



---

Konrad-Zuse-Zentrum  
für Informationstechnik Berlin

Takustraße 7  
D-14195 Berlin-Dahlem  
Germany

ARMIN FÜGENSCHUH, JESCO HUMPOLA

# **A New Class of Valid Inequalities for Nonlinear Network Design Problems**

# A New Class of Valid Inequalities for Nonlinear Network Design Problems

Armin Fügenschuh and Jesco Humpola

Zuse Institute Berlin,  
Takustr. 7, 14195 Berlin, Germany  
{fuegenschuh,humpola}@zib.de  
<http://www.zib.de>

**Abstract.** We consider a nonlinear nonconvex network design problem that arises in the extension of natural gas transmission networks. Given is such network with active and passive components, that is, valves, compressors, pressure regulators (active) and pipelines (passive), and a desired amount of flow at certain specified entry and exit nodes of the network. Besides flow conservation constraints in the nodes the flow must fulfill nonlinear nonconvex pressure loss constraints on the arcs subject to potential values (i.e., pressure levels) in both end nodes of each arc. Assume that there does not exist a feasible flow that fulfills all physical constraints and meets the desired entry and exit amounts. Then a natural question is where to extend the network by adding pipes in the most economic way such that this flow becomes feasible. Answering this question is computationally demanding because of the difficult problem structure. We use mixed-integer nonlinear programming techniques that rely on an outer approximation of the overall problem, and a branching on decision variables. We formulate a new class of valid inequalities (or cutting planes) which reduce the overall solution time when added to the formulation. We demonstrate the computational merits of our approach on test instances.

**Keywords:** Network Design; Mixed-Integer Nonlinear Programming; Cutting Planes.

## 1 Introduction

Natural gas is a nontoxic, odorless, transparent, and flammable gas that originates from underground deposits. It is often found together with crude oil or coal since it is formed by similar biological and geological processes. The main ingredients of this fossil source of energy are methane (75-99%), ethane (1-15%), propane (1-10%), and small fractions of butane and ethene. Today natural gas is mainly used for heating private houses and office buildings, for the generation of electrical power, and as fuel for vehicles. It is also used for several reactions in chemical process engineering. The world's joint resources in natural gas are assumed to last for the next 60 to 500 years, if conveyance and consumption remain on the current level. In many regions natural gas was found and excavated. The largest producers are the USA, Russia, and Canada. With natural gas around one quarter of the world's energy demand is covered. Since natural gas is the "greenest" energy source among the fossil ones, its market share is estimated to grow to 50% towards the end of this century.

The natural gas must be transported from the deposits to the customers, sometimes over distances of several thousand kilometers. For very long distances (more than 4000km) it is more economic to cool down the gas to  $-160^{\circ}\text{C}$  such that it becomes liquid and to transport it via ships. For shorter distances or for the delivery to the end customers large pipeline systems are used. The long distance steel pipelines have a diameter of up to 1.5m and the gas has a pressure of up to 100 bar. Due to friction of the gas molecules at the pipeline walls the pressure decreases. Thus every 100 to 150km a compressor station is needed to maintain the pressure level. The stations' turbines consume parts of the gas they should transport. For a 4000km distance, around 8% of the gas is lost as transport energy. For the delivery of the gas to cities a fine grid of low pressure (4-16 bar) pipelines is used. A third inner-city grid delivers gas to end customers, where the pressure is only slightly higher (20 mbar) than the atmospheric pressure.

The existing German gas network has grown over time. It was built by gas supply companies such that it can ensure the transportation of exactly the required amounts of gas to their customers. In the past, the German gas supply companies were gas vendors and gas network operators at the same time: They purchased gas from other suppliers and set up and operated the necessary infrastructure to

transport the gas from those suppliers to their customers. In course of the liberalization of the German gas market, these roles and business units were separated by regulatory authorities. Now there are companies whose sole task is the transportation of gas and who operate gas transportation networks for this purpose. Several previously independent networks were aggregated into bigger units. A discrimination free access to these networks has to be granted to everyone. This increase in flexibility for gas vendors and customers requires a higher degree of operational flexibility from the gas network operators. Although the total amount of transported gas is approximately the same, today's gas networks cannot cope with this. Various congestions show up obstructing the desired flexibility. To overcome these shortcomings a massive investment in the network infrastructure is necessary in the near future. Extension management becomes a crucial issue, since each single investment into a new compressor or a new pipe costs up to several hundreds of million Euros.

A gas network may be extended in several ways to increase the local transportation capacity. It is possible to build new pipes and to extend the capabilities of compressor stations and control valves or to build new ones. A special case of building a new pipe is looping: A loop is a pipe that follows an existing one. Loops are to some extent cheaper to build than a new pipe somewhere in the country, because land owner rights are already settled and building permissions are easier to obtain. Hence loop extensions are first-choice, long before a non-loop pipeline is considered. Thus in the computational section we focus on loop extensions, the methods, however, can be applied to any kind of extension by new pipelines.

Several approaches to improve the topology of a gas network are reported in the literature. Mainly various heuristic and local optimization methods are in use. Boyd et al. [5] apply a genetic algorithm to solve a pipe-sizing problem for a network with 25 nodes and 25 pipes, each of which could have one of six possible diameters. Castillo and Gonzalez [6] also apply a genetic algorithm for finding a tree topology solution for a network problem with up to 21 nodes and 20 arcs. In addition to pipes, also compressors can be placed into the network. Mariani et al. [17] describe the design problem of a natural gas pipeline. They present a set of parameters to evaluate the quality of the transportation system. Based on these they evaluate a number of potential configurations to identify the best among them. Osiadacz and Gorecki [20] formulate a network design problem for a given topology as a nonlinear optimization problem, for which they iteratively compute a local optimum. For a given topology the diameter of the pipes is a free design variable. Their method is applied to a network with up to 108 pipes and 83 nodes. De Wolf and Smeers [10] also use a nonlinear formulation and apply a local solver. They distinguish the operational problem (running the network) from the strategical investment problem (extending the network). For a given topology with up to 30 arcs and nodes they can determine (locally) optimized pipe diameters. Bonnans and André [2, 4] consider the optimal design problem of a straight pipeline system, and derive some theoretical properties of an optimal design. So far, we are not aware of approaches from the literature that apply global methods to solve network design problems for gas transmission networks.

Our contribution in this field is to apply exact optimization methods that can converge to and prove global optimality. Our theoretical contribution is the introduction of a new class of valid inequalities that improve the relaxation of the model, and thus have a positive impact on the running time of a branch-and-bound solution algorithm. Our methods were developed in cooperation with Open Grid Europe GmbH (OGE), a large gas transportation company.

The remainder of this article is organized as follows. In Section 2 we introduce the physical background of gas flows and in Section 3 the mathematical background of mixed-integer nonlinear programming. Section 4 presents a mathematical programming model for the simultaneous gas nomination and extension of a given network by selecting from a set of additional loop pipes. For this model we derive valid inequalities and discuss their properties in Section 5. These inequalities are applied to local problems, to be defined in Section 6. We demonstrate by an example that our cuts are necessary, but not sufficient feasibility conditions in Section 7. Our cuts are derived from dual Lagrange multipliers from a leaf problem, to be defined in Section 8. Under certain circumstances our cuts are not only locally but also globally valid, details of which can be found in Section 9. We show computational results obtained using this procedure in Section 10. Finally we conclude in Section 11 and give some ideas for future research directions.

## 2 Physical and Technical Background of Transmission Networks

We give mathematical descriptions for active and passive elements that are the basic building blocks of the transmission networks we study.

## 2.1 Pipes

The majority of the edges in a transmission network are passive pipes. In a network with node potentials the amount of flow over an edge is determined by the actual node potential values at both ends. Depending on the physical properties of the flow different functional relationships are described in the literature to approximatively determine the flow value. The fundamental equation we assume for an edge  $e = (v, w)$  is

$$\alpha_e q_e |q_e|^{k_e} = \pi_v - \pi_w. \quad (2.1)$$

Here  $\alpha_e$  and  $k_e$  are constants that subsume all physical properties of the edge, the flow, and the interactions of the flow with the edge. Although each edge  $e$  in principle might have a different value for  $k_e$  it is natural to assume that all edges have the same constant. The mathematical theory which we present in the following relies on the fact that  $k_e \in \{0, \text{const}\}$ , with  $\text{const} > 0$ . The variable  $q_e \in \mathbb{R}$  represents the flow, where a positive value is a flow from  $v$  to  $w$ , and a negative value is a flow in the opposite direction from  $w$  to  $v$ . The variables  $\pi_v, \pi_w$  are the node potential values.

**Gas Networks.** The Weymouth equation [27] is an old but still used equation to approximate the flow of gas in long pipelines. It relates the pressure of the gas in the end nodes,  $p_v$  and  $p_w$ , to the flow in the following way:

$$q_e |q_e| = C_e^2 \cdot (p_v^2 - p_w^2), \quad (2.2)$$

where  $C_e^2$  is computed by the following formula:

$$C_e^2 := 96.074830 \cdot 10^{-15} \frac{d_e^5}{\lambda_e z T L_e \delta},$$

where

$$\frac{1}{\lambda_e} = \left( 2 \log \left( \frac{3.7 d_e}{\varepsilon} \right) \right)^2,$$

with  $L_e$  being the length of the pipe (km),  $d_e$  the inner diameter of the pipe (mm),  $T$  the gas temperature (K),  $\varepsilon$  the absolute roughness of the pipe (mm),  $\delta$  the density of the gas relative to air, and  $z$  the gas compressibility factor [19, 11, 22]. Note that our approach cannot cope with other more accurate pressure loss equations, if  $C_e^2$  is not a constant but a flow- or pressure-dependent variable. After substituting  $\pi_v = p_v^2$  and  $\pi_w = p_w^2$  Weymouth's equation takes the form of (2.1). For more details we refer to [21].

**Water Networks.** Our mathematical approach is also suited for water networks. The flow of water in pipelines can, for instance, be approximated by the Hazen-Williams equation [15]:

$$\frac{10.67}{C_e^{1.85}} \frac{L_e}{d_e^{4.87}} q_e |q_e|^{0.85} = \rho \cdot g \cdot (\pi_v - \pi_w), \quad (2.3)$$

where  $C_e$  is a dimensionless roughness coefficient (typically ranging between 90 and 150),  $L_e$  is the length (meters),  $d_e$  is the inside diameter (meters),  $\rho$  is the density of the fluid ( $\text{kg}/\text{m}^3$ ),  $g$  is the local acceleration due to gravity ( $\text{m}/\text{s}^2$ ),  $q_e$  is the volumetric flow rate (cubic meters per second), and  $\pi_v, \pi_w$  are the pressures in nodes  $v$  and  $w$  (Pa).

## 2.2 Valves

A valve is installed in the network to separate or join two independent pipes. They allow for a discrete decision, either being open or close. The spatial dimension of a valve is assumed to be small in comparison to the pipes. Hence in our model the node potential values are identified when the valve is open. If the valve is closed then they are decoupled. Mathematically a valve is an edge  $e = (v, w)$  with the following description:

$$x_e = 1 \Rightarrow \pi_v - \pi_w = 0, \quad (2.4a)$$

$$x_e = 0 \Rightarrow q_e = 0, \quad (2.4b)$$

where  $x_e \in \{0, 1\}$  is a binary decision variable.

### 2.3 Increasing the Node Potential

In transmission networks it is necessary at certain locations to increase the node potential value. For example, in gas networks the pressure is too low after a transport distance of 100-150km. Here gas turbines are used as compressors. For the mathematical description of such active network elements, various models exist in the literature. We follow the approach of De Wolf and Smeers [11], and make use of the following formulation for a pipe  $e = (v, w)$  with a compressor:

$$\alpha_e q_e |q_e|^{k_e} \geq \pi_v - \pi_w, \quad (2.5)$$

which allows a flow larger than the one corresponding to the pressure decrease in the pipe. We rewrite this inequality as equality by introducing a weighted slack variable  $y_e$ :

$$\alpha_e q_e |q_e|^{k_e} + \beta_e y_e = \pi_v - \pi_w. \quad (2.6)$$

Note that the flow can only go in positive direction through a compressor, hence the lower bound needs to be set accordingly, i.e.,  $q_e \geq 0$ .

