Algorithmic Results for Potential-Based Flows: Easy and Hard Cases

Martin Gross, Marc E. Pfetsch, Lars Schewe, Martin Schmidt, and Martin Skutella

Abstract. Potential-based flows are an extension of classical network flows in which the flow on an arc is determined by the difference of the potentials of its incident nodes. Such flows are unique and arise, for example, in energy networks. Two important algorithmic problems are to determine whether there exists a feasible flow and to maximize the flow between two designated nodes. We show that these problems can be solved for the single source and sink case by reducing the network to a single arc. However, if we additionally consider switches that allow to force the flow to 0 and decouple the potentials, these problems are NP-hard. Nevertheless, for particular series-parallel networks, one can use algorithms for the subset sum problem. Moreover, applying network presolving based on generalized series-parallel structures allows to significantly reduce the size of realistic energy networks.

1. Introduction

This paper deals with algorithmic questions related to *potential-based flows*, which form an extension of classical network flows. Given a directed graph, these flows depend on the differences of potentials at the incident nodes. Such potential-based flows often appear in energy networks like in gas, power, or water transportation. In contrast to the classical case, the flow in such networks follows physical laws that make it unique. On the one side, this seems to make the analysis easier, but on the other side, the dependency between the flows and potentials is often nonlinear, which leads to a harder class of optimization problems.

Motivated by their physical applications, such potential-based networks have been studied repeatedly in the literature. One of the earliest references seems to be Birkhoff and Diaz [3]. Clearly, there is an abundance of further articles dealing with the networks of particular applications. Research of potential-based networks has mainly focused on two topics: uniqueness of the solutions and algorithms. Uniqueness was studied, for instance, in [3], Collins et al. [12], and Maugis [29]; see Section 3 below for a review. However, the cited articles also discuss algorithms to compute the solution; see Rockafellar [32] for a general treatment.

In this paper, we extend these investigations by studying algorithmic questions related to potential-based networks. In particular, we are interested in the following two algorithmic problems: Compute a feasible flow, if it exists, and find a maximal flow between two designated nodes. These two problems can be seen as a direct analogue of the corresponding classical flow problems. In addition, we study networks that may contain types of arcs that act as switches, i.e., it is possible to force the flow to 0 and decouple the potentials on the corresponding nodes.

Date: August 4, 2017.

2010 *Mathematics Subject Classification.* 90-XX, 90B10, 90C35, 05C21.

Key words and phrases. Potential networks, Potential-based flows, Maximum flow problem, Series-parallel graphs, Network reduction, NP-hardness.

We begin our study by introducing the model and its assumptions in Section 2. In this section, we also explain the relation to gas, water, and power transmission networks and review existing hardness results. We then treat passive networks, i.e., networks that do not contain switches, in Section 3. We show that networks that contain a single source and sink can be reduced to an equivalent network with only one arc. This allows to solve the above mentioned algorithmic problems efficiently. Next, we introduce generalized series-parallel graphs in Section 4 and investigate whether the two problems can be efficiently solved on this class of graphs. We then turn to networks that may contain switches, which turn out to be harder to handle. In Section 5, we discuss monotonicity properties of active networks and their relation to Braess-paradoxes. Moreover, we discuss an approximate reduction of the multiple source/sink to the single source/sink case. We then exploit the favorable algorithmic properties of generalized series-parallel graphs and investigate the structure of realistic energy networks by applying series-parallel reductions. The computations in Section 6 show that using such presolving steps can considerably reduce the size of realistic networks, highlighting the special role of series-parallel network structures. In fact, we can reduce the size of large real-world gas networks by up to 76 %.

In total, this paper provides analysis of the algorithmic questions related to potential-based networks and shows that, on the one hand, the problems on passive single-source and sink networks can be treated efficiently and, on the other hand, investigates networks with switches. In fact, using presolving operations significantly helps to reduce the size of realistic energy networks.

2. Basic Model

In this section, we present the general framework of potential-based flows and illustrate it by explaining how the already mentioned examples of gas, power, and water flows fit into this framework.

2.1. **Potential Networks.** We define a *potential network* G as follows. It consists of a directed graph $G = (V, A)$ with node set $V = V(G)$ and arc set $A = A(G)$. To each node $v \in V$ a *potential* π_v is assigned with lower and upper bounds $\underline{\pi}_v$, $\overline{\pi}_v \in \mathbb{R}$, $\underline{\pi}_v \leq \overline{\pi}_v$, and to every arc $a \in A$ a *flow* x_a is assigned with lower and upper bounds $\underline{x}_a$, $\overline{x}_a \in \mathbb{R}$, $\underline{x}_a \leq \overline{x}_a$. Note that the flow x_a on an arc $a = (u, v)$ can be negative with the interpretation of flow against the direction of the arc, whereas flow from u to v is modeled by positive flows. The arc set A is partitioned into arcs $A_L = A_L(G)$ representing so-called *lines* and arcs $A_S = A_S(G)$ representing *switches*. We call a potential network *passive* if $A_S = \emptyset$ holds. Otherwise, we call it *active*.

For every *switch-arc* $a = (u, v) \in A_S$, we have a control variable $y_a \in \{0, 1\}$ specifying the state of the switch. If $y_a = 1$, then $\pi_u = \pi_v$ has to hold and the flow x_a is not restricted by the incident potentials. Otherwise, if $y_a = 0$, the flow on the arc has to be 0, i.e., $x_a = 0$, and the potentials are decoupled from (1) below. In these cases, the switch is said to be "on" and "off", respectively.

For a *line-arc* $a = (u, v) \in A_L$, the *flow* x_a depends on the potential difference at its incident nodes u and v:

$$\pi_u - \pi_v = \beta_a \, \psi(x_a). \tag{1}$$

Here, $\beta_a > 0$ is an arc specific constant. Moreover, $\psi : \mathbb{R} \to \mathbb{R}$ is a *potential function* that is the same for all arcs $a \in A_L$ and has the following properties:

(1) ψ is continuous,
(2) ψ is strictly increasing, and
(3) ψ is an odd function, i.e., $\psi(-x) = -\psi(x)$.

The potential function ψ depends on the application; we will later see typical choices for ψ in natural gas, water, and power networks, where potential functions have the properties (1)–(3). Note that different line arcs in a potential network are only distinguished by their values of β_a. The assumptions that ψ is continuous and strictly increasing are natural in the mentioned physical contexts. Note that ψ^{-1} exists under these assumptions. Moreover, the assumption that ψ is odd makes the situation symmetric with respect to the direction of the flow.

If ψ is *positively homogeneous* of order $r > 0$, i.e.,

$$\psi(\lambda\, x) = \lambda^r\, \psi(x), \quad \lambda > 0,$$

we call the potential network *homogeneous*. Homogeneity is motivated by physical laws, and its order r depends on the different applications. Moreover, it implies that ψ is of the form

$$\psi(x) = \alpha \operatorname{sgn}(x)\, |x|^{\,r}, \tag{2}$$

for some constant $\alpha = \psi(1) > 0$ and order $r > 0$.

We also assume that both ψ and ψ^{-1} are efficiently computable. We note that when we write about computational complexity, e.g., in the literature review in Section 2.4, we take the classical Turing machine as our computational model. For the applications with more involved potential functions ψ (e.g., in water networks), another model such as the BSS-machine [4, 5] or a real RAM [37]. We refrain from these extensions, however, since for our positive results we do not need to compute ψ^{-1}. This would require to compute roots, which need to be approximated in all of these models. This can be done in polynomial time if we allow for a fixed $\varepsilon > 0$ for the approximation error as discussed in [1].

Overall, given the potentials at the incident nodes, we can compute the flow on an arc by

$$x_a = \psi^{-1}\big((\pi_u - \pi_v)/\beta_a\big).$$

Finally, note that the name potential is motivated by the physical applications. The term is also used with a different meaning for the dual variables of the classical flow problem.

2.2. **Potential-Based Flows.** A *potential-based flow* (x, π, y) in a potential network $G = (V, A)$ consists of a flow $x \in \mathbb{R}^A$, potentials $\pi \in \mathbb{R}^V$, and control variables $y \in \{0,1\}^{A_S}$. We are mainly interested in the setting with one source $s \in V$ and one sink $t \in V$, where $s \neq t$. For a *potential-based s-t-flow* (x, π, y) we require flow conservation at the inner nodes,

$$\sum_{a\in\delta^{\text{out}}(v)} x_a - \sum_{a\in\delta^{\text{in}}(v)} x_a = 0, \quad v \in V \setminus \{s,t\}\,,$$

where $\delta^{\text{out}}(v) := \{(v,w) \in A\}$, $\delta^{\text{in}}(v) := \{(u,v) \in A\}$. The flow *value* is defined as the net amount of flow sent from the source:

$$\operatorname{val}(x,\pi,y) := \sum_{a\in\delta^{\text{out}}(s)} x_a - \sum_{a\in\delta^{\text{in}}(s)} x_a.$$

Notice that this amount might be negative, meaning that $-\operatorname{val}(x,\pi,y)$ flow is actually being sent from t to s. If the network is passive, we sometimes omit y, i.e., we write $\operatorname{val}(x,\pi)$.

