flow algorithms. *arXiv preprint arXiv:1908.02788*, 2019. doi: 10.48550/arXiv.1908.02788.

BDEW. Redispatch in deutschland, 2023. URL https://www.bdew.de/media/documents/BDEW-Redispatch_Bericht_2023_zum_Berichtsjahr_2022.pdf.

Jacques F Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962. doi: 10.1007/BF01386316.

Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Belmont, MA, 1997.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017. doi: 10.1137/141000671.

Silvio Binato, Mário Veiga F Pereira, and Sérgio Granville. A new Benders decomposition approach to solve power transmission network design problems. *IEEE Transactions on Power Systems*, 16(2): 235–240, 2001. doi: 10.1109/59.918292.

Garrett Birkhoff and Joaquin B Diaz. Non-linear network problems. *Quarterly of Applied Mathematics*, 13(4):431–443, 1956. doi: 10.1090/QAM/77398.

BNetzA. Netzausbau Netzentwicklungspläne, 2016. URL https://www.netzausbau.de/SharedDocs/Downloads/DE/Infomaterial/BroschuereNEP.pdf?__blob=publicationFile.

BNetzA. Strom, 2023. URL https://www.bundesnetzagentur.de/DE/Fachthemen/ElektrizitaetundGas/Versorgungssicherheit/Strom/start.html.

Pascal Börner, Max Klimm, Annette Lutz, Marc E Pfetsch, Martin Skutella, and Lea Strubberg. Valid cuts for the design of potential-based flow networks. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 142–156. Springer, 2025. doi: 10.1007/978-3-031-93112-3_11.

Jens Burgschweiger, Bernd Gnädig, and Marc C Steinbach. Optimization models for operative planning in drinking water networks. *Optimization and Engineering*, 10(1):43–73, 2009. doi: 10.1007/s11081-008-9040-8.

Carleton Coffrin, Russell Bent, Kaarthik Sundar, Yeesian Ng, and Miles Lubin. Powermodels.jl: An open-source framework for exploring power flow formulations. In *2018 Power Systems Computation Conference (PSCC)*, pages 1–8, June 2018. doi: 10.23919/PSCC.2018.8442948.

Ivan Contreras, Jean-François Cordeau, and Gilbert Laporte. Benders decomposition for large-scale uncapacitated hub location. *Operations research*, 59(6):1477–1490, 2011. doi: 10.1287/OPRE.1110.0965.

Carlos M Correa-Posada and Pedro Sánchez-Martín. Gas network optimization: A comparison of piecewise linear models. *Optimization Online*, 2014.

Stella C Dafermos and Frederick T Sparrow. The traffic assignment problem for a general network. *Journal of Research of the National Bureau of Standards, Section B*, 73B(2):91–118, 1969. doi: 10.6028/JRES.073B.010.

Matteo Fischetti, Ivana Ljubić, and Markus Sinnl. Redesigning Benders decomposition for large-scale facility location. *Management Science*, 63(7):2146–2162, 2017. doi: 10.1287/MNSC.2016.2461.

Emily B Fisher, Richard P O'Neill, and Michael C Ferris. Optimal transmission switching. *IEEE Transactions on Power Systems*, 23(3):1346–1355, 2008. doi: 10.1109/TPWRS.2008.922256.

Arthur M Geoffrion. Generalized Benders decomposition. *Journal of optimization theory and applications*, 10(4):237–260, 1972. doi: 10.1007/BF00934810.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU Press, 2013.

Martin Groß, Marc E Pfetsch, Lars Schewe, Martin Schmidt, and Martin Skutella. Algorithmic results for potential-based flows: Easy and hard cases. *Networks*, 73(3):306–324, 2019. doi: 10.1002/NET.21865.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2026. URL https://www.gurobi.com.

Kory W. Hedman, Richard P. O'Neill, Emily Bartholomew Fisher, and Shmuel S. Oren. Optimal transmission switching with contingency analysis. *IEEE Transactions on Power Systems*, 24(3):1577–1586, August 2009. ISSN 1558-0679. doi: 10.1109/TPWRS.2009.2020530.

Majid Heidarifar and Hassan Ghasemi. A network topology optimization model based on substation and node-breaker modeling. *IEEE Transactions on Power Systems*, 31(1):247–255, 2016. doi: 10.1109/TPWRS.2015.2399473.

Chris T Hendrickson and Bruce N Janson. A common network flow formulation for several civil engineering problems. *Civil Engineering Systems*, 1(4):195–203, 1984. doi: 10.1080/02630258408970343.

Jesco Humpola. *Gas network optimization by MINLP*. Logos Verlag Berlin GmbH, 2017.