### 2.4 Reducing the Node Potential

It can be necessary to reduce the node potential along an edge  $e = (v, w)$  in the network, for example, to protect parts of the network from high potentials. In gas networks, for instance, these are pressure regulation stations that reduce the gas pressure. A pipe with a pressure regulator  $e = (v, w)$  is inverse to a pipe with a compressor. Hence we need to turn the sense of the inequality (2.5) around:

$$\alpha_e q_e |q_e|^{k_e} \leq \pi_v - \pi_w, \quad (2.7)$$

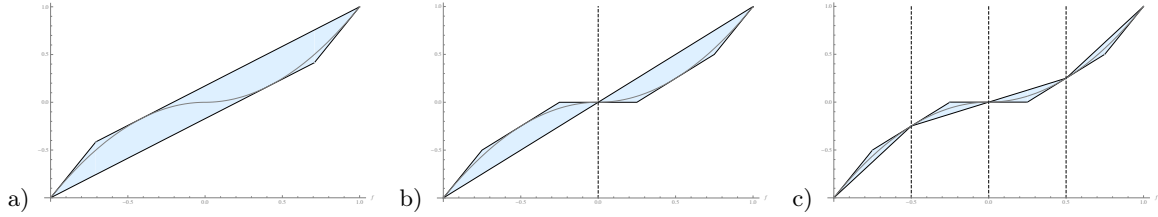
in order to decrease the pressure in  $w$  more than the flow and the input pressure would actually require. After introducing weighting slack variables  $y_e$  equation (2.7) appears similar to equation (2.6). (The only difference between a compressor and a regulator is either the sign of  $\beta$  or the bounds on  $y_e$ .) Note that the flow direction through a pressure regulator is also fixed by setting the lower bound to zero, i.e.,  $q_e \geq 0$ .

## 3 Mathematical Background

In order to obtain proven global optimal solutions we apply linear and nonlinear mixed-integer programming techniques, which we briefly introduce here.

### 3.1 Global Mixed-Integer Nonlinear Programming

We formulate the topology extension problem as mixed-integer nonlinear program (MINLP). Solving optimization problems from this class is theoretically intractable and also known to be computationally difficult in general. By “solving” we mean to compute a feasible solution for a given instance of the problem together with a computational proof of its optimality. Therefore we apply the general framework of a branch-and-bound approach, where the bounds are obtained from relaxations of the original model. To this end, we relax the MINLP first to a mixed-integer linear program (MILP) and then further to a linear program (LP), which is solved efficiently using the simplex algorithm. The so obtained solution value defines a (lower) bound on the optimal value of the original MINLP problem. In case this solution is MINLP feasible, it would be a proven global optimal MINLP solution. However, this rarely happens in practice. Hence we either add cutting planes to strengthen the relaxation, or we decide to branch on a variable. As an example, consider the nonlinear pressure loss constraint (2.1), c.f. Fügenschuh et al. [12]. In the LP relaxation this function is replaced by a polyhedral (linear) outer approximation, which is iteratively refined during the branch-and-bound process by branching on variables (spatial branching), see Figure 3.1. For more details on cutting planes and branch-and-bound for MILP we refer to Nemhauser and Wolsey [18], and for an application of this framework to global mixed-integer nonlinear programming to Smith and Pantelides [23], and Tawarmalani and Sahinidis [24, 25]. Information on the framework MINLP framework SCIP which we apply is given by Achterberg [1], and in particular on nonlinear aspects of SCIP in Berthold, Heinz, and Vigerske [3].



**Fig. 3.1.** a) Polyhedral outer approximation of  $q_e \mapsto \alpha_e q_e |q_e|$ , b) initial spatial branching on zero, c) further spatial branching.

### 3.2 Nonlinear Programming

In addition to the simplex algorithm for linear programs we use nonlinear solvers on nodes of the branch-and-bound tree. As soon as all binary decisions and the flow directions are fixed, the remaining problem at this node is (equivalent to) a convex nonlinear problem (see Collins et al. [7]). To compute optimal solutions for these subproblems we apply the solver IPOPT from Wächter and Biegler [26]. It applies a primal-dual interior point (or barrier) method with a filter line-search method. One of the central underlying method in nonlinear programming, which is part of IPOPT and which we also apply directly in our solution approach, are the Karush-Kuhn-Tucker (KKT) conditions. Under certain additional assumptions they provide necessary conditions for a (local) optimum. For a nonlinear optimization problem of the form  $\min\{f(x) : g_i(x) \leq 0, h_j(x) = 0, x \in \mathbb{R}^n\}$ , where  $f$  is the objective function,  $g_i (i = 1, \dots, m)$  are continuously differentiable inequality constraint functions and  $h_j (j = 1, \dots, \ell)$  are continuously differentiable equality constraint functions, the KKT system reads as follows

$$\nabla f(x^*) + \sum_{i=1}^m \lambda_i \nabla g_i(x^*) + \sum_{j=1}^{\ell} \mu_j \nabla h_j(x^*) = 0, \quad (3.1)$$

$$g_i(x^*) \leq 0, \forall i = 1, \dots, m, \quad (3.2)$$

$$h_j(x^*) = 0, \forall j = 1, \dots, \ell, \quad (3.3)$$

$$\lambda_i \geq 0, \forall i = 1, \dots, m, \quad (3.4)$$

$$\lambda_i g_i(x^*) = 0, \forall i = 1, \dots, m, \quad (3.5)$$

where  $x^*$  is a local minimum, and  $\lambda_i (i = 1, \dots, m), \mu_j (j = 1, \dots, \ell)$  are constants (called KKT multipliers). The existence of these constants is guaranteed if  $x^*$  satisfies some regularity conditions (to be discussed later). In the special case of  $m = 0$ , i.e., no inequality constraints exist, the KKT multipliers are also called Lagrange multipliers. For more details we refer to Conn, Gould, and Toint [8].

## 4 Topology Optimization of Transmission Networks

In the following we describe a mixed-integer nonlinear model for the extension problem in a transmission network. Our model integrates two features: if the set of potential extensions is empty, it can be used to determine if a configuration of all active elements is possible such that all physical, technical, and contractual constraints are fulfilled. For a non-empty set of potential extensions it can be used to find a subset of extensions having minimal cost and allowing a feasible flow. To this end, the model must be solved numerically with a solving technique that has the potential to give a certificate for optimality, or to proof that no solution exists. The details of our solution technique will be subject of the following sections.

### 4.1 The Model

We use the following notation for sets. A transmission network is modeled by a directed graph  $G = (V, E)$  where  $V$  denotes the set of nodes and  $E \subset V \times V$  the set of arcs. We define a set of potential extension pipes  $E_X \subset V \times V \times \mathbb{N}_0$ , where a new pipe can be built between any pair of existing nodes  $v, w \in V, v \neq w$ . By  $G_X := (V, E_X)$  we denote the transmission network together with its potential extensions. Note that  $G_X$

is a graph with multiple arcs. In the following we write  $(e, i) \in G_X$  as abbreviation for  $(e, i) \in E_X$ . For the ease of notation when formulating the model below we embed the originally given arcs from  $E$  in  $E_X$  with index 1, that is, all arcs  $(v, w, 1) \in E_X$  are not extension arcs, but original arcs from  $E$ . Furthermore, all arcs  $(v, w) \in E$  that represent valves are additionally represented by the arc  $(v, w, 0) \in E_X$ . For these valves we set  $\beta_e = 0$ . All potential extensions are indexed with  $i > 1$ . For arcs  $e$  with  $k_e = 0$  extensions are not allowed. For arcs with  $\alpha_e = 0$  we assume w.l.o.g.  $k_e = 0$ . As abbreviation we set  $k := \max\{0, \max\{k_e : e \in E\}\}$ . Besides the decision if a new pipe between two nodes should be built, we further need to decide further design parameters. For this we can select from a finite set of possible configurations, such as pipe diameters or coatings, which is reflected by index  $i$  for  $i > 1$ . For practical purpose we restrict to loop extensions  $E_X \subseteq E \times \mathbb{N}_0$ , however, the model and methods we describe below would also allow for larger sets  $E_X$ , that is, it allows for extensions between nodes  $v$  and  $w$  that were not connected in  $E$ .

We assume the following data to be given as parameters. For each node  $v \in V$  we have lower and upper bounds on the node potential,  $\underline{\pi}_v, \bar{\pi}_v \in \mathbb{R}$  with  $\underline{\pi}_v \leq \bar{\pi}_v$ . For each node  $v \in V$  the value  $s_v \in \mathbb{R}$  denotes the amount of flow that is either led into the network (for  $s_v > 0$ ), or taken out of the network (for  $s_v < 0$ ). A node with  $s_v > 0$  is also called source or entry node, and nodes with  $s_v < 0$  are sinks or exit nodes. All other nodes with  $s_v = 0$  are inner or transmission nodes. Vector  $v$  is also called *nomination*. In order not to pose a problem that is trivially infeasible, only those nominations are allowed that have equal entry and exit flows, that is,

$$\sum_{v \in V} s_v = 0. \quad (4.1)$$

Such nominations are said to be *balanced*. For each arc  $(v, w, i) \in E_X$  we have a transmission coefficient  $\alpha_{e,i} \in \mathbb{R}_+ \setminus \{0\}$ , bounds on the range coefficient  $\underline{y}_{e,i}, \bar{y}_{e,i} \in \mathbb{R}$  with  $\underline{y}_e \leq \bar{y}_e$ , a scaling factor  $\beta_{e,i}$  for the range coefficient, and a cost coefficient  $c_{e,i} \in \mathbb{R}_+$ .

Let us introduce the following variables. The flow on arc  $(e, i) \in E_X$  is denoted by  $q_{e,i} \in \mathbb{R}$ , where a positive value means the flow is heading in the same direction as the arc, and a negative value indicates the opposite direction. We remark that there are no explicit bounds on  $q_{e,i}$  that would specify a maximal absolute flow value in either direction, but implicitly given bounds by the bounds on the potential values and the pressure-flow-coupling constraints (to be defined below). The potential value of a vertex  $v \in V$  is given by  $\pi_v \in \mathbb{R}$ . For example, in a gas transmission network this variable refers to the pressure in this node. The variable  $y_{e,i} \in \mathbb{Z}$  specifies that additive component of the pressure loss term. For passive pipelines this variable is fixed to zero, whereas for active elements it defines the operating range. We introduce a binary decision variable  $x_{e,i} \in \{0, 1\}$  for each potential extension  $(e, i) \in E_X$ .

The following non-linear non-convex mixed-integer model is called topology optimization problem:

$$\min \sum_{(e,i) \in E_X} c_{e,i} x_{e,i} \quad \text{s.t.} \quad (4.2a)$$

$$x_{e,i} = 1 \Rightarrow \alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \beta_{e,i} y_{e,i} - (\pi_v - \pi_w) = 0 \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2b)$$

$$x_{e,i} = 0 \Rightarrow q_{e,i} = 0 \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2c)$$

$$x_{e,i} = 0 \Rightarrow y_{e,i} = 0 \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2d)$$

$$\sum_{i:(v,w,i) \in E_X} x_{e,i} = 1 \quad \forall e \in E, \quad (4.2e)$$

$$\sum_{\substack{w, i:(w,v,i) \in E_X \\ i \neq 0}} q_{w,v,i} - \sum_{\substack{w, i:(v,w,i) \in E_X \\ i \neq 0}} q_{v,w,i} = s_v \quad \forall v \in V, \quad (4.2f)$$

$$\pi_v \leq \bar{\pi}_v \quad \forall v \in V, \quad (4.2g)$$

$$\pi_v \geq \underline{\pi}_v \quad \forall v \in V, \quad (4.2h)$$

$$x_{e,i} = 1 \Rightarrow y_{e,i} \leq \bar{y}_{e,i} \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2i)$$

$$x_{e,i} = 1 \Rightarrow y_{e,i} \geq \underline{y}_{e,i} \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2j)$$

$$x_{e,i} \leq 1 \quad \forall (e, i) \in E_X, \quad (4.2k)$$

$$x_{e,i} \geq 0 \quad \forall (e, i) \in E_X, \quad (4.2l)$$

$$q_{e,i} \in \mathbb{R} \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2m)$$

$$y_{e,i} \in \mathbb{Z} \quad \forall (e, i) \in E_X, i \neq 0, \quad (4.2n)$$

$$x_{e,i} \in \mathbb{Z} \quad \forall (e, i) \in E_X. \quad (4.2o)$$

The objective function (4.2a) calculates the extension costs for those new pipes that are actually built. The indicator constraints (4.2b) are switching on only those pressure-flow coupling constraints for potential arcs that are actually built. The indicator constraints (4.2c) forbid flow on those arcs that are not used, that is, they are either not built or switched off by a closed valve. Exactly one pressure loss constraint (4.2b) must be selected which is guaranteed by constraints (4.2e). The node flow conservation constraints (also called Kirchhoff's constraints) are defined in (4.2f). Constraints (4.2g) – (4.2l) define the trivial bounds on the variables, and constraints (4.2m) – (4.2o) specify the continuous or discrete range of the variables.