We call (x, π, y) a *feasible potential-based s-t-flow* if it satisfies the following constraints:

$$\sum_{a\in\delta^{\text{out}}(v)} x_a - \sum_{a\in\delta^{\text{in}}(v)} x_a = 0, \qquad v \in V \setminus \{s,t\}, \tag{3a}$$

$$\pi_u - \pi_v = \beta_a \, \psi(x_a), \qquad a = (u,v) \in A_L, \tag{3b}$$

$$-F\, y_a \le x_a \le F\, y_a, \qquad a \in A_S, \tag{3c}$$

$$(\underline{\pi}_u - \overline{\pi}_v)\,(1 - y_a) \le \pi_u - \pi_v \le (\overline{\pi}_u - \underline{\pi}_v)\,(1 - y_a), \qquad a = (u,v) \in A_S, \tag{3d}$$

$$x_a \in [\underline{x}_a, \overline{x}_a], \qquad a \in A, \tag{3e}$$

$$\pi_v \in [\underline{\pi}_v, \overline{\pi}_v], \qquad v \in V, \tag{3f}$$

$$y_a \in \{0,1\}, \qquad a \in A_S. \tag{3g}$$

In (3c), F is an upper bound on the flow on any arc. Further flow bounds are given by $\underline{x}_a$, $\overline{x}_a$ or can be derived using the potential bounds $\underline{\pi}_v$, $\overline{\pi}_v$ and (3b). Note that the system implies that if $y_a = 1$ for a switch arc $a \in A_S$, then $\pi_u = \pi_v$ and the flow is not restricted by (3c). Similarly, if $y_a = 0$, then $x_a = 0$ and the potentials are not restricted by (3d).

Note that (sub)networks that only consist of switches allow to represent classical flows. This also implies that the flow is not necessarily unique in these cases (see Section 3 below for a discussion of uniqueness in passive networks). But such (sub)networks are not realistic, and it makes sense to exclude them. The methods developed below, however, do not need this exclusion.

In the remainder of this section, we discuss different specific examples of the setting described so far. These examples are given by gas, water, and power networks, which illustrate the broad applicability of our model and results. In all of these cases, the models are an approximation of a description of the relevant physics, typically described by partial differential equations; see, e.g., Hante et al. [18].

Example 2.1 (Gas Transport Networks)**.** In stationary models of gas transport networks, lines correspond to gas pipes and switches model valves. In this context, potentials at nodes correspond to squares of gas pressures and flows are gas mass flows. The relation between mass flows and gas pressure squares at gas pipes $a = (u,v) \in A_L$ is typically approximated by pressure loss functions of the type

$$\pi_u - \pi_v = \beta_a \, \psi(x_a), \quad \psi(x_a) = |x_a| \, x_a. \tag{4}$$

The arc-specific constant β_a is positive and depends on technical parameters of the pipe like its diameter, its length, or the roughness of its inner wall, etc. For more details on modeling stationary gas flow in pipeline networks see the chapter Fügenschuh et al. [15] of the book Koch et al. [23] and the references therein.

Note that (4) assumes a constant elevation of the network. However, we can also treat the following model, which is one of the most accurate algebraic approximations of gas flow on a pipe, see again the chapter [15] in [23] for a derivation.

Observation 2.2. *Let h_v be the height of node $v \in V$. The algebraic gas flow model*

$$\pi_v = \left(\pi_u - \Lambda_a \, |x_a| \, x_a \frac{e^{S_a} - 1}{S_a}\right) e^{-S_a}, \tag{5}$$

where Λ_a are arc-dependent constants and $S_a = \delta\,(h_v - h_u)$, $\delta > 0$, is a scaled version of the potential-flow model (4).

Indeed, multiplying (5) with $e^{\delta h_v}$, we obtain

$$e^{\delta\, h_v} \pi_v = e^{\delta\, h_u} \pi_u - \Lambda_a \, |x_a| \, x_a \frac{(1 - e^{-S_a})}{S_a} \, e^{\delta h_v}.$$

This is just a scaled version of (4) for an appropriate definition of β_a.

Example 2.3 (Water Transport Networks)**.** In water networks, potentials correspond to hydraulic heads. The head-loss model is also based on (1) with

$$\psi(x_a) = \operatorname{sgn}(x_a)\,|x_a|^{\,1.852}$$

and arc-specific constants $\beta_a > 0$. For a more detailed description we refer to Larock et al. [26].

Example 2.4 (Lossless DC Power Flow Networks)**.** In lossless DC power flow networks, the flow model is linear, but also satisfies all properties discussed above. In this context, $\beta_a = 1/B_a$ is the reciprocal of the susceptance B_a of the line $a \in A_L$ and the linear potential function is given by $\psi(x_a) = x_a$. See Kirschen and Strbac [22] as well as Bienstock [2] for a discussion of this and the more involved AC case.

2.3. **Algorithmic Problems for Potential-Based Flows.** We now introduce two natural algorithmic problems for potential-based s-t-flows that we consider in the following. The first problem deals with the question whether a feasible flow exists.

> s-t-FlowFeasibility
> **Input:** A potential network G.
> **Problem:** Is there a feasible potential-based s-t-flow (x, π, y) for G?

Recall that a feasible potential-based flow consists of (x, π, y) such that the conditions in (3) are satisfied. Moreover, we consider the problem to maximize the flow value as well.

> s-t-MaxFlow
> **Input:** A potential network G.
> **Problem:** Find a feasible potential-based s-t-flow (x, π, y) for G of maximal value $\operatorname{val}(x, \pi, y)$.

These two problems are natural analogues of algorithmic problems for classical flows. One could also consider the analogue of the min-cost flow problem, but this seems to be less natural, since the flow is unique on passive networks.

2.4. **Hardness.** In this subsection, we discuss the computational complexity of the s-t-FlowFeasibility and s-t-MaxFlow problems. To the best of our knowledge, the complexity of these problems has not been studied in the settings of potential flows that we use. However, there are results known for analogous feasibility and maximization problems for gas and DC power networks. These complexity results carry over to the potential flow setting—at least in the cases that we will mention now.

Lehmann et al. [28] showed that the DC power analogue of s-t-FlowFeasibility is strongly NP-hard for planar graphs of maximum degree 3. They also show that a generalization of s-t-MaxFlow with at least two sources and sinks cannot be approximated in polynomial time better than $2^{(\log n)^{1-\varepsilon}}$ for an $\varepsilon > 0$ unless the problems in NP can be solved in quasi-polynomial deterministic time. For an arbitrary number of sources and sinks, they show that their analogue of the s-t-FlowFeasibility problem is weakly NP-hard even on cactus graphs of maximum degree 3. For AC networks, the same authors show in a different paper [27] that an analogue of the s-t-FlowFeasibility problem is hard even on trees.

For gas networks, Szabó [38] showed that for a feasibility problem that he calls the *active gas nomination validation problem*, which is analogous to s-t-FlowFeasibility, is weakly NP-hard even for series-parallel networks. For arbitrary networks, Humpola [20] showed that a similar problem to s-t-FlowFeasibility, which he calls the *topology optimization problem* is strongly NP-hard.

3. Passive Networks

In this section, we discuss passive potential networks. A known property of these networks is that, in the absence of potential and flow bounds, every balanced set of supplies and demands can be transported and once a single potential in every connected component is fixed, the solutions are uniquely determined. This statement holds for the case of node balances, i.e., the net outflow of a node $u \in V$ is required to be equal to a given balance b_u. Given node balances $b \in \mathbb{R}^V$ with $\sum_{v \in V} b_v = 0$, a *potential-based b-flow* (x, π) satisfies (1) and flow conservation constraints

$$\sum_{a \in \delta^{\text{out}}(v)} x_a - \sum_{a \in \delta^{\text{in}}(v)} x_a = b_v, \quad v \in V. \tag{6}$$

In classical network flow theory, scaling a given b-flow x by some parameter $\lambda \in \mathbb{R}$ yields a λb-flow λx. The following observation is a straightforward generalization to potential-based flows in homogeneous passive networks.

Observation 3.1. *Consider a homogeneous passive network of order $r > 0$ with node balances $b \in \mathbb{R}^V$ and a potential-based b-flow (x, π). Then, for any $\lambda \in \mathbb{R}$, $(\lambda x, \operatorname{sgn}(\lambda) \, |\lambda|^r \, \pi)$ is a potential-based λb-flow.*

Proof. The observation follows directly from Equations (1) and (2). □

A potential-based b-flow is said to be *feasible* if, in addition to (6), it obeys the flow bounds (3e) and potential bounds (3f).

Theorem 3.2 (Collins et al. [12] and Ríos-Mercado et al. [31])**.** *Let G be a passive connected potential network and let $b \in \mathbb{R}^V$ be a vector of node balances with $\sum_{v \in V} b_v = 0$. Furthermore, assume that no potential and flow bounds are given and that for a given node $s \in V$ the potential π_s is fixed. Then, there exists a unique feasible potential-based b-flow (x, π).*

One can prove this theorem in different ways: The approach of [31] shows that the corresponding solution operators are monotone, which implies that a reduced system of equations has a unique solution.

Here we recall the historically earlier approach of [12] and [29], based on a classical convex min-cost flow problem. This convex min-cost flow problem and its dual turn out to be useful for various purposes. Let $\Psi : \mathbb{R} \to \mathbb{R}$ with

$$\Psi(x) := \int_0^x \psi(\xi) \, \mathrm{d}\xi,$$

which is strictly convex as ψ is strictly increasing. The convex optimization problem is given as

$$\min_x \quad \sum_{a \in A} \beta_a \Psi(x_a) \tag{7a}$$

$$\text{s.t.} \quad \sum_{a \in \delta^{\text{out}}(v)} x_a - \sum_{a \in \delta^{\text{in}}(v)} x_a = b_v, \quad v \in V. \tag{7b}$$

Notice that the objective function (7a) is strictly convex. For Lagrangean multipliers π_v, $v \in V$, the corresponding Lagrange function reads

$$\begin{aligned} L(x, \pi) &= \sum_{a \in A} \beta_a \Psi(x_a) + \sum_{a=(u,v) \in A} x_a(\pi_v - \pi_u) + \sum_{v \in V} b_v \pi_v \\ &= \sum_{a=(u,v) \in A} \big(\beta_a \Psi(x_a) + x_a(\pi_v - \pi_u)\big) + \sum_{v \in V} b_v \pi_v. \end{aligned}$$

The KKT conditions thus yield (1) as well as the flow conservation constraints (7b). Fixing one potential makes the solution unique, which gives Theorem 3.2.