Charles Royal Johnson, Carlos Marijuán, and Miriam Pisonero. Diagonal dominance and invertibility of matrices. *Special Matrices*, 11(1):20220181, 2023. doi: 10.1515/spma-2022-0181. URL https://doi.org/10.1515/spma-2022-0181.

Amin Khodaei and Mohammad Shahidehpour. Transmission switching in security-constrained unit commitment. *IEEE Transactions on Power Systems*, 25(4):1937–1945, 2010. doi: 10.1109/TPWRS.2010.2046344.

Will Kimmerer, Vedant Puri, Jash Ambaliya, and Chris Rackauckas. Linearsolve.jl, 2026. URL https://github.com/SciML/LinearSolve.jl.

Thorsten Koch, Benjamin Hiller, Marc E Pfetsch, and Lars Schewe, editors. *Evaluating gas network capacities*. SIAM, 2015.

Larry J LeBlanc, Edward K Morlok, and William P Pierskalla. An efficient approach to solving the road network equilibrium traffic assignment problem. *Transportation Research*, 9(5):309–318, 1975. doi: 10.1016/0041-1647(75)90030-1.

Karsten Lehmann, Alban Grastien, and Pascal Van Hentenryck. The complexity of DC-switching problems. *arXiv preprint arXiv:1411.4369*, 2014. doi: 10.48550/arXiv.1411.4369.

Yang Li, Zhiyi Li, Fushuan Wen, and Mohammad Shahidehpour. Minimax-regret robust co-optimization for enhancing the resilience of integrated power distribution and natural gas systems. *IEEE Transactions on Sustainable Energy*, 11(1):61–71, 2020. doi: 10.1109/TSTE.2018.2883718.

Miles Lubin, Oscar Dowson, Joaquim Dias Garcia, Joey Huchette, Benoît Legat, and Juan Pablo Vielma. JuMP 1.0: Recent improvements to a modeling language for mathematical optimization. *Mathematical Programming Computation*, 15:581–589, 2023. doi: 10.1007/s12532-023-00239-3.

Marzieh Mozafari and Alireza Zabihi. Robust water supply chain network design under uncertainty in capacity. *Water Resources Management*, 34(13):4093–4112, 2020. doi: 10.1007/s11269-020-02658-6.

Marc E. Pfetsch, Martin Schmidt, Martin Skutella, and Johannes Thürauf. Potential-based flows – an overview, January 2026. URL https://optimization-online.org/wp-content/uploads/2026/01/ch_potential.pdf.

Ragheb Rahmaniani, Teodor Gabriel Crainic, Michel Gendreau, and Walter Rei. The Benders decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3):801–817, 2017. doi: 10.1016/j.ejor.2016.12.005.

R Tyrrell Rockafellar. *Network Flows and Monotropic Optimization*. Athena Scientific, 1998.

Carlos Ruiz and Antonio J Conejo. Robust transmission expansion planning. *European Journal of Operational Research*, 242(2):390–401, 2015. doi: 10.1016/j.ejor.2014.10.030.

Michael T Schaub, Maguy Trefois, Paul Van Dooren, and Jean-Charles Delvenne. Sparse matrix factorizations for fast linear solvers with application to Laplacian systems. *SIAM Journal on Matrix Analysis and Applications*, 38(2):505–529, 2017. doi: 10.1137/16M1077398.

Martin Schmidt and Johannes Thürauf. An exact method for nonlinear network flow interdiction problems. *SIAM Journal on Optimization*, 34(4):3623–3652, 2024. doi: 10.1137/22M152983X.

Daniel A Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 81–90, 2004. doi: 10.1145/1007352.1007372.

Peter Stange, Andreas Griewank, and Matthias Bollhöfer. On the efficient update of rectangular LU-factorizations subject to low-rank modifications. *Electronic Transactions on Numerical Analysis*, 26:161–177, 2007.

Johannes Thürauf, Julia Grübel, and Martin Schmidt. Adjustable robust nonlinear network design without controllable elements under load scenario uncertainties. *Mathematical Programming*, 218: 403–437, 2026. doi: 10.1007/s10107-025-02207-2.

Robert J. Vanderbei. *Linear Programming: Foundations and Extensions*. Springer, 1998.

Lei Wu and Mohammad Shahidehpour. Accelerating the Benders decomposition for network-constrained unit commitment problems. *Energy Systems*, 1(3):339–376, 2010. doi: 10.1007/s12667-010-0015-4.

Jingwei Yang, Ning Zhang, Chongqing Kang, and Pierre Pinson. Modeling the transient security constraints of natural gas network in day-ahead power system scheduling. In *2017 IEEE Power & Energy Society General Meeting*, pages 1–5. IEEE, 2017. doi: 10.1109/PESGM.2017.8274497.