For a given nomination  $s$ , the topology optimization problem (4.2) is to find a cost optimal selection of pipe capacities for the transmission of the specific flow  $s$  in the transmission network  $G_X$ . Otherwise, if this transport is not possible for any selection of pipe capacities, the nomination is infeasible.

## 4.2 Outline of Our Method

We give an outline of our method here. The details are given in the following sections. In order to solve the topology optimization model above we apply a branch-and-bound approach, where the subproblems at the nodes of the branching tree are solved by linear or nonlinear programming methods.

Initially, at the root node of the tree, we consider the linear relaxation of (4.2). This relaxation is obtained by relaxing all binary decision variables (for active elements and for topology extension measures) to their continuous counterparts, and by replacing the nonlinear pressure-flow coupling constraints (4.2b) by an initially coarse outer linear approximations. The relaxation is solved by Dantzig's simplex algorithm, and further cutting planes are added by standard routines of the solver (such as Gomory cuts, mixed-integer rounding cuts, and others). Further cuts are also generated by the constraint handler for the nonlinear constraints, if necessary, to strengthen the outer approximation in the convex parts of the nonlinear function.

When no further cuts are added, the root LP solution is usually neither integral nor fulfilling the nonlinearities. We then start branching, that is, we select a fractional variable with is required to be binary integral, and split the problem into two subproblem: in the one we fix it to its lower bound zero, and in the other we fix it to its upper bound one. In this way we obtained two linear subproblems out of one. It is important to remark that we only branch on variables that belong to active elements (i.e., valves) or perform spatial branchings to decide the flow direction on an arc, but never on topology extension decision variables. These subproblems are further strengthened by standard integrality cuts and by outer approximation cuts. The are all linear programs, and thus can still be solved by the simplex algorithm. This way of branching creates a tree of subproblems, where the leaves are in generally still not feasible solutions for the initial problem (4.2) (note that the topology variables might still be fractional, and the nonlinearities are not fulfilled). These open subproblems are now investigated by different methods.

Consider an open subproblem at a leave of the above branching tree. This leave node is the sub-root for a new subtree. We aim to generate a cut which is a valid inequality for the whole subtree. This cut specifies a number of topology variables to be activated in this subtree. To this end, we solve the new root subproblem by further branching on the remaining fractional variables, i.e., now the binary variables corresponding to topology extension decisions are also allowed. We also apply further spatial branchings or branching on other binary variables, if necessary. The feasible solutions at the leaves in this subtree fulfill the integrality constraints on all binary variables, and also the flow directions are settled. So far, all solutions were obtained by linear programming (simplex algorithm). However, the solutions might still be not feasible for the nonlinear constraints.

In order to obtain feasible nonlinear solutions, we apply a continuous nonlinear solver to an NLP variant of the leaf node problem. Since the problem might be linear feasible, but nonlinear infeasible, we reformulate the problem. We relax the upper and lower bound constraints on the pressure variables



by adding non-negative slack variables to the bounds. The objective function of the problem is then to minimize the 1-norm of the slack vector (i.e., the sum of all slacks). Moreover, we take back all branching decisions on the flow directions and offer again both forward and backward directions for the arc gas flow. Despite the fact that the constraints are nonconvex it can be shown that this problem is equivalent to a convex problem (see Collins et al. [7]), hence a local optimal solution is already a global one. After solving the NLP with a convex nonlinear solver one of the following three cases occurs. Note that because of the relaxation of the bound by slack variables the NLP is always feasible.

In the first case, the NLP has a zero objective function value in the optimal solution and all flows are heading in the same direction as in the linear subproblem. We obtain a primal feasible solution.

In the second case, the NLP also has a zero objective function value in the optimal solution, but reverses some flows compared to the linear subproblem. Here we also obtain a feasible solution.

In the third case, the NLP might have a positive objective function value in the optimal solution. This means that some slack variables could not become zero. In this case we set up a certain KKT system, which turns out to be a large but easy to solve linear program, using the NLP solution vector as input. The strength of this cut is related to the objective function value of the NLP. The solution of the KKT linear program determines optimal dual variables. From these we can define a local cutting plane which is a valid inequality for a whole subtree. It is stored in a cut pool and added by the solver as soon as this subtree is being generated. Moreover this proves that the node is nonlinear infeasible and can be pruned.

## 5 Inequalities for the Topology Optimization Problem

In the following section we define a *potential difference inequality* (PDI). We provide a proof showing that it gives a valid cut on the pipe capacities for the topology optimization problem (4.2). To clarify, this cut will forbid infeasible selections of pipe capacities.

**Definition 1.** *When fixing all flow directions and all integer variables in (4.2) we speak of a leaf problem of (4.2).*

In order to formulate the leaf problem as MINLP we give now some notations as abbreviations. We define  $\tilde{\beta}_{e,i} := \beta_{e,i} y_{e,i}$ , where  $y_{e,i}$  is a given integer value. We define  $E'_X := \{(v, w, i) \in V \times V \times \mathbb{N}_0 : (v, w, i) \in E_X \wedge x_{v,w,i} = 1 \wedge q_{v,w,i} \geq 0\} \cup \{(w, v, i) \in V \times V \times \mathbb{N}_0 : (v, w, i) \in E_X \wedge x_{v,w,i} = 1 \wedge q_{v,w,i} < 0\}$ .  $E'_X$  contains all arcs from  $E_X$  that are selected (i.e., where the corresponding  $x$ -variable is nonzero), and these arcs are oriented in such way that the flow is nonnegative. Using these definitions the leaf problem of (4.2) is as follows:

$$\alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i} - (\pi_v - \pi_w) = 0 \quad \forall (e, i) \in E'_X, i \neq 0, \quad (5.1a)$$

$$\sum_{\substack{w, i: (w, v, i) \in E'_X \\ i \neq 0}} q_{w, v, i} - \sum_{\substack{w, i: (v, w, i) \in E'_X \\ i \neq 0}} q_{v, w, i} = s_v \quad \forall v \in V, \quad (5.1b)$$

$$\pi_v \leq \bar{\pi}_v \quad \forall v \in V, \quad (5.1c)$$

$$\pi_v \geq \underline{\pi}_v \quad \forall v \in V, \quad (5.1d)$$

$$q_{e,i} \in \mathbb{R}_+ \quad \forall (e, i) \in E'_X, i \neq 0. \quad (5.1e)$$

The leaf problem is a pure existence problem, because all coefficients in the objective function belong to integer variables which are already assumed to be fixed to certain values.

**Definition 2.** *Every vector  $(\mu, \lambda) = (\mu_v, \mu_{e,i}, \lambda_v^+, \lambda_v^-)_{v \in V, (e,i) \in E'_X}$ , such that  $\mu_v, \mu_{e,i} \in \mathbb{R}$  where for each  $e \in E$  exactly one  $\mu_{e,i} > 0$  for some  $i \in \mathbb{N}_0$ , and  $\lambda_v^+, \lambda_v^- \in \mathbb{R}_{\geq 0}$  where not both are positive, which fulfills the following constraints*

$$0 \leq \text{sign}(\mu_{e,i}) = \text{sign}(\mu_v - \mu_w) \quad ((v, w, i) = (e, i) \in E'_X, k_e \neq 0), \quad (5.2)$$

$$\frac{\alpha_{e,i}}{k+1} \cdot \mu_{e,i} = \mu_v - \mu_w \quad ((v, w, i) = (e, i) \in E'_X, k_e = 0), \quad (5.3)$$

$$\sum_{e \in \delta^+(v)} \mu_{e,i} - \sum_{e \in \delta^-(v)} \mu_{e,i} = \lambda_v^+ - \lambda_v^- \quad (v \in V), \quad (5.4)$$

is called **dual transmission flow**.

Let us analyze a dual transmission flow  $(\mu, \lambda)$ . We observe that  $(\mu_{e,i})_{(e,i) \in E'_X}$  represents a network flow in  $G'_X$  with node flow  $\lambda_v^+ - \lambda_v^-$  for a node  $v \in V$ . For arcs  $(e, i) \in E'_X$  with  $k_e = 0$  this flow  $\mu_e$  is induced by the potential difference  $\mu_v - \mu_w$ , and for all others (with  $k_e \neq 0$ ) we require that the flow  $\mu_{e,i}$  and the potential difference  $\mu_v - \mu_w$  have the same direction. Later we will show that the dual values corresponding to an optimal solution of the transmission network flow problem fulfill these constraints except the non-negativity condition in (5.2). This will justify to regard  $(\mu, \lambda)$  as a dual transmission flow. Furthermore we observe that the dual flow  $\mu_{e,i}$  and the primal flow  $q_{e,i}$  are never going into opposite directions: From (5.2) and the orientation of  $E'_X$  it follows that  $q_{e,i} \cdot \mu_{e,i} \geq 0$ .

**Definition 3.** For a dual transmission flow  $(\mu, \lambda)$  the inequality

$$\begin{aligned} (k+1) \sum_{v \in V} s_v \mu_v - \sum_{(v,w,i) = (e,i) \in E'_X} k_e \left( \frac{(\mu_v - \mu_w)^{k+1}}{\alpha_{e,i} \mu_{e,i}} \right)^{\frac{1}{k}} + \sum_{(e,i) \in E'_X} \mu_{e,i} \tilde{\beta}_{e,i} \\ \leq \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v) \end{aligned}$$

is called a **potential difference inequality (PDI)**.

Using the definitions above we give the following theorem on valid inequalities for (5.1).

**Theorem 1.** The PDI of a dual transmission flow  $(\mu, \lambda)$  is a necessary condition for the leaf problem (5.1) being feasible.

**Lemma 1.** For  $\tilde{\delta}_e \in [0, 1]$  the inequality

$$\sum_{\substack{(e,i) \in E'_X, \\ i \neq 0}} \mu_{e,i} (\alpha_{e,i} \tilde{\delta}_{e,i} q_{e,i}^{k_e+1} + \tilde{\beta}_{e,i}) \leq \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v). \quad (5.5)$$

is a necessary condition for the feasibility of (5.1).