For testing feasibility, the restriction that we do not consider potential and flow bounds is not an obstacle due to the following observation, which follows directly from Constraint (3b).

Lemma 3.3 (Szabó [38]). *Let G be a passive potential network with node balances $b \in \mathbb{R}^V$. If (x, π) is a potential-based b-flow, then $(x, \pi + c\,\mathbb{1})$ is a potential-based b-flow for every constant $c \in \mathbb{R}$.*

So, to check feasibility in a passive potential network, one can first ignore potential bounds and compute a potential-based b-flow. If the flow bounds are violated, there exists no feasible potential-based b-flow. Otherwise, one checks whether the potentials can be shifted such that the flow becomes feasible for the potential bounds.

For future reference, we state the following.

Lemma 3.4 ([29]). *Consider a potential network and let the potential function be positively homogeneous, i.e., $\psi(x) = \alpha \operatorname{sgn}(x)\,|x|^r$ for some $\alpha > 0$ and $r > 0$. Then, the Lagrange dual $\max_\pi \min_x L(x, \pi)$ of (7) is given by*

$$\max_\pi \left(\sum_{v \in V} b_v \pi_v - \sum_{a=(u,v) \in A} \frac{r}{r+1} \frac{|\pi_u - \pi_v|^{\frac{r+1}{r}}}{(\beta_a\,\alpha)^{\frac{1}{r}}} \right). \tag{8}$$

Proof. Notice that for fixed π, taking the partial derivatives of $L(x, \pi)$ w.r.t. the flow variables x_a, the inner minimum over these variables is attained by x^* satisfying, for each arc $a = (u, v) \in A$,

$$\beta_a \psi(x_a^*) + \pi_v - \pi_u = 0 \qquad \text{or, equivalently,} \qquad x_a^* = \psi^{-1}\left(\frac{\pi_u - \pi_v}{\beta_a}\right).$$

Thus, the Lagrange dual can be rewritten as

$$\max_\pi \sum_{a=(u,v) \in A} \left(\beta_a \Psi\left(\psi^{-1}\left(\frac{\pi_u - \pi_v}{\beta_a}\right)\right) + \psi^{-1}\left(\frac{\pi_u - \pi_v}{\beta_a}\right)(\pi_v - \pi_u) \right) + \sum_{v \in V} b_v \pi_v.$$

For positively homogeneous ψ, we have

$$\Psi(x) = \frac{\alpha}{r+1}\,|x|^{\,r+1} \qquad \text{and} \qquad \psi^{-1}(y) = \operatorname{sgn}(y)\left(\frac{|y|}{\alpha}\right)^{\frac{1}{r}}.$$

Thus, the first sum of the Lagrange dual's objective is

$$\sum_{a=(u,v) \in A} \left(\frac{\alpha\,\beta_a}{r+1} \left(\frac{|\pi_u - \pi_v|}{\alpha\,\beta_a}\right)^{\frac{r+1}{r}} + \operatorname{sgn}(\pi_u - \pi_v)\left(\frac{|\pi_u - \pi_v|}{\alpha\,\beta_a}\right)^{\frac{1}{r}} (\pi_v - \pi_u) \right).$$

Simplification yields (8). □

For later use, we note that, using the notation of the proof of Lemma 3.4, the Lagrange dual can also be written as

$$\max_\pi \sum_{v \in V} b_v \pi_v - r \sum_{a \in A} \beta_a\,\Psi(x_a^*). \tag{9}$$

This duality is discussed in Rockafellar [32] in more detail and more generality.

Returning to the s-t-flow case, we can show that in this case passive potential networks with a positively homogeneous potential function can equivalently be represented by a single line. To make the statement of this result simpler, we use the following shorthand notation. We say that a triple $(x_{st}, \bar{\pi}_s, \bar{\pi}_t)$ *can be extended to a potential-based flow* if there exists a potential-based s-t-flow (x, π) with flow value $\operatorname{val}(x, \pi) = x_{st}$ such that $\pi_s = \bar{\pi}_s$ and $\pi_t = \bar{\pi}_t$. We note that, due to Theorem 3.2, these extensions are unique.

Lemma 3.5. *Let G be a homogeneous passive potential network of order $r > 0$ with two distinguished nodes $s, t \in V$. Assume (x_{st}, π_s, π_t) with $x_{st} \neq 0$ can be extended to a potential-based s-t-flow. Then, for any $\lambda \in \mathbb{R}$, the triple $\big(\lambda\, x_{st},\, \operatorname{sgn}(\lambda)\, |\lambda|^r\, \pi_s,\, \operatorname{sgn}(\lambda)\, |\lambda|^r\, \pi_t\big)$ can be extended to a potential-based s-t-flow.*

Proof. The stated result is an immediate consequence of Observation 3.1. □

The nice characterization of the space of solutions of the preceding lemma immediately breaks down as soon as one introduces new element types (even non-homogeneous passive elements) or allows multiple sources or sinks. We can, however, prove that a homogeneous passive network with a single source and a single sink has the same behavior as a single line, which has been already observed in different settings, e.g., for water networks in Burgschweiger et al. [10].

Theorem 3.6. *Let G be a homogeneous passive potential network of order $r > 0$ with two distinguished nodes $s, t \in V$. Then, there exists a constant β_{st} such that a triple (x_{st}, π_s, π_t) can be extended to a potential-based s-t-flow if and only if*

$$x_{st} = \psi^{-1}\big((\pi_s - \pi_t)/\beta_{st}\big). \tag{10}$$

Proof. Consider a potential-based s-t-flow (x', π') of value 1, that is, an optimal solution x' to (7) together with a corresponding optimal dual solution π' to (8), for node balances $b \in \mathbb{R}^V$ with $b_s = -b_t = 1$ and $b_v = 0$ for all $v \in V \setminus \{s, t\}$. By Lemma 3.3, i.e., shifting the node potentials by $-\pi'_t$, we may assume that $\pi'_t = 0$. Thus, the triple $(1, \pi'_s, 0)$ can be extended to the potential-based s-t-flow (x', π').

We define $\beta_{st} := \pi'_s/\psi(1)$ and show that a triple (x_{st}, π_s, π_t) can be extended to a potential-based s-t-flow if and only if Equation (10) holds. Making use of Lemma 3.3 once more, we may assume $\pi_t = 0$. Now let

$$\tilde{\pi}_s := \psi(x_{st})\, \beta_{st} = \operatorname{sgn}(x_{st})\ |x_{st}|^{\,r}\, \psi(1)\, \beta_{st} = \operatorname{sgn}(x_{st})\ |x_{st}|^{\,r}\, \pi'_s.$$

By Lemma 3.5 the triple $(x_{st}, \tilde{\pi}_s, 0)$ can be extended to a potential-based s-t-flow. Moreover, due to Theorem 3.2, $\tilde{\pi}_s$ is the unique potential on s that allows this extension. Hence, $(x_{st}, \pi_s, 0)$ can be extended if and only if $\pi_s = \tilde{\pi}_s$. The latter condition is equivalent to (10). □

We next show that for a homogeneous passive potential network, the s-t-MaxFlow problem can be solved efficiently.

Theorem 3.7. *Let G be a homogeneous passive potential network with upper and lower bounds on the node potentials and flows. Then, there is an interval $I_G \subseteq \mathbb{R}$ such that a feasible potential-based s-t-flow of value x_{st} exists if and only if $x_{st} \in I_G$. Moreover, I_G can be efficiently computed.*

This theorem implies that the value of an s-t-MaxFlow and a corresponding feasible potential-based s-t-flow can be efficiently computed.

Proof. Let x be a (not necessarily feasible) potential-based s-t-flow of value 1. It follows from Observation 3.1 and Theorem 3.2 that in any potential-based s-t-flow of value λ, the flow on line $a \in A$ is equal to $\lambda\, x_a$. In particular, a potential-based s-t-flow of value λ obeys the flow bounds $[\underline{x}_a, \overline{x}_a]$ on a if and only if $\lambda \in [\underline{x}_a/x_a, \overline{x}_a/x_a] =: I_a$. Therefore, there exists a potential-based s-t-flow of value λ obeying all flow bounds if and only if $\lambda \in \bigcap_{a \in A} I_a =: I_A$. Notice that the interval I_A can be computed in linear time.

We next deal with the bounds on node potentials. For this purpose, let π with $\pi_s = 0$ be the potential corresponding to the potential-based s-t-flow x of value 1. By Observation 3.1 and Theorem 3.2, a potential-based s-t-flow of value λ obeys the bounds $[\underline{\pi}_v, \overline{\pi}_v]$ on v's node potential, for some node $v \in V$, if and only if the potential of the source node s is in $[\underline{\pi}_v - \lambda\, \pi_v, \overline{\pi}_v - \lambda\, \pi_v] =: J_v$. In particular, there is a potential-based s-t-flow of value λ obeying all bounds on node potentials if and only if $\bigcap_{v\in V} J_v$ is non-empty, that is, if and only if

$$\max_{v\in V} (\underline{\pi}_v - \lambda\,\pi_v) \le \min_{v\in V} (\overline{\pi}_v - \lambda\,\pi_v). \tag{11}$$

Notice that inequality (11) is fulfilled for some value $\lambda \in \mathbb{R}$ if and only if λ is contained in an interval I_V whose borders can be determined in time $O(|V|^2)$.

Summarizing, there is a feasible potential-based flow of value λ if and only if $\lambda \in I_A \cap I_V$ and the interval $I_A \cap I_V$ can be efficiently computed. □

4. Series-Parallel Graphs

Series-parallel graphs often appear in applications, either directly or as substructures, see Section 6. In this section, we exploit their favorable properties for positive results on passive and active potential-based networks. We describe them in their directed form by a constructive characterization loosely following Eppstein [14].