Proof: We denote by  $\mathcal{P}$  the set of all simple paths in  $(V, E'_X)$  and by  $\mathcal{C}$  the set of all circuits in  $G_X$ . Let  $\mu_{e,i}$  be described by path flows from source  $s(P)$  to target  $t(P)$  with an amount of  $f_P \geq 0, P \in \mathcal{P}$  and circuit flows  $f_C \geq 0, C \in \mathcal{C}$  such that

$$\mu_{e,i} = \sum_{P \in \mathcal{P}: (e,i) \in P} f_P + \sum_{C \in \mathcal{C}: (e,i) \in C} f_C. \quad (5.6)$$

For a path  $P \in \mathcal{P}$  we observe

$$\sum_{(v,w,i) \in P} \pi_v - \pi_w = \pi_{s(P)} - \pi_{t(P)} \leq \bar{\pi}_{s(P)} - \underline{\pi}_{t(P)}.$$

From this follows

$$\sum_{P \in \mathcal{P}} f_P \left( \sum_{(v,w,i) \in P} \pi_v - \pi_w \right) \leq \sum_{P \in \mathcal{P}} f_P \left( \bar{\pi}_{s(P)} - \underline{\pi}_{t(P)} \right). \quad (5.7)$$

We rewrite the left hand side by a double counting argument and obtain

$$\sum_{P \in \mathcal{P}} f_P \left( \sum_{(v,w,i) \in P} \pi_v - \pi_w \right) = \sum_{\substack{(v,w,i) \in E'_X, \\ i \neq 0}} (\pi_v - \pi_w) \left( \sum_{P \in \mathcal{P}: (v,w,i) \in P} f_P \right).$$

Note that we can expand the right hand side by adding any constant number on a circle because the values cancel to zero by a telescope sum argument:

$$\begin{aligned} & \sum_{\substack{(v,w,i) \in E'_X, \\ i \neq 0}} (\pi_v - \pi_w) \left( \sum_{P \in \mathcal{P}: (v,w,i) \in P} f_P \right) = \\ & \sum_{\substack{(v,w,i) \in E'_X, \\ i \neq 0}} (\pi_v - \pi_w) \left( \sum_{P \in \mathcal{P}: (v,w,i) \in P} f_P + \sum_{C \in \mathcal{C}: (e,i) \in C} f_C \right). \end{aligned}$$

Let  $\tilde{\delta}_{e,i} \in [0, 1]$ . We apply the equation (4.2b) and (5.6) to the right hand side and obtain

$$\begin{aligned} & \sum_{\substack{(v,w,i) \in E'_X, \\ i \neq 0}} (\pi_v - \pi_w) \left( \sum_{P \in \mathcal{P}: (v,w,i) \in P} f_P + \sum_{C \in \mathcal{C}: (e,i) \in C} f_C \right) \geq \\ & \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0}} \mu_{e,i} (\alpha_{e,i} \tilde{\delta}_{e,i} q_{e,i}^{k_e+1} + \tilde{\beta}_{e,i}). \end{aligned}$$

We proceed by reformulating the right hand side of (4.2b). We define

$$\lambda_{P,s(P)}^+ = f_P, \lambda_{P,t(P)}^- = f_P, \lambda_{P,v}^\pm = 0, v \neq s(P), t(P), \quad (5.8)$$

and thus obtain

$$\sum_{P \in \mathcal{P}} f_P (\bar{\pi}_{s(P)} - \underline{\pi}_{t(P)}) = \sum_{P \in \mathcal{P}} \sum_{v \in V} (\lambda_{P,v}^+ \bar{\pi}_v - \lambda_{P,v}^- \underline{\pi}_v).$$

By a double counting argument we get

$$\sum_{P \in \mathcal{P}} \sum_{v \in V} (\lambda_{P,v}^+ \bar{\pi}_v - \lambda_{P,v}^- \underline{\pi}_v) = \sum_{v \in V} \left( \bar{\pi}_v \left( \sum_{P \in \mathcal{P}} \lambda_{P,v}^+ \right) - \underline{\pi}_v \left( \sum_{P \in \mathcal{P}} \lambda_{P,v}^- \right) \right)$$

Applying equation (5.8), (5.6) and (5.4) and gives

$$\sum_{v \in V} \left( \bar{\pi}_v \left( \sum_{P \in \mathcal{P}} \lambda_{P,v}^+ \right) - \underline{\pi}_v \left( \sum_{P \in \mathcal{P}} \lambda_{P,v}^- \right) \right) = \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v).$$

Putting the reformulation of the right hand side of (5.7) and the left hand side estimation together we derive

$$\sum_{\substack{(e,i) \in E'_X, \\ i \neq 0}} \mu_{e,i} (\alpha_{e,i} \tilde{\delta}_{e,i} q_{e,i}^{k_e+1} + \tilde{\beta}_{e,i}) \leq \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v). \quad (5.9)$$

□

Proof: For  $k_e = 0$  we set  $\delta_{e,i} = 1$ , and for  $k_e \neq 0$  we chose  $\delta_{e,i} \geq 0$  such that

$$\alpha_{e,i} (\delta_{e,i} q_{e,i})^k \mu_{e,i} = \mu_v - \mu_w, \quad \forall (e,i) = (v,w,i) \in E'_X, k_e \neq 0 \quad (5.10)$$

holds. We now set  $\tilde{\delta}_{e,i} := \delta_{e,i}^k [k+1 - k\delta_{e,i}]$ . It follows from easy calculus that  $\tilde{\delta}_{e,i}$  achieves a global maximum for  $\delta_{e,i} = 1$  which is  $\tilde{\delta}_{e,i} = 1$ . In order to prove the theorem, we give the following reformulations:

$$\sum_{(e,i) \in E'_X, i \neq 0} \alpha_{e,i} q_{e,i}^{k_e+1} \mu_{e,i} \tilde{\delta}_{e,i} \quad (5.11)$$

$$= \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e \neq 0}} \alpha_{e,i} q_{e,i}^{k+1} \mu_{e,i} \tilde{\delta}_{e,i} + \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e = 0}} \alpha_{e,i} q_{e,i} \mu_{e,i} \quad (5.12)$$

$$= \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e \neq 0}} \alpha_{e,i} q_{e,i}^{k_e+1} \mu_{e,i} \delta_e^k [k+1 - k\delta_{e,i}] + \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e = 0}} \alpha_{e,i} q_{e,i} \mu_{e,i} \quad (5.13)$$

$$= \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e \neq 0}} q_{e,i} (\mu_v - \mu_w) [k+1 - k\delta_{e,i}] + (k+1) \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e = 0}} q_{e,i} (\mu_v - \mu_w) \quad (5.14)$$

$$= (k+1) \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0}} q_{e,i} (\mu_v - \mu_w) - k \sum_{\substack{(e,i) \in E'_X, \\ i \neq 0, k_e \neq 0}} \delta_{e,i} q_{e,i} (\mu_v - \mu_w) \quad (5.15)$$

$$= (k+1) \sum_{(e,i) \in E'_X, i \neq 0} q_{e,i} (\mu_v - \mu_w) - \sum_{(e,i) \in E'_X, i \neq 0} k_e \left( \frac{(\mu_v - \mu_w)^{k+1}}{\alpha_{e,i} \mu_{e,i}} \right)^{\frac{1}{k}} \quad (5.16)$$

Because of

$$\sum_{(e,i) \in E'_X, i \neq 0} q_{e,i} (\mu_v - \mu_w) = \sum_{v \in V} s_v \mu_v \quad (5.17)$$

we obtain from the previous Lemma

$$\begin{aligned} (k+1) \sum_{v \in V} s_v \mu_v - \sum_{(e,i) \in E'_X} k_e \left( \frac{|\mu_v - \mu_w|^{k+1}}{\alpha_{e,i} |\mu_{e,i}|} \right)^{\frac{1}{k}} - \sum_{(e,i) \in E'_X} \mu_{e,i} \tilde{\beta}_{e,i} \\ \leq \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v) \end{aligned}$$

□

## 6 PDIs for Local Problems

In this section we define a sub-MINLP that consists of the topology optimization problem (4.2) at a certain node in the branch-and-bound tree where already all decisions on the direction of the flow and all decisions on opening and closing valves are settled, that is,  $x_{e,0}$  are fixed to either 0 or 1.

**Definition 4.** *When fixing all flow directions and all integer variables corresponding to valves in (4.2) we speak of a local problem of (4.2).*

Note that the local problem is still an optimization problem because the capacity integer decision variables are not fixed yet, and thus the objective function remains as in the topology optimization problem (4.2). Furthermore the operating range variable for active elements  $y_{e,i}$  is also not fixed. So the local problem is an optimization problem that depends on the given parameters  $\text{sign}(q_{e,i})$  for all  $(e,i) \in E_X$  and  $x_{e,0}$  for all  $(e,0) \in E_X$ .

In order to formulate the local problem as MINLP we give now some notations as abbreviations. We define  $E''_X := \{(v,w,i) \in V \times V \times \mathbb{N}_0 : (v,w,i) \in E_X \wedge q_{v,w,i} \geq 0\} \cup \{(w,v,i) \in V \times V \times \mathbb{N}_0 : (v,w,i) \in E_X \wedge q_{v,w,i} < 0\}$ .  $E''_X$  contains all arcs from  $E_X$  with an orientation in such way that the flow is nonnegative. Using these definitions the local problem of (4.2) is as follows:

$$\min \sum_{(e,i) \in E''_X} c_{e,i} x_{e,i} \quad \text{s.t.} \quad (6.1a)$$

$$x_{e,i} = 1 \Rightarrow \alpha_{e,i} q_{e,i}^{k_e+1} + \beta_{e,i} y_{e,i} - (\pi_v - \pi_w) = 0 \quad \forall (e,i) \in E''_X, i \neq 0, \quad (6.1b)$$

$$x_{e,i} = 0 \Rightarrow q_{e,i} = 0 \quad \forall (e,i) \in E''_X, i \neq 0, \quad (6.1c)$$

$$x_{e,i} = 0 \Rightarrow y_{e,i} = 0 \quad \forall (e,i) \in E''_X, i \neq 0, \quad (6.1d)$$

$$\sum_{i: (v,w,i) \in E''_X} x_{e,i} = 1 \quad \forall e \in E, \quad (6.1e)$$

$$\sum_{\substack{w, i: (w,v,i) \in E''_X \\ i \neq 0}} q_{w,v,i} - \sum_{\substack{w, i: (v,w,i) \in E''_X \\ i \neq 0}} q_{v,w,i} = s_v \quad \forall v \in V, \quad (6.1f)$$

$$\pi_v \leq \bar{\pi}_v \quad \forall v \in V, \quad (6.1g)$$

$$\pi_v \geq \underline{\pi}_v \quad \forall v \in V, \quad (6.1h)$$

$$x_{e,i} = 1 \Rightarrow y_{e,i} \leq \bar{y}_{e,i} \quad \forall (e,i) \in E''_X, i \neq 0 \quad (6.1i)$$

$$x_{e,i} = 1 \Rightarrow y_{e,i} \geq \underline{y}_{e,i} \quad \forall (e,i) \in E''_X, i \neq 0 \quad (6.1j)$$

$$x_{e,i} \leq 1 \quad \forall (e,i) \in E''_X, i \neq 0 \quad (6.1k)$$

$$x_{e,i} \geq 0 \quad \forall (e,i) \in E''_X, i \neq 0 \quad (6.1l)$$

$$q_{e,i} \in \mathbb{R} \quad \forall (e,i) \in E''_X, i \neq 0, \quad (6.1m)$$

$$y_{e,i} \in \mathbb{Z} \quad \forall (e,i) \in E''_X, i \neq 0, \quad (6.1n)$$

$$x_{e,i} \in \mathbb{Z} \quad \forall (e,i) \in E''_X, i \neq 0. \quad (6.1o)$$

Based on the PDI inequality for the leaf problem we formulate a valid inequality for the local problem (6.1).

**Lemma 2.** *Let  $(\mu, \lambda)$  be a dual transmission flow. The inequality*

$$\begin{aligned} & (k+1) \sum_{v \in V} s_v \mu_v - \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v) \\ & \leq \sum_{\substack{(e,i) \in E''_X, \\ i \neq 0}} k_e \left( \frac{(\mu_v - \mu_w)^{k+1}}{\alpha_{e,i} \mu_e} \right)^{\frac{1}{k}} x_{e,i} + \sum_{\substack{(e,i) \in E''_X, \\ i \neq 0}} \mu_e \beta_{e,i} y_{e,i}, \end{aligned}$$

where  $\mu_e := \sum_{i \in \mathbb{N}_0} \mu_{e,i}$  for all  $e \in E$  is valid for the local problem (6.1).

Proof: We consider a local problem, and any of its leaf problems. By Theorem 1 the PDI

$$\begin{aligned} & (k+1) \sum_{v \in V} s_v \mu_v - \sum_{(v,w,i)=(e,i) \in E'_X} k_e \left( \frac{|\mu_v - \mu_w|^{k+1}}{\alpha_{e,i} |\mu_{e,i}|} \right)^{\frac{1}{k}} + \sum_{(e,i) \in E'_X} \mu_{e,i} \tilde{\beta}_{e,i} \\ & \leq \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v) \end{aligned}$$

is valid for this leaf. Note that the PDI is valid for any dual transmission flow  $(\mu, \lambda)$ . In particular, it does not depend on the actual selection of the leaf. Hence the same  $(\mu, \lambda)$  can be used to formulate PDIs for all leaves corresponding to the same local problem. Consider any other leaf with a different choice of extension pipes. This means that for some edge  $e$  a different value  $i'$  would be selected. In order to re-use the PDI we have to transform the dual flow value  $\mu_{e,i}$  to  $\mu_{e,i'}$ . Since exactly one  $i \in \mathbb{N}_0$  is selected in any leaf problem we

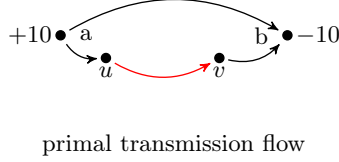
Consider two leaves of a given local problem. Then the two PDIs only differ in their actual choice of  $\alpha_{e,i}$  and  $\tilde{\beta}_{e,i}$ , because these selections depend on the actual branching decisions that lead from the local problem to the respective leaf problem. The corresponding branching decisions are settled by the integer variables  $x_{e,i}$  and  $y_{e,i}$ , respectively. Therefore we can formulate a “local” PDI which combines all different “leaf” PDIs by attaching these variables to the coefficients in the sums. This proves the validity of the desired linear inequality:

$$\begin{aligned} & (k+1) \sum_{v \in V} s_v \mu_v - \sum_{v \in V} (\lambda_v^+ \bar{\pi}_v - \lambda_v^- \underline{\pi}_v) \\ & \leq \sum_{\substack{(e,i) \in E''_X, \\ i \neq 0}} k_e \left( \frac{(\mu_v - \mu_w)^{k+1}}{\alpha_{e,i} \mu_e} \right)^{\frac{1}{k}} x_{e,i} - \sum_{\substack{(e,i) \in E''_X, \\ i \neq 0}} \mu_e \beta_{e,i} y_{e,i}. \end{aligned}$$

□

## 7 A Counterexample

In the proof of Theorem 1 we showed that a PDI gives a necessary condition for the feasibility of the leaf problem. It is natural to ask if the converse is also true, that is, if a valid PDI is also a sufficient condition for the feasibility of the leaf problem. We now give a counterexample to demonstrate that this is not true in general. From a polyhedral point-of-view it means that the integer convex hull of all feasible solutions cannot be described by PDI inequalities alone, because PDIs define valid inequalities but not facets in general.



**Fig. 7.1.** A counterexample network.

*Example 1.* We consider the network shown in Figure 7.1. This network consists of four nodes  $\{a, b, u, v\}$  and four pipelines  $e_1 = (a, b), e_2 = (a, u), e_3 = (u, v), e_4 = (v, b)$  and no active elements (compressors, valves, pressure regulators), and no topology extension pipes. The pipeline data is as follows:  $\alpha_1 = 3, \alpha_2 = \alpha_3 = \alpha_4 = 1, k_1 = \dots = k_4 = 1$ , and  $\beta_1 = \dots = \beta_4 = 0$ . Node  $a$  is an entry node with  $s_a = 10$  and node  $b$  is an exit with  $s_b = -10$ . Nodes  $u$  and  $v$  are intermediate nodes with  $s_u = s_v = 0$ . The pressure bounds are as follows:  $\bar{\pi}_a = 100, \bar{\pi}_u = 60, \bar{\pi}_v = 100, \bar{\pi}_b = 100$  and  $\underline{\pi}_a = 100, \underline{\pi}_u = 0, \underline{\pi}_v = 40, \underline{\pi}_b = 0$ .

The corresponding constraint system, which consists of pressure loss equations and flow conservation constraints, then is:

$$3q_1^2 = \pi_a, \quad (7.1)$$

$$q_2^2 = \pi_a - \pi_u, \quad (7.2)$$

$$q_3^2 = \pi_u - \pi_v, \quad (7.3)$$

$$q_4^2 = \pi_v, \quad (7.4)$$

$$q_1 + q_2 = 10, \quad (7.5)$$

$$q_3 - q_2 = 0, \quad (7.6)$$

$$q_4 - q_3 = 0, \quad (7.7)$$

$$-q_1 - q_4 = -10. \quad (7.8)$$

Solving this system algebraically yields the unique solution  $q_1 = \dots = q_4 = 5$ , and  $\pi_a = 100, \pi_u = 75, \pi_v = 50, \pi_b = 25$ . Note that this solution violates the upper bound in node  $u$  which was set to 60. Hence the leaf problem for this system, which additionally contains the flow bounds as trivial constraints, would be infeasible. We now show that a PDI for any dual transmission flow  $(\mu, \lambda)$  is not violated, hence it cannot be used as a sufficient condition.

Let  $(\mu, \lambda)$  be an arbitrary dual transmission flow. We have to show that the corresponding PDI

$$75\mu_1\tilde{\delta}_1 + 25\mu_2\tilde{\delta}_2 + 25\mu_3\tilde{\delta}_3 + 25\mu_4\tilde{\delta}_4 \leq 100\lambda_a^+ - 100\lambda_a^- + 60\lambda_u^+ + 100\lambda_v^+ - 40\lambda_v^- + 100\lambda_b^+ \quad (7.9)$$

is valid. Note that  $\lambda_a^- = \lambda_b^+ = 0$ , since these two nodes are entry and exit, respectively. The general idea is to show that there are numbers  $A$  and  $B$  estimating the difference between left- and right-hand side of inequality (7.9) such that

$$75\mu_1\tilde{\delta}_1 + 25\mu_2\tilde{\delta}_2 + 25\mu_3\tilde{\delta}_3 + 25\mu_4\tilde{\delta}_4 - (100\lambda_a^+ + 60\lambda_u^+ + 100\lambda_v^+ - 40\lambda_v^-) \leq A \leq B \leq 0. \quad (7.10)$$

To derive  $A$ , we split the dual arc flow  $\mu_e$  into a sum of path flows. Denote by  $f_P$  the flow on path  $P \in \mathcal{P} := \{P_1, \dots, P_6\}$ , where  $P_1 := (a, u), P_2 := (u, v), P_3 := (v, b), P_4 := (a, u, v), P_5 := (u, v, b), P_6 := (a, u, v, b)$ . We then obtain

$$\mu_2 = f_{P_1} + f_{P_4} + f_{P_6}, \quad (7.11)$$

$$\mu_3 = f_{P_2} + f_{P_4} + f_{P_5} + f_{P_6}, \quad (7.12)$$

$$\mu_4 = f_{P_3} + f_{P_5} + f_{P_6}. \quad (7.13)$$

By the definition of the dual transmission flow, i.e. equation (5.4), we get the system of equations

$$\lambda_1^+ - \lambda_1^- = f_{P_1} + f_{P_4} + f_{P_6}, \quad (7.14)$$

$$\lambda_2^+ - \lambda_2^- = f_{P_2} + f_{P_5} - f_{P_1}, \quad (7.15)$$

$$\lambda_3^+ - \lambda_3^- = f_{P_3} - f_{P_2} - f_{P_4}, \quad (7.16)$$

$$\lambda_4^+ - \lambda_4^- = -f_{P_3} - f_{P_5} - f_{P_6}, \quad (7.17)$$

By definition of the dual transmission flow at most one of  $\lambda_i^+, \lambda_i^-$  (for  $i = 1, \dots, 4$ ) can be positive. Thus we can split the system (7.14) into the following system:

$$\lambda_1^+ = f_{P_1} + f_{P_4} + f_{P_6}, \quad \lambda_1^- = 0, \quad (7.18)$$

$$\lambda_2^+ = f_{P_2} + f_{P_5}, \quad \lambda_2^- = f_{P_1}, \quad (7.19)$$

$$\lambda_3^+ = f_{P_3}, \quad \lambda_3^- = f_{P_2} + f_{P_4}, \quad (7.20)$$

$$\lambda_4^+ = 0, \quad \lambda_4^- = f_{P_3} + f_{P_5} + f_{P_6}. \quad (7.21)$$

We multiply the  $i$ -th left equation in (7.18) by  $\bar{\pi}_i$  and the  $i$ -th right equation in (7.18) by  $-\underline{\pi}_i$  and sum up all these scaled equations to obtain

$$100\lambda_a^+ + 60\lambda_u^+ + 100\lambda_v^+ - 40\lambda_v^- = 100\mu_1 + 100f_{P_1} + 20f_{P_2} + 100f_{P_3} + 60f_{P_4} + 60f_{P_5} + 100f_{P_6}. \quad (7.22)$$

Furthermore, we obtain

$$75\mu_1\tilde{\delta}_1 + 25\mu_2\tilde{\delta}_2 + 25\mu_3\tilde{\delta}_3 + 25\mu_4\tilde{\delta}_4 \quad (7.23)$$

$$= 75\mu_1\tilde{\delta}_1 + 25\tilde{\delta}_2f_{P_1} + 25\tilde{\delta}_3f_{P_2} + 25\tilde{\delta}_4f_{P_3} \quad (7.24)$$

$$+ 25(\tilde{\delta}_2 + \tilde{\delta}_3)f_{P_4} + 25(\tilde{\delta}_3 + \tilde{\delta}_4)f_{P_5} + 25(\tilde{\delta}_2 + \tilde{\delta}_3 + \tilde{\delta}_4)f_{P_6}. \quad (7.25)$$

We are now putting (7.22) and (7.23) together into (7.9). Since  $\tilde{\delta}_i \leq 1$  (for  $i = 1, \dots, 4$ ) we can omit some terms less-or-equal zero and obtain the following estimation:

$$75\mu_1\tilde{\delta}_1 + 25\tilde{\delta}_2f_{P_1} + 25\tilde{\delta}_3f_{P_2} + 25\tilde{\delta}_4f_{P_3} \quad (7.26)$$

$$+ 25(\tilde{\delta}_2 + \tilde{\delta}_3)f_{P_4} + 25(\tilde{\delta}_3 + \tilde{\delta}_4)f_{P_5} + 25(\tilde{\delta}_2 + \tilde{\delta}_3 + \tilde{\delta}_4)f_{P_6} \quad (7.27)$$

$$- (100\mu_1 + 100f_{P_1} + 20f_{P_2} + 100f_{P_3} + 60f_{P_4} + 60f_{P_5} + 100f_{P_6}) \quad (7.28)$$

$$\leq (75\tilde{\delta}_1 - 100)\mu_1 + (25\tilde{\delta}_3 - 20)(f_{P_2} + f_{P_4} + f_{P_5} + f_{P_6}) \quad (7.29)$$

$$= (75\tilde{\delta}_1 - 100)\mu_1 + (25\tilde{\delta}_3 - 20)\mu_3 =: A. \quad (7.30)$$

We analyze the two summands of  $A$  which has the general form  $(\xi_e\tilde{\delta}_e - \eta_e)\mu_e$  (for  $e = 1, 3$ ). Using the definition of  $\tilde{\delta}_e$  we get the following function in  $\mu_e$  (for edge  $e = (i, j)$ ):

$$\mu_e \mapsto \left( \xi_e \left( 2\frac{\mu_i - \mu_j}{\alpha_e q_e \mu_e} - \left( \frac{\mu_i - \mu_j}{\alpha_e q_e \mu_e} \right)^2 \right) - \eta_e \right) \mu_e. \quad (7.31)$$

Elementary calculus shows that this function has a global maximum in

$$\mu_e^* = \frac{\mu_i - \mu_j}{\alpha_e q_e} \sqrt{\frac{\xi_e}{\eta_e}}. \quad (7.32)$$

Putting (7.32) in (7.31) yields the following estimation for  $A$ :

$$A \leq 2\frac{\mu_a - \mu_b}{\alpha_1 q_1} \left( \xi_1 - \sqrt{\xi_1 \eta_1} \right) + 2\frac{\mu_u - \mu_v}{\alpha_3 q_3} \left( \xi_3 - \sqrt{\xi_3 \eta_3} \right). \quad (7.33)$$

Because of (5.2) we have that  $\mu_a \geq \mu_u \geq \mu_v \geq \mu_b$ . Using the numerical values for the parameters we obtain the following estimation.

$$A \leq -0.4912 \dots \cdot (\mu_a - \mu_b) =: B \leq 0. \quad (7.34)$$

Hence we showed that every PDI is a valid inequality, but there is no feasible primal transmission flow.