Definition 4.1. An *s-t-series-parallel graph (SPG) G* is a directed graph with two distinguished nodes s and t, called the *source* and *sink* of G, that can be created from two SPGs using the following two operations:

- *Parallel composition*: Let X be an s_1-t_1-SPG and Y be an s_2-t_2-SPG. The parallel composition of X and Y is an s-t-SPG P defined by merging the sources and sinks of X and Y, respectively:
$$V(P) := (V(X) \cup V(Y) \cup \{s,t\}) \setminus \{s_1, s_2, t_1, t_2\},$$
$$A(P) := (A(X) \cup A(Y) \cup A') \setminus \bigcup_{v\in\{s_1,s_2,t_1,t_2\}} \delta(v)$$
with
$$\begin{aligned} A' := \;& \{\,(s,v) \mid (s_1,v) \in A(X) \text{ or } (s_2,v) \in A(Y)\} \\ & \cup \{\,(v,s) \mid (v,s_1) \in A(X) \text{ or } (v,s_2) \in A(Y)\} \\ & \cup \{\,(t,v) \mid (t_1,v) \in A(X) \text{ or } (t_2,v) \in A(Y)\} \\ & \cup \{\,(v,t) \mid (v,t_1) \in A(X) \text{ or } (v,t_2) \in A(Y)\}\,. \end{aligned}$$
- *Series composition*: Let X be an s_1-t_1-SPG and Y be an s_2-t_2-SPG. The series composition of X and Y is an s_1-t_2-SPG S defined by merging the sink of X with the source of Y into a node u:
$$V(S) := (V(X) \cup V(Y) \cup \{u\}) \setminus \{t_1, s_2\},$$
$$A(S) := (A(X) \cup A(Y) \cup A') \setminus \bigcup_{v\in\{t_1,s_2\}} \delta(v)$$
with
$$\begin{aligned} A' := \;& \{\,(u,v) \mid (t_1,v) \in A(X) \text{ or } (s_2,v) \in A(Y)\} \\ & \cup \{\,(v,u) \mid (v,t_1) \in A(X) \text{ or } (v,s_2) \in A(Y)\}\,. \end{aligned}$$

Finally, the graph consisting of two nodes v, w and a single arc (v,w) is a v-w- and w-v-SPG.

Given an SPG, it is possible to reverse-engineer a sequence of parallel and series compositions that was used for its construction. This is referred to as *series-parallel decomposition* and results in a tree with series- and parallel compositions as inner nodes and single arcs as the leaves. This decomposition can be found in linear time (see Takamizawa et al. [39]) and allows for faster algorithms for many problems like maximum independent set or maximum matching, which can be solved in linear time on SPGs, see [39].

Generalized series-parallel graphs (see Korneyenko [25]) are an extension of series-parallel graphs that retains the decomposability of series-parallel graphs. In addition to the series- and parallel compositions, a *tree composition* is allowed:

- *Tree composition.* Let X be an s_1-t_1-SPG and Y be an s_2-t_2-SPG. The tree composition of X and Y is an s-t_2-SPG T defined by merging the sources of X and Y into a source s:
$$V(T) := (V(X) \cup V(Y) \cup \{s\}) \setminus \{s_1, s_2\},$$
$$A(T) := (A(X) \cup A(Y) \cup A') \setminus \bigcup_{v \in \{s_1, s_2\}} \delta(v)$$
with
$$A' := \{(s, v) \mid (s_1, v) \in A(X) \text{ or } (s_2, v) \in A(Y)\} \cup \{(v, s) \mid (v, s_1) \in A(X) \text{ or } (v, s_2) \in A(Y)\}.$$

In particular, all trees are generalized series-parallel graphs.

An SPG decomposition can be used to presolve a given potential-based network, as the following result shows.

Lemma 4.2.

(1) Consider two serial lines $a_1 = (u, v)$ and $a_2 = (v, w)$ with $b_v = 0$ and $|\delta^{in}(v)| = |\delta^{out}(v)| = 1$. Then, this serial combination can be equivalently replaced by the line $a = (u, w)$ with $\beta_a = \beta_{a_1} + \beta_{a_2}$.

(2) Consider two parallel lines a_1 and a_2 between nodes u and v with $\beta_{a_1}, \beta_{a_2} > 0$. Then, this parallel combination can be equivalently replaced by the line $a = (u, v)$ with
$$\beta_a = \frac{\beta_{a_1}\beta_{a_1}}{\left(\sqrt[r]{\beta_{a_1}} + \sqrt[r]{\beta_{a_1}}\right)^r}.$$

Proof.

(1) The given situation leads to the model
$$\pi_u - \pi_v = \beta_{a_1}\psi(x_{a_1}) \quad \text{and} \quad \pi_v - \pi_w = \beta_{a_2}\psi(x_{a_2}).$$
Since $b_v = 0$ and $|\delta^{\text{in}}(v)| = |\delta^{\text{out}}(v)| = 1$, we know that $x_{a_1} = x_{a_2} =: x_a$ and adding the above two equalities yields
$$\pi_u - \pi_w = \pi_u - \pi_v + \pi_v - \pi_w = (\beta_{a_1} + \beta_{a_2})\psi(x_a),$$
which proves the first result.

(2) For the two parallel arcs, we know that
$$\pi_u - \pi_v = \beta_{a_1}\psi(x_{a_1}) = \beta_{a_2}\psi(x_{a_2}) \tag{12}$$
holds. Thus, we have to find a β_a that satisfies
$$\pi_u - \pi_v = \beta_a\psi(x_{a_1} + x_{a_2}). \tag{13}$$
To this end, define $z := x_{a_1} + x_{a_2}$ and let $\lambda \in [0, 1]$ be such that $x_{a_1} = (1 - \lambda)\, z$, $x_{a_2} = \lambda\, z$. Using this re-parameterization, (12), and the homogeneity of ψ yields the equation
$$\frac{\beta_{a_1}}{\beta_{a_2}} = \frac{\psi(x_{a_2})}{\psi(x_{a_1})} = \frac{\psi(\lambda\, z)}{\psi((1 - \lambda)\, z)} = \frac{\lambda^r}{(1 - \lambda)^r}.$$
Solving this equation for λ yields
$$\lambda = \frac{\sqrt[r]{\beta_{a_1}/\beta_{a_2}}}{1 + \sqrt[r]{\beta_{a_1}/\beta_{a_2}}}. \tag{14}$$
Equations (12) and (13) lead to
$$\beta_{a_1}(1 - \lambda)^r\psi(z) + \beta_{a_2}\lambda^r\psi(z) = 2\,\beta_a\,\psi(z)$$

and solving for β_a, we obtain

$$\beta_a = \tfrac{1}{2}\left(\beta_{a_1}(1-\lambda)^r + \beta_2\lambda^r\right).$$

Finally, using (14) and a simple calculation proves the statement. □

For instance, β_a for the surrogate line a for two parallel lines a_1, a_2 in gas networks satisfies

$$\beta_a = \frac{\beta_{a_1}\beta_{a_2}}{\left(\sqrt{\beta_{a_1}} + \sqrt{\beta_{a_2}}\right)^2}.$$

For lossless DC power flow, we get

$$\beta_a = \frac{\beta_{a_1}\beta_{a_2}}{\beta_{a_1} + \beta_{a_2}},$$

which is equivalent to the well-known parallel resistance formula

$$\frac{1}{\beta_a} = \frac{1}{\beta_{a_1}} + \frac{1}{\beta_{a_2}}.$$

This application of SPGs for DC power flow was already presented by Duffin [13]. The specific formula for water flow has also been studied in the literature, see [10].

The series-parallel decomposition of SPGs together with Lemma 4.2 reveals that passive s-t-SPGs can be replaced equivalently by a graph with a single surrogate line connecting s and t. The existence of such a single surrogate line is already known from Theorem 3.6. However, for SPGs we obtain an explicit way of computing the surrogate line without the need to solve a potential-based flow problem.

Theorem 4.3. *For a passive s-t-SPG G, we can reduce G to a single arc with $\mathcal{O}(|A(G)|)$ applications of Lemma 4.2.*

Proof. The binary decomposition tree of G can be computed in $\mathcal{O}(|A(G)|)$ and has size $\mathcal{O}(|A(G)|)$, see, e.g., [39]. We now traverse this binary decomposition tree from the leaves to the root. If we encounter an inner node during this traversal that corresponds to a serial composition we apply Part (1) of Lemma 4.2; if we encounter an inner node during this traversal that corresponds to a parallel composition we apply Part (2). □

We can exploit the series-parallel structure for active networks, too, as the next theorem shows.

Theorem 4.4. *Let G be an s-t-SPG that consists of internally node-disjoint s-t-paths that contain switches and lines. Then, we can transform the question whether we can send X units of flow from s to t for fixed π_s, π_t into a* SUBSETSUM *problem (if ψ^{-1} and X are rational). The* SUBSETSUM *problem is:*

> SUBSETSUM
> ***Input:*** *Numbers $k_1, k_2, \ldots, k_n, X \in \mathbb{Z}$.*
> ***Problem:*** *Is there a set $I \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in I} k_i = X$?*

Proof. First, we notice that there are only two meaningful cases for each s-t-path: either all switches are on, in which case we can contract the switch arcs and use Theorem 4.3 to obtain a single arc for the path, or at least one switch is off, in which case the whole path transports no flow at all. Thus, each path P can be replaced by a switch followed by a line whose parameter β_P can be computed using Theorem 4.3.

Let there be n s-t-paths with x_i being the flow on path $i \in \{1, \dots, n\}$, β_i the constant of path i, and $y_i \in \{0, 1\}$ denotes whether all switches are open ($y_i = 1$) or not ($y_i = 0$). Then, we have that the flow sent from s to t is

$$b_s := \sum_{i=1}^{n} x_i = \sum_{i=1,\dots,n:\, y_i=1} \psi^{-1}\left(\frac{\pi_s - \pi_t}{\beta_i}\right).$$

Our objective is to determine whether there is a variable assignment such that $X = b_s$ for a given X, which can be reformulated as a SUBSETSUM problem: Is there a subset $I \subseteq \{1, \dots, n\}$ such that

$$\sum_{i \in I} \psi^{-1}\left(\frac{\pi_s - \pi_r}{\beta_i}\right) = X? \tag{15}$$

Since the SUBSETSUM problem is defined for integer inputs, we have to scale X and $\psi^{-1}\left((\pi_s - \pi_r)/\beta_i\right)$ values so that they are become integer, which is always possible due to the assumption of ψ^{-1} and X being rational. □

The advantage of this reduction is that SUBSETSUM problem is well studied. Possible solutions are an $O(2^{n/2})$ algorithm due to Horowitz and Sahni [19], an $O(n \max_{i \in I} \{k_i\})$ algorithm by Pisinger [30], and an $\tilde{O}(\sqrt{n}\, X)$ algorithm due to Koiliaris and Xu [24].

We can also extend our results to generalized series-parallel networks with multiple sources and sinks, as long as the sources and sinks are conveniently placed with regard to the structure of the network.

Theorem 4.5. *Let G be a passive potential network with balances b such that G can be decomposed into s-t-SPGs $G_1, \dots, G_\ell$ in which every node with $b_v \neq 0$ is the source and/or the sink of an SPG G_i, $i = 1, \dots, \ell$. Then, we can reduce G into a tree.*

Proof. Let G_i, $i = 1, \dots, \ell$, be the decomposition of G into SPGs such that all nodes v of G_i that are neither source nor sink of G_i have $b_v = 0$. Then, we can apply Theorem 4.3 to reduce G_i into a single arc. Performing this reduction for all G_i reduces G to a tree. □

5. Active Networks: Monotonicity and Multiple Sources and Sinks

In this section, we consider active networks and relate the previous results to the problem of minimizing the *power-loss*

$$\sum_{v \in V} b_v\, \pi_v = \sum_{a=(u,v) \in A} (\pi_u - \pi_v)\, x_a$$

for some potential-based flow (x, π) with respect to the balance vector $b \in \mathbb{R}^V$ with $\sum_{v \in V} b_v = 0$. In the s-t-case, the power-loss simply reads $b_s(\pi_s - \pi_t)$.

Theorem 5.1. *Consider a potential-based network G with homogeneous potential function and assume that the potential and flow bounds are infinite. Then, for a given balance vector $b \in \mathbb{R}^V$, $\sum_{v \in V} b_v = 0$, the power-loss of a feasible potential-based flow is minimized if all switches are on.*

Proof. Let $\psi(x) = \alpha \operatorname{sgn}(x)\, |x|^r$ for some $r > 0$ and first fix the decisions y for the switches to be off. Consider the primal problem (7) and the dual problem (8) and observe that strong duality holds. Let z^* be the optimal value of these two problems. Then, by (9), we obtain the following for the optimal solution (x^*, π^*):

$$z^* = \sum_{v \in V} b_v\, \pi_v^* - r \sum_{a \in A} \beta_a \Psi(x_a^*).$$

By the primal problem, the last sum is equal to $r\,z^*$, which yields $\sum_{v\in V} b_v\pi_v = (r+1)z^*$. If we turn on switches, then clearly z^* does not increase, since this increases the degree of freedom (some flows are allowed to take nonzero values afterward). This shows that the power-loss is not increasing when turning on switches. □

Remark 5.2.

(1) We note that Calvert and Keady [11, Theorem 1] showed that the power-loss is monotone with respect to the change of one β_a, $a \in A$. By observing that $\beta_a = 0$ corresponds to an open switch, this can be used to show Theorem 5.1, but the above proof is simpler.
(2) For the s-t-case, Theorem 5.1 implies that the problem to minimize power-loss given a fixed flow value and without flow and potential bounds is easy (obtained by turning all switches on). However, it is known from the literature that the problem becomes NP-hard, if potential bounds are present, see Section 2.4 for a discussion.

Using similar arguments as in Theorem 5.1, we see the following for the s-t-case:

Corollary 5.3. *The value β_{st} of Theorem 3.6 does not increase when turning on switches. Moreover, if $\pi_s - \pi_t > 0$ is fixed, then the maximal flow value is non-decreasing with respect to turning on switches.*

Remark 5.4. Corollary 5.3 implies that no equivalent to the so-called Braess-paradox exists, see Braess [6]. In our situation, this would mean that we can increase the flow value by turning off switches. In fact, the main purpose of the paper [11] is to show a converse to the monotonicity property mentioned above: Calvert and Keady showed that if the power-loss is monotone with respect to the change of one β_a, then the network has to be homogeneous and all $\psi(x)$ have the same order. Indeed, note that it is crucial for the proof of Theorem 5.1 that all $\psi(x)$ are the same. If this condition is violated, for instance, by modeling capacities, Braess paradoxes may appear. In this case, the network has to contain a diamond graph (Wheatstone bridge in the electrical network community) as an induced subgraph. This implies that the graph is not series-parallel.

Notice that we can transform active networks with multiple sources and sinks into active networks with only a single sink, if we are willing to introduce a small error and flow bounds are present. We describe this procedure for transferring balance from a sink to a super-sink; an analogous procedure can be applied to shift balance from a source to a super-source.

Let t be the sink we are considering and ω our super-sink. Let $\hat{\pi}_t$ be a lower bound for the potential of t in any feasible flow. We assume that $\hat{\pi}_t > 0$, since otherwise we may shift potentials as discussed in Lemma 3.3. Without loss of generality, we assume that $\pi_\omega = 0$. We now add an intermediate node v_t and a line-arc $a_t = (t, v_t)$ with flow bounds $\overline{x}_{a_t} = \underline{x}_{a_t} = -b_t$ and $\beta_{a_t} = -\hat{\pi}_t/(2b_t)$. This ensures that we can send $-b_t$ flow along this arc if $\pi_{v_t} = \pi_t - \hat{\pi}_t/2$, which is always possible if there exists a feasible potential-based flow in the original network.

In order to send $-b_t$ flow from v_t to ω, we need an arc with parameter β, since we always need to send $-b_t$ flow, but we do not know the potential of v_t beforehand and the potential of ω is fixed. We can simulate this by adding paths $P_1, \dots, P_k$ between v_t and ω that consist of a switch followed by a line. For each such path, we vary the arc-constant of the line arc—the introduced error then depends on the number of used paths (and the spread of the arc-constants in the line-arcs of each path).

6. Network Reduction Techniques

The results of Section 3 and 4, especially Lemma 4.2, suggest graph reduction methods for potential-based networks. These methods are described in this section and we present and discuss numerical results for water, power, and gas networks. We start by describing the general ideas. Specific adaptions for the concrete network types are discussed in the corresponding sections.

For our reductions, we assume that we are given demands and supplies $b = (b_v)_{v \in V} \in \mathbb{R}^V$ and, possibly, lower and upper bounds on the potentials on all nodes of the network $G = (V, A)$. We assume that for supply nodes $v \in V$ we have $b_v > 0$ and that $b_v < 0$ holds for demand nodes $v \in V$. The remaining set of nodes v with $b_v = 0$ are called inner nodes in the following.

Our overall graph reduction consists of two ingredients: (1) the reduction of leaves and (2) the reduction of serial and parallel arcs. We note that our reductions are independent of the actual flow rates. Therefore, they do not depend on a specific balance vector b, but only on the specification whether a node is a demand or a supply node.

We describe both techniques in the following for passive networks. The only networks considered here that have active elements are gas networks. The required adaptions are thus described in Section 6.3, where we also discuss the numerical results for gas transport networks.

Leaf Reduction. For every leaf $v \in V$, we know the flow value $x_a = \pm b_v$ of its unique incident arc $a \in \delta(v)$ since a is a bridge. Without loss of generality, let $a = (u, v)$. In this situation, we update $b_u \leftarrow b_u + b_v$ and delete the arc $a = (u, v)$ and the node v.

If we are additionally given potential bounds $\underline{\pi}_u, \overline{\pi}_u, \underline{\pi}_v, \overline{\pi}_v$, we update the potential bounds $\underline{\pi}_u, \overline{\pi}_u$ as follows:

$$\underline{\pi}_u \leftarrow \max \{\underline{\pi}_u,\ \underline{\pi}_v + \beta_a \psi(x_a)\}, \quad \overline{\pi}_u \leftarrow \min \{\overline{\pi}_u,\ \overline{\pi}_v + \beta_a \psi(x_a)\}.$$

Let $G' = (V \setminus \{v\}, A \setminus \{a\})$ be the reduced network. It is immediately clear that there exists a feasible flow for the network G if and only if there is a feasible solution for the reduced network G'. Obviously, the leaf reduction can be iteratively applied until no more leaf nodes are present in G'.

Reduction of Serial and Parallel Arcs. By Lemma 4.2 we know that we can replace parallel arcs a_1 and a_2 from u to v by a single new arc $a = (u, v)$ as well as serial arcs $a_1 = (u, v)$, $a_2 = (v, w)$ that are connected by an inner degree-2 node v of balance 0 with a new arc $a = (u, w)$. In both situations, we can apply Lemma 4.2 to compute an equivalent arc parameter β_a.

As before, these reductions can be applied iteratively. If this iteration terminates, the reduced graph has no more degree-2 nodes with balance 0 and no more parallel arcs. In particular, if we apply both the leaf reduction and the reduction of serial and parallel arcs to a passive generalized s-t-SPG, we end up with an equivalent network $G' = (\{s\}, \emptyset)$ with a single node. Note also that the reduction technique for serial and parallel arcs cannot eliminate supply or demand nodes—which is possible for the leaf reduction method.