## 8 Dual Transmission Flows from KKT Systems

Dual transmission flows are obtained during the branch-and-bound process that solves the topology optimization problem. To this end we branch until we are at a leaf problem (5.1). Then we consider the following relaxation of this leaf:

$$\begin{aligned}
\min \quad & \sum_{v \in V} \Delta_v + \sum_{(e,i) \in E'_X, i \neq 0} \Delta_{e,i} \quad \text{s. t.} \\
& \alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i} - (\pi_v - \pi_w) = 0 \quad \forall (e,i) \in E'_X, i \neq 0, \\
& \sum_{\substack{w,i:(w,v,i) \in E'_X \\ i \neq 0}} q_{w,v,i} - \sum_{\substack{w,i:(v,w,i) \in E'_X \\ i \neq 0}} q_{v,w,i} = s_v \quad \forall v \in V, \\
& \pi_v - \Delta_v \leq \bar{\pi}_v \quad (v \in V), \\
& \pi_v + \Delta_v \geq \underline{\pi}_v \quad (v \in V), \\
& \Delta_v \geq 0 \quad (v \in V)
\end{aligned} \tag{8.1}$$

That is, all variables are unbounded and we minimize the sum of the slack variables  $\Delta_v$ . Numerically we solve the nonlinear problem (8.1) using IPOPT. If the solver converges to local optimum, we additionally obtain a set of Lagrange parameters that fulfill the KKT conditions. We now show that these Lagrange parameters can easily be modified such that these modified values are a dual transmission flow, hence we can use them to derive a valid cut for a local topology optimization problem.

For some Lagrange parameters  $(\mu, \lambda) = (\mu_{e,i}, \mu_v, \lambda_v^+, \lambda_v^-, \lambda_v)_{v \in V, (e,i) \in E'_X, i \neq 0}$ , such that  $\mu_{e,i}, \mu_v \in \mathbb{R}$  and  $\lambda_v^+, \lambda_v^-, \lambda_v \in \mathbb{R}_{\geq 0}$ , the Lagrangian of problem (8.1) has the form

$$\begin{aligned}
L(q, \pi, \Delta, \mu, \lambda) = & \sum_{(e,i) \in E'_X, i \neq 0} \mu_{e,i} \left( \alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i} - (\pi_v - \pi_w) \right) \\
& + \sum_{v \in V} \mu_v \left( s_v - \sum_{\substack{w,i:(w,v,i) \in E'_X \\ i \neq 0}} q_{w,v,i} + \sum_{\substack{w,i:(v,w,i) \in E'_X \\ i \neq 0}} q_{v,w,i} \right) \\
& + \sum_{v \in V} (\lambda_v^+ (\pi_v - \Delta_v - \bar{\pi}_v) + \lambda_v^- (\underline{\pi}_v - \pi_v - \Delta_v)) \\
& - \sum_{v \in V} \lambda_v \Delta_v
\end{aligned} \tag{8.2}$$

Consider now values  $(q, \pi, \Delta, \mu, \lambda)$  as local optimal values for (8.1) and their duals, fulfilling the KKT conditions. This implies

$$\nabla_{(q,\pi,\Delta)} L(q, \pi, \Delta, \mu, \lambda) = 0,$$

which means that for the dual variables  $\mu, \lambda$  the following holds:

$$\mu_{e,i} \nabla (\alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i}) = \mu_v - \mu_w \quad ((e,i) \in E'_X, i \neq 0), \tag{8.3}$$

$$\sum_{\substack{w,i:(w,v,i) \in E'_X \\ i \neq 0}} \mu_{(w,v),i} - \sum_{\substack{w,i:(v,w,i) \in E'_X \\ i \neq 0}} \mu_{(v,w),i} = \lambda_v^+ - \lambda_v^- \quad (v \in V). \tag{8.4}$$

We set

$$\tilde{\mu}_{e,i} := \begin{cases} (k+1)\mu_{e,i}, & \mu_v \neq \mu_w \\ 0, & \mu_v = \mu_w \end{cases} \quad \forall ((v,w),i) = (e,i) \in E'_X, i \neq 0,$$

and  $\tilde{\mu}_v := \mu_v$  for all  $v \in V$ . With these settings the conditions (5.2) and (5.3) of Definition 2 are fulfilled. Then condition (5.4) implicitly defines new values for  $\tilde{\lambda}_v^+, \tilde{\lambda}_v^-$  for  $v \in V$  (note that we still keep the property that at most one of the two is non-zero).



Using the dual transmission flow derived from the KKT system as described above we obtain a valid inequality for a local topology optimization problem. To find the “right” local problem it is necessary to analyze the flow directions of the dual transmission flow  $\mu_{e,i}$ . The cut is now valid for the local problem having the same flow directions as indicated by the signums of  $\mu_{e,i}$ .

## 9 Globally Valid Cuts

In the discussion so far we derived a valid cut at a leaf problem, that was valid for the corresponding local problem. In this section we will show that we can obtain a valid cut at a leaf problem, that is globally valid (i.e., valid also for the root node). However, we have to make stronger assumptions, hence not every dual feasible solution at a leaf will lead to such cut.

We proceed as in the previous section by solving the relaxation (8.1) of a leaf problem. Assuming that the nonlinear solver converges to a local optimum with a positive objective function value (that is, the solution is infeasible for the topology optimization problem), we obtain a KKT point  $q_{e,i}^*, \pi_v^*, \Delta_v^*, \mu_{e,i}^*, \mu_v^*, \lambda_v^{+,*}, \lambda_v^{-,*}$  for all  $(e, i) \in E'_X$  and  $v \in V$ , respectively. Note that for each  $e \in E_X$  exactly one arc  $(e, i)$  is chosen in  $E'_X$ . Hence as an abbreviation we can set  $q_e^* := q_{e,i}^*$  and  $\mu_e^* := \mu_{e,i}^*$ .

**Lemma 3.** *Let  $q, x, y$  be a primal solution for the topology optimization problem (4.2). Then the following inequality is valid:*

$$\sum_{v \in V} s_v \mu_v^* + \sum_{(e,i) \in E'_X, i \neq 0} (\mu_e^* + (q_{e,i} - q_e^*)) (\alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i}) \quad (9.1)$$

$$- \sum_{(e,i)=(v,w,i) \in E'_X, i \neq 0} q_{e,i} (\mu_v^* - \mu_w^*) \leq \sum_{v \in V} \lambda_v^{+,*} \bar{\pi}_v - \lambda_v^{-,*} \underline{\pi}_v. \quad (9.2)$$

Note that the  $(x, y)$ -part of the primal solution in Lemma 3 also defines a leaf problem. This is in general a different leaf than the one that we used to obtain the KKT point above.

Proof: From the flow conservation it follows that

$$\sum_{v \in V} s_v \mu_v^* = \sum_{(e,i) \in E'_X} q_{e,i} (\mu_v^* - \mu_w^*). \quad (9.3)$$

Then from Lemma 1 with  $\tilde{\delta}_e = 1$  we obtain

$$\sum_{(e,i) \in E'_X} \mu_e^* (\alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i}) \leq \sum_{v \in V} \lambda_v^{+,*} \bar{\pi}_v - \lambda_v^{-,*} \underline{\pi}_v. \quad (9.4)$$

Note that  $q_{e,i}$  for all  $(e, i) \in E'_X$  defines a feasible flow from the sources to the sinks in the network. The same hold for  $q_e^*$  for all  $e \in E$ . Hence the flow  $q_{e,i} - q_e^*$  is a circulation in the network. There exists a set of cycles  $\mathcal{C}$  such that the flow on cycle  $C \in \mathcal{C}$  is  $q_C$  and for all  $(e, i) \in E'_X$  it holds that

$$\sum_{C \in \mathcal{C}: e \in C} q_C = q_{e,i} - q_e^*. \quad (9.5)$$

From this follows

$$\begin{aligned} \sum_{(e,i) \in E'_X} (q_{e,i} - q_e^*) (\alpha_{e,i} q_{e,i} |q_{e,i}|^{k_e} + \tilde{\beta}_{e,i}) &= \sum_{(e,i)=(v,w,i) \in E'_X} (q_{e,i} - q_e^*) (\pi_v - \pi_w) \\ &= \sum_{(e,i)=(v,w,i) \in E'_X} \left( \sum_{C \in \mathcal{C}: e \in C} q_C \right) (\pi_v - \pi_w) \\ &= \sum_{C \in \mathcal{C}} q_C \underbrace{\sum_{(e,i)=(v,w,i) \in C} (\pi_v - \pi_w)}_0 = 0. \end{aligned}$$

Putting these equations together, the lemma follows.  $\square$

We introduce  $E_X''$  as the set of all arcs from  $E_X$  that are oriented in such way that the flow  $q_e^*$  is nonnegative on all arcs and  $\mu_v^* - \mu_w^* \geq 0$  for the closed valves. Let  $\hat{E}_X''$  be the set of arcs in  $E_X''$  that are not bridges together with the set of arcs from  $E_X$  that are closed valves in this solution. Since for the closed valves  $(e, 0) = (v, w, 0)$  there does not exist the variables  $\mu_{e,0}$  we introduce them by setting  $\mu_{e,0} := 0$ . Now we make the following three additional assumptions:

- The KKT point satisfies  $\mu_e^* q_e^* \geq 0$  for all edges  $(e, i) \in \hat{E}_X''$
- The KKT point satisfies  $\mu_e^* \tilde{\beta}_{e,i} \leq 0$  for all edges  $(e, i) \in \hat{E}_X''$ .
- If  $\mu_e^* = 0$  then  $\tilde{\beta}_{e,i} = 0$  for all edges  $(e, i) \in \hat{E}_X''$ .

With loss of generality we can assume that  $0 \leq \mu_e^* \leq q_e^*$  for all edges  $e \in \hat{E}_X''$ , that is, we set  $\theta := \min\{q_e^*/\mu_e^* : (e, i) \in \hat{E}_X''\}$  and replace all  $\mu_e^*$  by  $\theta\mu_e^*$  for all  $(e, i) \in E_X''$  (including the bridges).

The next goal is to make the valid inequality from Lemma 3 as strong as possible. This is achieved by minimizing the left-hand side, which means we need to find values  $q_{e,i}$  for all  $(e, i) \in E_X'$  that minimize each summand individually.

**Lemma 4.** For all  $(e, i) \in \hat{E}_X''$  the function

$$\rho \mapsto (\mu_e^* + (\rho - q_e^*))(\alpha_{e,i}\rho|\rho|^{k_e} + \tilde{\beta}_{e,i}) - \rho(\mu_v^* - \mu_w^*) \quad (9.6)$$

has a unique global minimum  $\rho^*$  with  $\rho^* \geq 0$ .

From elementary calculus it is easy to see that this minimum satisfies

$$(k_e + 2)\alpha_{e,i}(\rho^*)^{k_e+1} + (k_e + 1)(\mu_e^* - q_e^*)(\rho^*)^{k_e} - (\mu_v^* - \mu_w^* - \tilde{\beta}_{e,i}) = 0. \quad (9.7)$$

Proof: We split the function into four summands:

$$\rho \mapsto \alpha_{e,i}\rho^2|\rho|^{k_e} \quad (9.8)$$

$$+(\mu_e^* - q_e^*)(\alpha_{e,i}\rho|\rho|^{k_e}) \quad (9.9)$$

$$-\rho(\mu_v^* - \mu_w^* - \tilde{\beta}_{e,i}) \quad (9.10)$$

$$+(\mu_e^* - q_e^*)\tilde{\beta}_{e,i}. \quad (9.11)$$

The first summand (9.8) is non-negative in  $\rho$  and will grow to  $+\infty$  if  $\rho$  goes to either  $+\infty$  or  $-\infty$ . The second summand (9.9) either equals zero if  $\mu_e^* - q_e^* = 0$  or it will grow to  $+\infty$  if  $\rho$  tends to  $-\infty$  and will grow to  $-\infty$  if  $\rho$  tends to  $+\infty$ , because we assumed  $\mu_e^* - q_e^* \leq 0$ . Furthermore it has a lesser exponent than the first summand. It holds  $0 \leq \text{sign}(\mu_e^*) \leq \text{sign}(\mu_v^* - \mu_w^*)$ . Moreover we assumed that  $\mu_e^* \tilde{\beta}_{e,i} \leq 0$ . Hence with  $\mu_e^* \geq 0$ , we have that  $\tilde{\beta}_{e,i} \leq 0$ , and thus  $\mu_v^* - \mu_w^* - \tilde{\beta}_{e,i} \geq 0$  (here we also used the third assumption from above). Putting this together, if all coefficients are nonzero, we have that the function has a unique global minimum  $\rho^*$ . The last summand is constant (9.11) and thus can be neglected.  $\square$

The next step is to further modify all  $\mu_e^*, \mu_v^*$ , so that the minimum  $\rho_e^*$  from Lemma 4 is achieved for  $\rho_e^* = q_e^*$ . If this is the case, then the valid inequality from Lemma 3 reduces to

$$\sum_{(e,i) \in E_X'} \mu_e^* (\alpha_{e,i} q_e^* |q_e^*|^{k_e} + \tilde{\beta}_{e,i}) \leq \sum_{v \in V} \lambda_v^{+,*} \bar{\pi}_v - \lambda_v^{-,*} \underline{\pi}_v. \quad (9.12)$$

Since we started with an infeasible solution for the topology optimization problem (remember that there was a positive objective function value for the solution  $q^*$ ), it shows that this valid inequality is indeed violated by the current extension decisions, and (at least) the leaf node is cut off.

The modification is actually carried out by solving a suitable linear programming problem. This LP is as follows:

$$\begin{aligned} \sum_{e \in \delta^+(v)} \mu_{e,i} - \sum_{e \in \delta^-(v)} \mu_{e,i} &= \lambda_v^+ - \lambda_v^- \quad (v \in V), \\ (k_e + 2)\alpha_{e,i}(q_e^*)^{k_e+1} + (k_e + 1)(\mu_{e,i} - q_e^*)(q_e^*)^{k_e} & \end{aligned}$$

$$\begin{aligned}
-(\mu_v - \mu_w - \tilde{\beta}_{e,i}) &= 0, & ((e, i) = (v, w, i) \in \hat{E}''_X) \\
0 &\leq \mu_{e,i}, & ((e, i) \in \hat{E}''_X) \\
\mu_{e,i} &\leq q_e^*, & ((e, i) \in \hat{E}''_X) \\
\mu_{e,i} &\in \mathbb{R}_+, & ((e, i) \in \hat{E}''_X) \\
\mu_v &\in \mathbb{R}, & (v \in V).
\end{aligned}$$

Note that this LP has no objective function, we just search for a feasible solution (note that the LP could be infeasible), which corresponds to the modified coefficients mentioned before. Since this LP contains the optimality condition (9.7) as a constraint, it is assured that the coefficients are modified in such way that  $q^* = (q_e^*)_{e \in \hat{E}''_X}$  becomes a global minimum.

## 10 Computational Results

We consider two different transmission networks with different topologies. Both networks consist of pipelines only. The first network **Net1** has 20 nodes and 29 pipelines, which represent the backbone network of the Belgium natural gas network. The data for this network can be found in [13], and results using this network in De Wolf et al. [11, 9]. The second network **Net2** consists of 32 nodes and 37 pipelines. It is an approximation to the Germany L-gas network in the Rhine-Main-Ruhr area. We consider a single transport scenario for each network. To obtain different instances from the two basis network data, we scale these scenarios by multiplying each node flow by the same amount. In addition to that we allow a different maximal number of loops to be built, where each loop adds a pipe of the same characteristics (diameter, length) as the original pipe to the network. We distinguish between results that were solved to proven global optimality within a given time limit of 12 hours and those that did not. In the latter case an optimality gap remains, that is defined as the ratio between the best upper bound  $u$  (i.e., the objective function value of best feasible solution) and the best lower bound  $\ell$ . That is,  $\text{gap} = \frac{u-\ell}{\ell} \cdot 100\%$ .

We implemented the cut generation algorithms described in section 9 in C on a cluster of 64bit Intel Xeon X5672 CPUs at 3.20 GHz with 12 MByte cache and 48 GB main memory, running an OpenSuse 12.1 Linux with a gcc 4.6.2 compiler. We used the following software packages: SCIP 3.0.1 as mixed-integer nonlinear branch-and-cut framework (for details on SCIP we refer to [1]), CPLEX 12.1 [16] as linear programming solver, Ipopt 3.10 [26] as nonlinear solver, and Lamatto++ [14] as framework for handling the input data. Hyperthreading and Turboboost were disabled. In all experiments, we ran only one job per node to avoid random noise in the measured running time that might be caused by cache-misses if multiple processes share common resources. In our initial implementation we added all obtained cuts directly to the branch-and-bound process. In the final implementation, of which we report the results below, we do not add them immediately. Instead we store them in a cut pool, until a predefined number of inequalities is reached (experimentally, a pool size of 40 inequalities turned out to be a good value). Then, we restart the whole branch-and-cut solution process and multiply the cut pool size by 1.5. For the restart, only the best feasible solution and the valid inequalities are kept. Now the solver SCIP uses its proprietary routines to further strengthen our cuts (together with all other model inequalities). This presolve feature would otherwise not be available, and it contributes significantly to the overall performance of our method.

Given is a (balanced) flow demand at the entry (source) and exit (sink) nodes of the network. For this given demand, there exists a feasible flow in the network. Now we scale up this demand, that is, we multiply each entry and exit value by the same scalar  $> 1$ . For a certain value (2.0 for instance **net1** and 2.1 for instance **net2**) the instance is not longer feasible, i.e., there is no valid flow that fulfills all model constraints. In order to obtain a feasible flow again, the network topology needs to be extended, for which we introduce a number of loops. For instance **net1** we allow loops in  $\{7, 8, 9, 11\}$ , respectively, and for instance **net2** we allow between 2 and 4 loops, respectively. That is, each original pipeline can be extended by at most this number of pipelines having the same characteristic as the original ones.

The computational results are shown in the Tables 1–4. The first column shows the scaling factor of the demand vector. The second column gives the number of loops that can maximally be selected by the topology optimization. We compare the results that achieved with our cuts against the results that are achieved without our cuts. In the latter case, the model is solved using the default settings of the solver SCIP. For each run we report the optimality gap (as defined above). A gap of 0.0% means that the solver

was able to find a proven global optimal solution within the time limit of 12 hours. Furthermore we report in column “primal” the objective function value of the best feasible solution, i.e., the building cost for the chosen loop pipes, and in column “dual” the best lower bound. Note that in Table 1 and Table 3 these two columns are merged into one single “primal/dual” column, because both values coincide. In the column “time” we report the CPU time (runtime) in seconds, and column “nodes” shows the number of nodes that were solved during the branch-and-cut process. Since we use multiple restarts when adding our cuts, this “node” value refers to the total number of all nodes, summing up the individual restarted solution processes. The rows in Table 2 and Table 4 are ordered such that we first have those instances that can be solved to global optimality when using our cuts, then those instances where the gap can be reduced when using our cuts, and finally those instances where we do not have a gap (because of no primal feasible solutions).

For those instance that were solved to optimality within the time limit, Table 1 and Table 3 show a huge reduction of the nodes of the branch-and-bound tree and the runtime for almost all instances. Practically speaking, an instance is more computationally demanding, the more potential loops are offered and the higher the scaling factor. The runtime and node reduction becomes more significant, the more demanding the instance. Table 2 shows that all instances of **net1** that reached the time limit can be solved to proven global optimality when using our inequalities. Table 4 shows that for all instances of **net2** that reached the time limit, the gap was reduced and much better primal feasible solutions were found. Infeasibility could be proven for a one instance which was not possible without the cuts.

scale	loops	no cuts				with cuts			
		gap	primal/dual	time	nodes	gap	primal/dual	time	nodes
2.0	7	0.0	1500.00	254.07	185166	0.0	1500.00	192.79	13614
2.0	8	0.0	1500.00	241.41	172182	0.0	1500.00	246.80	17894
2.0	9	0.0	1500.00	128.04	88381	0.0	1500.00	457.48	8523
2.0	11	0.0	1500.00	151.74	85406	0.0	1500.00	465.63	9540
2.1	7	0.0	1800.00	1799.87	1105397	0.0	1800.00	140.31	32521
2.1	8	0.0	1800.00	1126.88	748074	0.0	1800.00	250.69	33848
2.1	9	0.0	1700.00	789.95	452687	0.0	1700.00	415.26	49310
2.1	11	0.0	1700.00	810.44	464263	0.0	1700.00	543.65	20323
2.2	7	0.0	2200.00	29884.35	6107455	0.0	2200.00	214.04	94222
2.2	8	0.0	2000.00	4246.18	2508963	0.0	2000.00	306.48	86210
2.2	9	0.0	2000.00	2174.16	1337687	0.0	2000.00	382.74	52029
2.2	11	0.0	2000.00	3685.62	1749485	0.0	2000.00	454.71	40221
2.3	9	0.0	2300.00	14512.47	6411628	0.0	2300.00	411.34	41425
2.3	11	0.0	2200.00	10301.17	4792160	0.0	2200.00	887.75	94600
2.5	7	0.0	inf	0.93	1	0.0	inf	0.71	1
Average				5007.59	1872066			383.54	42448

**Table 1.** Instances for **Net1** solved to proven global optimality within the time limit.

scale	loops	no cuts					with cuts				
		gap	primal	dual	time	nodes	gap	primal	dual	time	nodes
2.3	7	74.54	3100.00	1776.14	43200.01	5568694	0.0	3000.00	3000.00	306.40	72166
2.3	8	37.37	2500.00	1819.93	43200.04	7823255	0.0	2400.00	2400.00	207.48	39124
2.4	8	66.77	3200.00	1918.79	43200.00	6512046	0.0	3100.00	3100.00	421.02	138860
2.4	9	47.18	2700.00	1834.49	43200.00	4699381	0.0	2700.00	2700.00	435.70	173544
2.4	11	28.79	2500.00	1941.16	43200.00	8597518	0.0	2500.00	2500.00	744.82	107385
2.5	8	142.03	5200.00	2148.48	43200.02	7643287	0.0	4400.00	4400.00	18868.35	4439886
2.5	9	73.09	3500.00	2022.04	43200.02	4719944	0.0	3300.00	3300.00	519.46	131165
2.5	11	43.29	2900.00	2023.86	43200.01	7192021	0.0	2900.00	2900.00	838.69	198226
2.4	7	116.82	5800.00	2675.04	43203.87	6310160	17.68	5100.00	4333.79	43200.00	4904754