We now turn to the discussion of the results. All graph reduction techniques have been implemented in Java 1.8.0_102 and all computations have been performed on an Intel Core i7-4510U with two 2 GHz cores and 8 GB RAM. Our focus here lies on the strength of the reduction techniques in terms of the reduction of the graph size. The used measure is thus the number of arcs before and after the reduction. All computation times do not include the computation of new arc parameters β or

TABLE 1. Results of the network reduction techniques on water network instances

Instance	$\lvert V\rvert$	$\lvert V_\pm\rvert$	$\lvert A\rvert$	$\lvert V'\rvert$	$\lvert A'\rvert$	L	P	S	t	ρ
shamir	7	7	8	6	7	1	0	0	0.0007	13 %
NewYork	20	20	21	16	17	4	0	0	0.0004	19 %
hanoi	32	32	34	25	27	7	0	0	0.0014	21 %
blacksburg	31	31	35	23	27	8	0	0	0.0035	23 %
fossiron	37	37	58	36	57	1	0	0	0.0008	2 %
pescara	71	64	99	64	91	7	1	0	0.0009	8 %
modena	272	245	317	268	313	4	0	0	0.0013	1 %

updating potential bounds. As a consequence of this, we do not have to deal with roots in this section.

6.1. **Water Networks.** We start by presenting network reduction results for seven water transport networks used in the publications Bragalli et al. [7] and Bragalli et al. [8, 9] on mixed-integer nonlinear models (MINLPs) for the optimal design of water distribution networks. All considered water networks are passive, i.e., all arcs are pipes. The networks shamir, NewYork, hanoi, and blacksburg are taken from the literature, whereas fossiron, pescara, and modena are reduced versions of medium-sized Italian cities.

The statistics for the network instances as well as for the applied reductions are given in Table 1. The number of nodes and arcs of the network is denoted by $|V|$ and $|A|$, respectively, and $|V_\pm|$ denotes the number of entries and exits in the network. The number of nodes and arcs of the reduced network is given in the columns $|V'|$ and $|A'|$, respectively. Leaf reductions as well as the reduction of parallel and serial arcs are captioned with "L", "P", and "S". Finally, t denotes runtimes (in seconds), and ρ is the percentage reduction of arcs.

The reduction results for the tested water network instances vary from 1 % to 23 % reduced arcs of the original network. We will later see that the same reduction techniques yield much better results for power and, especially, gas transport networks. For five out of seven water network instances, all nodes are either supply or demand nodes. This leads to the fact that the serial reduction, which requires serial arcs connected by an inner degree-2 node of balance 0, is never applied. Our experience with power and gas networks additionally shows that parallel arcs often arise from a sequence of serial reductions. Thus, there are almost no parallel reductions as well. Only the pescara instance has two parallel arcs that are merged. The leaf reduction thus is more or less the only reduction technique that is applied for water networks.

As an exemplary instance, Figure 1 depicts the fossiron network, which shows (i) that there are no serial arcs that can be merged since all nodes are supply or demand nodes, (ii) that there are no parallel arcs to be merged, and (iii) that there is only one leaf.

Fortunately, it turns out that our reduction techniques can be implemented quite efficiently: The total runtime for all water networks is significantly below 1 s.

6.2. **Power Networks.** We now present the results of our reduction techniques applied to 30 power network instances taken from the MATPOWER library of Zimmerman et al. [40]. These networks vary strongly in their size, from very small 4-arc instances to huge networks with more than 20 000 arcs. The results are given in Table 2 in the same way as in Table 1 for water networks. The reduction of arcs varies in the range of 0 % to 49 %. Since the instances from [40] are frequently used for solving non-convex optimal power flow problems, this reduction in the number

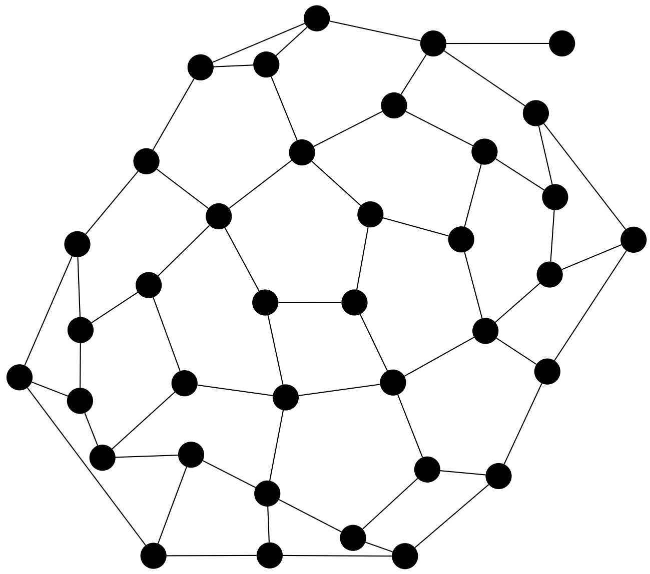

FIGURE 1. Schematic plot of the water network fossiron

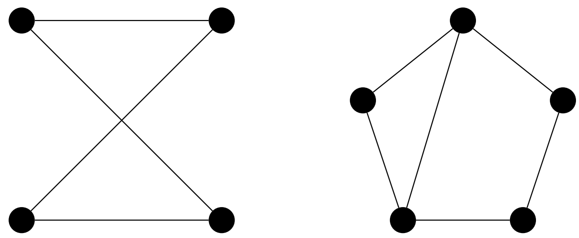

FIGURE 2. Schematic plots of the power networks case4gs (left) and case5 (right)

of arcs would roughly correspond to a reduction of the size (in terms of variables and constraints) of a non-convex problem up to almost 50 %, which would typically lead to significantly faster runtimes.

The most successful technique is the leaf reduction method, which can be applied for all but three instances (case4gs, case5, case6ww). In addition, for all instances with more than 245 arcs, our method also finds parallel arcs that can be reduced. This was not the case for the water instances and thus indicates an important topological difference between the considered water and power networks. Clearly, this difference is also the reason why the overall reduction is stronger on the power than on the water networks. However, we also observe some instances with a small ($\leq$ 10 % or even 0 %) number of reductions. These cases are mainly very small networks like case4gs and case5 in Figure 2. In these cases, all nodes are either supply or demand nodes and the network does not have any leaves, which explains the missing reductions. For the larger networks, the amount of supply or demand nodes reduces and thus our techniques can be applied more often. Since the MATPOWER test problems are designed for researchers and educators, they do not correspond to historically grown real-world networks, but academic ones. This might be the reason that these networks do not contain any serial arcs that are connected with inner degree-2 nodes. Thus, the serial reduction is never applied on this test set.

Table 2. Results of the network reduction techniques on power network instances

Instance	$\lvert V\rvert$	$\lvert V_\pm\rvert$	$\lvert A\rvert$	$\lvert V'\rvert$	$\lvert A'\rvert$	L	P	S	t	ρ
case4gs	4	4	4	4	4	0	0	0	0.0002	0 %
case5	5	5	6	5	6	0	0	0	0.0002	0 %
case9	9	6	9	6	6	3	0	0	0.0004	33 %
case9Q	9	6	9	6	6	3	0	0	0.0006	33 %
case9target	9	6	9	6	6	3	0	0	0.0005	33 %
case6ww	6	6	11	6	11	0	0	0	0.0002	0 %
case14	14	13	20	13	19	1	0	0	0.0003	5 %
case33bw	33	33	37	32	36	1	0	0	0.0005	3 %
case24ieeerts	24	20	38	23	33	1	4	0	0.0002	13 %
case30	30	24	41	27	38	3	0	0	0.0005	7 %
case30pwl	30	24	41	27	38	3	0	0	0.0003	7 %
case30Q	30	24	41	27	38	3	0	0	0.0008	7 %
caseieee30	30	24	41	27	38	3	0	0	0.0008	7 %
case39	39	29	46	28	35	11	0	0	0.0004	24 %
case57	57	42	80	56	77	1	2	0	0.0002	4 %
case118	118	108	186	109	170	9	7	0	0.0110	9 %
case89pegase	89	41	210	72	189	17	4	0	0.0006	10 %
caseillinois200	200	157	245	128	173	72	0	0	0.0008	29 %
case300	300	225	411	213	322	87	2	0	0.0019	22 %
case145	145	61	453	132	409	13	31	0	0.0017	10 %
case1354pegase	1354	881	1991	730	1086	624	281	0	0.0400	45 %
case1888rte	1888	1149	2531	886	1306	1002	223	0	0.0088	48 %
case1951rte	1951	1232	2596	895	1319	1056	221	0	0.0086	49 %
case2383wp	2383	1827	2896	1733	2236	650	10	0	0.0073	23 %
case2736sp	2736	2057	3504	2395	3154	341	9	0	0.0129	10 %
case2737sop	2737	2058	3506	2396	3156	341	9	0	0.0082	10 %
case2746wop	2746	2016	3514	2392	3151	354	9	0	0.0082	10 %
case2746wp	2746	2029	3514	2393	3152	353	9	0	0.0069	10 %
case3012wp	3012	2277	3572	2301	2855	711	6	0	0.0101	20 %
case3120sp	3120	2311	3693	2385	2949	735	9	0	0.0073	20 %
case2848rte	2848	1694	3776	1382	1976	1466	334	0	0.0080	48 %
case2868rte	2868	1748	3808	1389	1992	1479	337	0	0.0105	48 %
case3375wp	3374	2476	4161	2511	3205	863	93	0	0.0100	23 %
case2869pegase	2869	1815	4582	1989	3088	880	614	0	0.0160	33 %
case6468rte	6468	3791	9000	3796	5393	2672	935	0	0.0218	40 %
case6470rte	6470	3856	9005	3783	5379	2687	939	0	0.0264	40 %
case6495rte	6495	3871	9019	3771	5360	2724	935	0	0.0220	41 %
case6515rte	6515	3906	9037	3773	5362	2742	933	0	0.0236	41 %
case9241pegase	9241	5873	16049	7374	12340	1867	1842	0	0.0587	23 %
case13659pegase	13659	9135	20467	7374	12340	6285	1842	0	0.1623	40 %

Finally, regarding runtimes we see that our algorithm requires less than 1 s for all instances.

6.3. **Gas Networks.** Finally, we discuss the results of our network reduction techniques applied to three small- to medium-scale academic, but realistic gas networks and two large-scale real-world gas networks. The academic networks are freely available online (see Humpola et al. [21]) and the real-world instances are the low- and high-calorific gas networks of the German gas transport company Open Grid Europe GmbH.[1]

[1] https://www.open-grid-europe.com

TABLE 3. Results of the network reduction techniques on gas network instances

Instance	$\lvert V\rvert$	$\lvert V_{\pm}\rvert$	$\lvert A\rvert$	$\lvert V'\rvert$	$\lvert A'\rvert$	L	P	S	t	ρ
GasLib-40	40	32	45	29	34	9	0	2	0.0010	24 %
GasLib-135	135	105	170	92	114	43	13	0	0.0074	33 %
GasLib-582	582	160	609	122	144	294	5	163	0.0027	76 %
H-Gas	2735	474	3074	1255	1588	886	10	587	0.0213	48 %
L-Gas	4198	976	4460	1208	1436	1705	40	1269	0.0247	68 %

In contrast to the water and power network instances considered so far, the gas networks that we consider here contain active elements like valves (switches) to block gas flow, control valves to reduce gas pressure, and compressor stations that are used to increase gas pressure. Thus, the general reduction techniques described at the beginning of this section have to be adapted slightly: All techniques are only applied if the involved arcs are neither control valves nor compressor stations. These arc types stay completely untouched. This also implies that our reduction techniques are independent of the specific modeling of these two types of active elements. Valves also stay untouched for the reduction of serial and parallel arcs, but can be considered in the leaf reduction method. Here, a leaf v with incident valve (u, v) is processed as follows: If $b_v \neq 0$, we open the valve, whereas the valve status is indifferent for leaf nodes with $b_v = 0$. In our implementation, we close the valve in this situation. Potential bounds can be also updated accordingly.

Moreover, the status of valves (open or closed) can also be decided if they are bridges, i.e., if their deletion from the network's arc set decomposes the network into two disjoint components $V_1, V_2 \subseteq V$ with $V_1 \cup V_2 = V$. If now

$$\sum_{v \in V_1} b_v = \sum_{v \in V_2} b_v = 0$$

holds, the valve can be closed, whereas it has to be open otherwise. The former case decouples the two components to which the reduction techniques can then be applied independently.

Before we discuss the results, we finally remark that gas transport networks typically contain different passive arc types besides pipes, for instance, short cuts or resistors, see, e.g., [15]. For the ease of implementation and presentation we subsume all passive gas arc types as pipes.

The results are given in Table 3. The runtimes are always less than 1 s and the arc set is reduced by 24 % to 76 %. Thus, the reduction is strongest on the considered gas network instances. The main reason for this is that, as real-world or quite realistic networks, these networks also contain historically grown serial branches. Due to this aspect, the serial reduction is applied, which is not the case for the other two network types. Figure 3 displays the GasLib-582 network in its original and reduced version. It is clearly visible that the main structure of the network is preserved, while all detailed structures at the boundaries of the network are removed. The latter structures correspond to regional distribution networks that often have tree-like structure, which can be reduced almost completely by our methods.

In summary, we see that our reduction techniques can be very powerful. Due to their short runtimes they lend themselves as presolve techniques, especially when it comes to solving complex optimization problems like, e.g., MINLPs (see, e.g., Geißler et al. [16, 17] and Rose et al. [33]), or other hard problems like nonconvex NLPs or MPECs (see, e.g., Schmidt [34] and Schmidt et al. [35, 36]) on these graphs. In these models, the number of variables and constraints is typically linear in the number of nodes and arcs of the underlying graph. Thus, reductions up to 76 % will have a significant positive effect on the runtimes required for solving the corresponding MINLPs.

7. Conclusions

In this paper, we have analyzed the algorithmic properties of potential-based flows. It turns out that the studied flow problems on passive networks with a single source and sink can be solved efficiently, because the network can be represented equivalently by using a single arc. However, the problems are NP-hard for active networks. Nevertheless, we employ SPG-substructures to reduce the size of practical water, power, and gas transport network instances significantly.

There are several open questions: In extension of the approximating construction in Section 5, can the sum of the transported flow be efficiently optimized for passive networks with multiple sources and sinks? Does there exist a pseudo-polynomial algorithm for the s-t-MaxFlow problem in SPGs? Similarly, it is open whether any positive results can be shown for active general networks. Moreover, it remains an open research topic to consider a framework that models transient behavior, as motivated by the physical applications.

Acknowledgments

We acknowledge funding through the DFG SFB/Transregio 154, Subprojects A01, A07, B07, and B08. This research has been performed as part of the Energie Campus Nürnberg and is supported by funding of the Bavarian State Government. Moreover, we thank our former project partner Open Grid Europe GmbH for the provision of the data. The last author is supported by the Einstein Foundation Berlin.

References

[1] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in real algebraic geometry*. Second. Vol. 10. Algorithms and Computation in Mathematics. Springer-Verlag, Berlin, 2006.

[2] D. Bienstock. *Electrical Transmission System Cascades and Vulnerability: An Operations Research Viewpoint*. MOS-SIAM Series on Optimization. Philadelphia, PA, USA: SIAM, 2015. DOI: 10.1137/1.9781611974164.

[3] G. Birkhoff and J. B. Diaz. "Non-linear network problems." In: *Quart. Appl. Math.* 13 (1956), pp. 431–443. DOI: 10.1090/qam/77398.

[4] L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and real computation*. Springer-Verlag, New York, 1998. DOI: 10.1007/978-1-4612-0701-6.

[5] L. Blum, M. Shub, and S. Smale. "On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines." In: *Bull. Amer. Math. Soc. (N.S.)* 21.1 (1989), pp. 1–46. DOI: 10.1090/S0273-0979-1989-15750-9.

[6] D. Braess. "Über ein Paradoxon aus der Verkehrsplanung." In: *Unternehmensforschung* 12 (1968), pp. 258–268.

[7] C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi, and P. Toth. *Water Network Design by MINLP*. Research Report. Feb. 2008.

[8] C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi, and P. Toth. "An MINLP Solution Method for a Water Network Problem." In: *Algorithms – ESA 2006*. Ed. by Y. Azar and T. Erlebach. Vol. 4168. Lecture Notes in Computer Science. Berlin Heidelberg: Springer, 2006, pp. 696–707. DOI: 10.1007/11841036_62.

[9] C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi, and P. Toth. "On the optimal design of water distribution networks: a practical MINLP approach." In: *Optimization and Engineering* 13.2 (2012), pp. 219–246. DOI: 10.1007/s11081-011-9141-7.

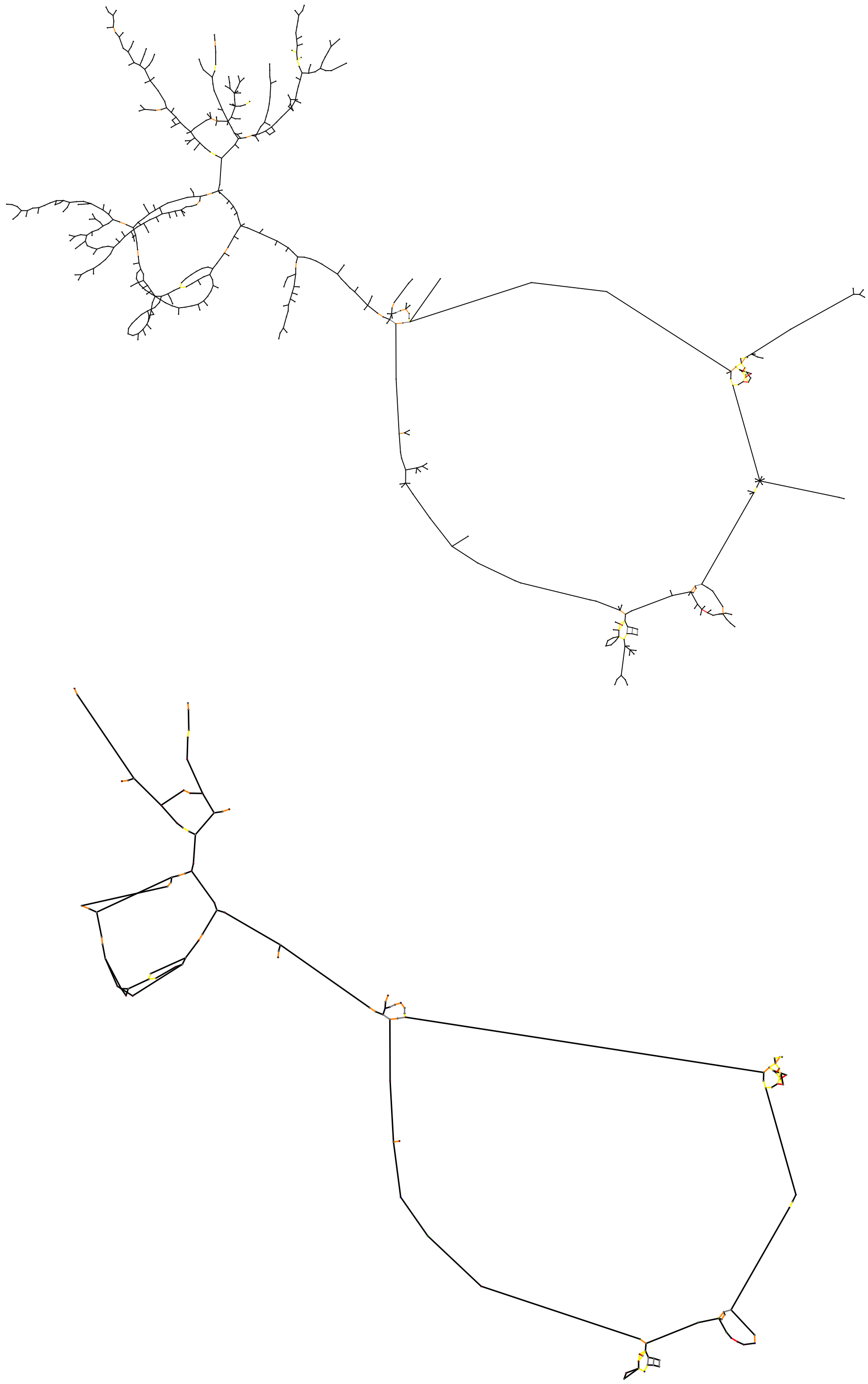

FIGURE 3. GasLib-582 before (top) and after (below) reduction

[10] J. Burgschweiger, B. Gnädig, and M. C. Steinbach. "Nonlinear Programming Techniques for Operative Planning in Large Drinking Water Networks." In: *The Open Appl. Math. J.* 3 (2009), pp. 14–28. DOI: 10.2174/1874114200903010014.