**Table 2.** Instances for **Net1** that reached the time limit of 12 hours (without cuts).

scale	loops	no cuts				with cuts			
		gap	primal/dual	time	nodes	gap	primal/dual	time	nodes
2.1	2	0.0	100.00	1.92	145	0.0	100.00	3.35	80
2.1	3	0.0	100.00	2.67	188	0.0	100.00	1.66	11
2.1	4	0.0	100.00	6.77	643	0.0	100.00	1.25	10
2.2	2	0.0	200.00	2.94	288	0.0	200.00	7.77	265
2.2	3	0.0	200.00	6.38	453	0.0	200.00	27.64	979
2.2	4	0.0	200.00	7.38	800	0.0	200.00	33.12	929
2.3	2	0.0	200.00	3.09	316	0.0	200.00	25.90	1158
2.3	3	0.0	200.00	5.43	783	0.0	200.00	14.00	323
2.3	4	0.0	200.00	2.93	265	0.0	200.00	65.26	1793
2.4	2	0.0	300.00	2.31	148	0.0	300.00	19.44	1266
2.4	3	0.0	300.00	4.44	280	0.0	300.00	52.47	2306
2.4	4	0.0	300.00	5.03	676	0.0	300.00	66.36	2265
2.5	2	0.0	400.00	11.59	3873	0.0	400.00	4.77	115
2.5	3	0.0	400.00	7.15	993	0.0	400.00	43.92	2132
2.5	4	0.0	400.00	8.56	1525	0.0	400.00	85.32	3092
2.6	2	0.0	500.00	11.07	4518	0.0	500.00	18.42	1462
2.6	3	0.0	500.00	7.24	2579	0.0	500.00	45.60	3688
2.6	4	0.0	500.00	6.97	2708	0.0	500.00	79.07	3787
2.7	2	0.0	700.00	69.98	39543	0.0	700.00	35.35	4744
2.7	3	0.0	600.00	47.16	28908	0.0	600.00	49.26	8506
2.7	4	0.0	600.00	22.80	13154	0.0	600.00	81.32	11078
2.8	2	0.0	800.00	191.10	122284	0.0	800.00	46.66	9889
2.8	3	0.0	800.00	281.03	201326	0.0	800.00	68.01	12916
2.8	4	0.0	800.00	198.71	142382	0.0	800.00	77.57	10791
2.9	2	0.0	1000.00	590.47	458810	0.0	1000.00	111.45	35947
2.9	3	0.0	900.00	1677.92	876756	0.0	900.00	96.81	28233
2.9	4	0.0	900.00	1547.63	999010	0.0	900.00	122.00	33291
3.0	2	0.0	1300.00	12503.69	8259949	0.0	1300.00	284.36	96166
3.0	3	0.0	1100.00	11070.69	3411710	0.0	1100.00	177.06	72224
3.0	4	0.0	1100.00	9478.49	5168230	0.0	1100.00	246.04	98727
3.1	4	0.0	1200.00	18351.30	8689438	0.0	1200.00	270.89	162622
Average				1807.45	917142			72.36	19691

**Table 3.** Instances for Net2 solved to proven global optimality within the time limit.

scale	loops	no cuts					with cuts				
		gap	primal	dual	time	nodes	gap	primal	dual	time	nodes
3.1	2	74.51	1900.00	1088.78	43200.00	24310579	0.0	1700.00	1700.00	11504.08	1064839
3.1	3	59.39	1300.00	815.62	43200.01	6705275	0.0	1300.00	1300.00	382.19	135622
3.2	3	104.11	1700.00	832.89	43200.10	7202393	0.0	1500.00	1500.00	768.38	279232
3.2	4	44.02	1400.00	972.08	43200.00	19961093	0.0	1400.00	1400.00	543.53	263223
3.3	3	128.89	2100.00	917.46	43200.11	9937525	0.0	1800.00	1800.00	22859.39	1726252
3.3	4	94.11	1800.00	927.30	43200.02	15223854	0.0	1600.00	1600.00	4548.17	910943
3.4	2	n/a	n/a	1758.70	43200.02	42118486	0.0	inf	inf	6383.27	1657450
3.4	4	85.28	1900.00	1025.45	43200.00	18360442	0.0	1800.00	1800.00	5982.06	1836888
3.2	2	191.05	3800.00	1305.59	43200.02	36089812	22.94	2200.00	1789.48	43200.03	2230710
3.3	2	314.25	6200.00	1496.67	43200.02	29401705	42.25	3400.00	2390.16	43200.01	2370429
3.4	3	141.85	2600.00	1075.02	43200.00	11458124	28.31	2200.00	1714.61	43200.02	2426174
3.5	3	181.25	3300.00	1173.31	43200.03	13402596	17.96	2500.00	2119.37	43200.02	2956148
3.5	4	132.47	2300.00	989.35	43200.01	7882263	21.6	2100.00	1726.97	43200.01	3415259
3.6	3	146.73	3200.00	1296.96	43200.01	21899795	29.46	3100.00	2394.56	43200.01	3472967
3.6	4	169.4	2900.00	1076.46	43200.00	14454356	27.13	2500.00	1966.44	43200.02	3585688
3.7	3	247.22	4800.00	1382.41	43200.03	13851067	26.28	3700.00	2929.98	43200.00	2475721
3.7	4	177.81	3200.00	1151.88	43200.01	15117237	30.4	2900.00	2223.88	43200.00	4183878
3.8	3	407.61	7500.00	1477.51	43200.03	11641971	38.19	5000.00	3618.27	43200.00	2513997
3.8	4	261.16	4400.00	1218.29	43200.01	14417166	25.61	3200.00	2547.61	43200.01	3404380
3.9	4	257.19	4500.00	1259.81	43200.03	13002158	33.86	3800.00	2838.79	43200.01	3995685
3.5	2	n/a	n/a	1866.95	43200.02	15418008	n/a	n/a	1828.87	43200.00	20790826
3.6	2	n/a	n/a	2362.24	43200.01	25438404	n/a	n/a	2207.23	43200.04	17939336
3.7	2	n/a	n/a	3041.41	43200.00	26982073	n/a	n/a	2694.57	43200.00	6307730
3.9	3	n/a	n/a	1629.09	43200.02	10632772	n/a	n/a	4733.85	43200.00	3364126

**Table 4.** Instances for **Net2** that reached the time limit of 12 hours (without cuts).

## 11 Conclusions and Outlook

We considered a network design and extension problem for natural gas transmission networks, where the gas flow is not only governed by flow conservation constraints in nodes (as in classical linear network flow problems), but also subject to nonlinear functions depending on node potential values (which represent the gas pressure in the nodes). We gave a model formulation, and applied a standard MINLP solver for its solution. We derived additional valid inequalities which depend on a certain linear flow problem where the model’s dual variables are used as input data. These cuts are separated during the branch-and-bound solution process in those subnodes where all decision variables on active elements are already settled by branching decisions. We could demonstrate by numerical experiments on a test set of instances that the separation of these new inequalities leads to significant smaller branch-and-bound trees and thus lower overall running times.

So far, we considered loop extensions only. From a practical point of view these kind of extensions are desired by network operations since they are cheaper than non-loop pipelines. From a computational point of view this restriction limits the amount of potential extensions. When adding all possible pipelines between all node pairs in the network, the model and the combinatorial decisions would be so large that the computations would not finish within a reasonable time limit, even when adding our cuts. Hence new techniques for making a reasonable selection of new pipes are necessary in the first place. This is one of our future research directions.

Further we investigate how to extend our cuts to active elements. When compressor stations or control valves for pressure regulation are among the network extension measures, it is unclear how to formulate a suitable cut that speeds up the solution process.

Finally, our method was applied to the case of a single invalid nomination. In practice, however, one has to deal with a whole set of different infeasible nominations, and needs to determine a topology extension that can cope with all of them simultaneously. The extension of our methods to this multi-scenario case is a further area of our current research.

## Acknowledgements

We are grateful to Open Grid Europe GmbH (OGE, Essen/Germany) for supporting our work.

## References

1. T. Achterberg. SCIP: Solving Constraint Integer Programs. *Mathematical Programming Computation*, 1(1):1–41, 2009.
2. J. André and J.F. Bonnans. Optimal features of gas transmission trunklines. In *EngOpt 2008 - International Conference on Engineering Optimization*, 2008. Rio de Janeiro, Brazil, June 1-5, 2008.
3. Timo Berthold, Stefan Heinz, and Stefan Vigerske. Extending a CIP framework to solve MIQCPs. In Jon Lee and Sven Leyffer, editors, *Mixed Integer Nonlinear Programming*, volume 154, part 6 of *The IMA Volumes in Mathematics and its Applications*, pages 427–444. Springer, 2012. Also available as ZIB-Report 09-23.
4. J. F. Bonnans and J. André. Optimal structure of gas transmission trunklines. Technical Report 6791, Institut National de Recherche en Informatique et en Automatique, January 2009.
5. Ian D. Boyd, Patrick D. Surry, and Nicholas J. Radcliffe. Constrained gas network pipe sizing with genetic algorithms. Technical Report EPCC-TR94-11, Edinburgh Parallel Computing Centre, 1994.
6. Luis Castillo and Antonio González. Distribution network optimization: Finding the most economic solution by using genetic algorithms. *European Journal of Operational Research*, 108(3):527–537, 1998.
7. M. Collins, L. Cooper, R. Helgason, J. Kennington, and L. LeBlanc. Solving the pipe network analysis problem using optimization techniques. *Management science*, 24(7):747–760, 1978.
8. A.R. Conn, N.I.M. Gould, and P.L. Toint. *Trust-Region Methods*. SIAM, Philadelphia, PA, USA, 2000.
9. D. De Wolf and B. Bakhouya. Optimal dimensioning of pipe networks: the new situation when the distribution and the transportation functions are disconnected. Technical Report 07/02, Ieseg, Université catholique de Lille, HEC Ecole de Gestion de l’ULG, July 2008.
10. D. de Wolf and Y. Smeers. Optimal dimensioning of pipe networks with application to gas transmission networks. *Operations Research*, 44(4):596–608, July-August 1996.
11. D. De Wolf and Y. Smeers. The gas transmission problem solved by an extension of the simplex algorithm. *Management Science*, 46(11):1454–1465, November 2000.
12. A. Fügenschuh, H. Homfeld, H. Schüllendorf, and S. Vigerske. Mixed-integer nonlinear problems in transportation applications. In H. Rodrigues et al., editor, *Proceedings of the 2nd International Conference on Engineering Optimization (CD-ROM)*, 2010.
13. General algebraic modeling system (GAMS) model library. <http://www.gams.com/modlib/modlib.htm/>.
14. B. Geißler, A. Martin, and A. Morsi. Lamatto++. information available at URL <http://www.mso.math.fau.de/edom/projects/lamatto.html>, February 2013.
15. A. Hazen and G. S. Williams. *Hydraulic Tables*. John Wiley and Sons, New York, 1920.
16. IBM Corporation, Armonk, USA. *User’s Manual for CPLEX*, 12.1 edition, 2011.
17. O. Mariani, F. Ancillai, and E. Donati. Design of a gas pipeline: Optimal configuration. Technical Report PSIG 9706, Pipeline Simulation Interest Group, 1997.
18. G. L. Nemhauser and L. A. Wolsey. Integer programming. In G. L. Nemhauser, A. H. G. Rinnooy Kan, and M. J. Todd, editors, *Optimization*, chapter 6, pages 447–527. Elsevier, 1989.
19. R. P. O’Neill, M. Williard, B. Wilkins, and R. Pike. A mathematical programming model for allocation of natural gas. *Operations Research*, 27(5):857–873, 1979.
20. A.J. Osiadacz and M. Górecki. Optimization of pipe sizes for distribution gas network design. Technical Report PSIG 9511, Pipeline Simulation Interest Group, 1995.
21. Marc E. Pfetsch, Armin Fügenschuh, Björn Geißler, Nina Geißler, Ralf Gollmer, Benjamin Hiller, Jesco Humpola, Thorsten Koch, Thomas Lehmann, Alexander Martin, Antonio Morsi, Jessica Rövekamp, Lars Schewe, Martin Schmidt, Rüdiger Schultz, Robert Schwarz, Jonas Schweiger, Claudia Stangl, Marc C. Steinbach, Stefan Vigerske, and Bernhard M. Willert. Validation of nominations in gas network optimization: Models, methods, and solutions. Technical report, ZIB-Report 12-41, Zuse Institute Berlin, Takustr.7, 14195 Berlin, 2012.
22. D.W. Schroeder. A tutorial on pipe flow equations. Technical Report PSIG 0112, Pipeline Simulation Interest Group, 2001.
23. E.M.B. Smith and C.C. Pantelides. A symbolic reformulation/spatial branch-and-bound algorithm for the global optimization of nonconvex MINLPs. *Computers & Chemical Engineering*, 23:457–478, 1999.
24. M. Tawarmalani and N.V. Sahinidis. Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming*, 99:563–591, 2004.
25. M. Tawarmalani and N.V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103:225–249, 2005.
26. A. Wächter and L. T. Biegler. On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming*, 106(1):25–57, 2006.
27. T. R. Weymouth. Problems in Natural Gas Engineering. *Transactions of the American Society of Mechanical Engineers*, 34:185–231, 1912.