[11] B. Calvert and G. Keady. "Braess's Paradox and Power-Law Nonlinearities in Networks." In: *J. Austral. Math. Soc., Ser. B* 35 (1993), pp. 1–22.

[12] M. Collins, L. Cooper, R. Helgason, J. Kennington, and L. LeBlanc. "Solving the pipe network analysis problem using optimization techniques." In: *Management Sci.* 24.7 (1978), pp. 747–760. DOI: 10.1287/mnsc.24.7.747.

[13] R. J. Duffin. "Topology of Series-Parallel Networks." In: *Journal of Mathematical Analysis and Applications* 10 (1965), pp. 303–318.

[14] D. Eppstein. "Parallel recognition of series-parallel graphs." In: *Information and Computation* 98 (1992), pp. 41–55. DOI: 10.1016/0890-5401(92)90041-D.

[15] A. Fügenschuh, B. Geißler, R. Gollmer, A. Morsi, M. E. Pfetsch, J. Rövekamp, M. Schmidt, K. Spreckelsen, and M. C. Steinbach. "Physical and technical fundamentals of gas networks." In: *Evaluating Gas Network Capacities*. Ed. by T. Koch, B. Hiller, M. E. Pfetsch, and L. Schewe. SIAM-MOS series on Optimization. Philadelphia, PA, USA: SIAM, 2015. Chap. 2, pp. 17–44. DOI: 10.1137/1.9781611973693.ch2.

[16] B. Geißler, A. Morsi, L. Schewe, and M. Schmidt. *Solving Highly Detailed Gas Transport MINLPs: Block Separability and Penalty Alternating Direction Methods*. Tech. rep. July 2016. URL: http://www.optimization-online.org/DB_HTML/2016/06/5523.html. Submitted.

[17] B. Geißler, A. Morsi, L. Schewe, and M. Schmidt. "Solving power-constrained gas transportation problems using an MIP-based alternating direction method." In: *Computers & Chemical Engineering* 82 (2015), pp. 303–317. DOI: 10.1016/j.compchemeng.2015.07.005.

[18] F. M. Hante, G. Leugering, A. Martin, L. Schewe, and M. Schmidt. "Challenges in optimal control problems for gas and fluid flow in networks of pipes and canals: From modeling to industrial applications." In: *Industrial Mathematics and Complex Systems. Emerging Mathematical Models, Methods and Algorithms*. Ed. by P. Manchanda, R. Lozi, and A. H. Siddiqi. Industrial and Applied Mathematics. Springer, Jan. 2017. DOI: 10.1007/978-981-10-3758-0_5. In press.

[19] E. Horowitz and S. Sahni. "Computing Partitions with Applications to the Knapsack Problem." In: *Journal of the ACM* 21.2 (Apr. 1974), pp. 277–292. DOI: 10.1145/321812.321823.

[20] J. Humpola. "Gas network optimization by MINLP." PhD thesis. Technische Universität Berlin, 2014. DOI: 10.14279/depositonce-4255.

[21] J. Humpola, I. Joormann, D. Oucherif, M. E. Pfetsch, L. Schewe, M. Schmidt, and R. Schwarz. *GasLib – A Library of Gas Network Instances*. Tech. rep. Nov. 2015. URL: http://www.optimization-online.org/DB_HTML/2015/11/5216.html.

[22] D. Kirschen and G. Strbac. "Transmission networks and electricity markets." In: *Fundamentals of Power System Economics* (2004), pp. 141–204.

[23] T. Koch, B. Hiller, M. E. Pfetsch, and L. Schewe, eds. *Evaluating Gas Network Capacities*. SIAM-MOS series on Optimization. Philadelphia, PA, USA: SIAM, 2015. DOI: 10.1137/1.9781611973693.

[24] K. Koiliaris and C. Xu. "A Faster Pseudopolynomial Time Algorithm for Subset Sum." In: *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*. 2017, pp. 1062–1072. DOI: 10.1137/1.9781611974782.68.

[25] N. M. Korneyenko. "Combinatorial algorithms on a class of graphs." In: *Discrete Applied Mathematics* 54 (1994), pp. 215–217. DOI: 10.1016/0166-218X(94)90022-1.

[26] B. E. Larock, R. W. Jeppson, and G. Z. Watters. *Hydraulics of pipeline systems*. CRC press, 2010.

[27] K. Lehmann, A. Grastien, and P. V. Hentenryck. "AC-Feasibility on Tree Networks is NP-Hard." In: *IEEE Transactions on Power Systems* 31.1 (2016), pp. 798–801. DOI: 10.1109/TPWRS.2015.2407363.

[28] K. Lehmann, A. Grastien, and P. Van Hentenryck. *The Complexity of DC-Switching Problems*. arXiv 1411.4369. 2014. URL: https://arxiv.org/abs/1411.4369.

[29] J. J. Maugis. "Etude de réseaux de transport et de distribution de fluide." In: *RAIRO Oper. Res.* 11.2 (1977), pp. 243–248.

[30] D. Pisinger. "Linear Time Algorithms for Knapsack Problems with Bounded Weights." In: *Journal of Algorithms* 33.1 (1999), pp. 1–14. DOI: 10.1006/jagm.1999.1034.

[31] R. Z. Ríos-Mercado, S. Wu, L. R. Scott, and E. A. Boyd. "A Reduction Technique for Natural Gas Transmission Network Optimization Problems." In: *Ann. Oper. Res.* 117.1 (2002), pp. 217–234. DOI: 10.1023/A:1021529709006.

[32] R. T. Rockafellar. *Network Flows and Monotropic Optimization*. 2nd. Belmont, Mass., USA: Athena Scientific, 1998.

[33] D. Rose, M. Schmidt, M. C. Steinbach, and B. M. Willert. "Computational optimization of gas compressor stations: MINLP models versus continuous reformulations." In: *Mathematical Methods of Operations Research* 83.3 (June 2016), pp. 409–444. DOI: 10.1007/s00186-016-0533-5.

[34] M. Schmidt. "A Generic Interior-Point Framework for Nonsmooth and Complementarity Constrained Nonlinear Optimization." PhD thesis. Leibniz Universität Hannover, 2013.

[35] M. Schmidt, M. C. Steinbach, and B. M. Willert. "A Primal Heuristic for Nonsmooth Mixed Integer Nonlinear Optimization." In: *Facets of Combinatorial Optimization*. Ed. by M. Jünger and G. Reinelt. Berlin Heidelberg: Springer, 2013, pp. 295–320. DOI: 10.1007/978-3-642-38189-8_13.

[36] M. Schmidt, M. C. Steinbach, and B. M. Willert. "High detail stationary optimization models for gas networks: validation and results." In: *Optimization and Engineering* 17.2 (2016), pp. 437–472. DOI: 10.1007/s11081-015-9300-3.

[37] M. I. Shamos. "Computational Geometry." PhD thesis. Yale University, 1978.

[38] J. Szabó. *The set of solutions to nomination validation in passive gas transportation networks with a generalized flow formula*. ZIB-Report 11-44. Zuse Intitute Berlin, 2012.

[39] K. Takamizawa, T. Nishizeki, and N. Saito. "Linear-time Computability of Combinatorial Problems on Series-parallel Graphs." In: *Journal of the ACM* 29 (1982), pp. 623–641. DOI: 10.1145/322326.322328.

[40] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas. "MATPOWER: Steady-State Operations, Planning and Analysis Tools for Power Systems Research and Education." In: *IEEE Transactions on Power Systems* 26.1 (2011), pp. 12–19. DOI: 10.1109/TPWRS.2010.2051168.

Martin Gross, University of Waterloo, 200 University Ave W, Waterloo, ON N2L 3G1, Canada

E-mail address: mgrob@uwaterloo.ca

Marc E. Pfetsch, Research Group Optimization, TU Darmstadt, Dolivostr. 15, 64293 Darmstadt, Germany

E-mail address: pfetsch@opt.tu-darmstadt.de

Lars Schewe, (a) Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Discrete Optimization, Cauerstr. 11, 91058 Erlangen, Germany; (b) Energie Campus Nürnberg, Fürther Str. 250, 90429 Nürnberg, Germany

E-mail address: lars.schewe@fau.de

Martin Schmidt, (a) Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Discrete Optimization, Cauerstr. 11, 91058 Erlangen, Germany; (b) Energie Campus Nürnberg, Fürther Str. 250, 90429 Nürnberg, Germany

E-mail address: mar.schmidt@fau.de

Martin Skutella, Technische Universität Berlin, Institut für Mathematik MA 5-2, Strasse des 17. Juni 136, 10623 Berlin, Germany

E-mail address: martin.skutella@tu-berlin.de