RALF LENZ iD AND KAI HELGE BECKER

Optimization of Capacity Expansion in Potential-driven Networks including Multiple Looping – A comparison of modelling approaches

Optimization of Capacity Expansion in Potential-driven Networks including Multiple Looping – A comparison of modelling approaches

Ralf Lenz [iD]*and Kai Helge Becker†

Abstract

In commodity transport networks such as natural gas, hydrogen and water networks, flows arise from nonlinear potential differences between the nodes, which can be represented by so-called "potential-driven" network models. When operators of these networks face increasing demand or the need to handle more diverse transport situations, they regularly seek to expand the capacity of their network by building new pipelines parallel to existing ones ("looping"). The paper introduces a new mixed-integer non-linear programming (MINLP) model and a new non-linear programming (NLP) model and compares these with existing models for the looping problem and related problems in the literature, both theoretically and experimentally. On this basis, we give recommendations about the circumstances under which a certain model should be used. In particular, it turns out that one of our novel models outperforms the existing models with respect to computational time, the number of solutions found, the number of instances solved and cost savings. Moreover, the paper is the first to include the practically relevant option that a particular pipeline may be looped several times.

1   Introduction

Operators of commodity transport networks such as natural gas, hydrogen and water networks regularly have to face both increasing demand and the need to handle more diverse transport situations. To deal with these challenges without having to resort to the expensive options of setting up new pipeline corridors or of demolishing pipelines and replacing them by larger ones, they must expand the capacity of the existing network. Without using compressor stations to increase the pressure in the pipelines, this can only be achieved by building pipelines parallel to existing ones, a process that is referred to as "looping" in the terminology of the industry. Due to the high costs involved there has long been research about finding the cost minimal way of looping an existing network, i.e. of determining both the pipelines that are to be looped and the diameters of the pipelines to be built.

---

*Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany, `lenz@zib.de`
†Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany, `becker@zib.de`

Commodity flows such as flows in gas and water networks arise from friction-induced potential differences between the nodes in the network, which leads to a type of nonlinear network model that is referred to as potential-driven network in the literature ([BD56], [Rag13], [RSS$^+$19]). Apart from the nonlinear character of potential-driven networks, a specific difficulty in finding optimal loops arises from the fact that the diameters of the pipelines typically have to be selected from a discrete set of commercially available diameters ([ABC09], [HMN91], [Fas99]), which additionally imparts to the problem a combinatorial flavour. For these reasons the problem of capacity expansion in potential-driven networks belongs to the family of mixed-integer nonlinear programming (MINLP) problems.

As a consequence, a large body of the literature on the topic is concerned with developing sophisticated special-purpose algorithms geared at finding locally optimal solutions or approximating a global optimum, partly based on novel ways of modelling the capacity expansion problem. While this research focus has certainly advanced our ability to solve this problem, it has also led to a variety of mathematical models for the looping problem that remain unconnected in the literature. In view of the recent advances in the development of general-purpose solvers for NLP and MINLP problems that allow for an efficient implementation of models that is significantly faster than implementing one of the special-purpose algorithms developed, it seems to be useful for practitioners involved in the design of networks to shift the research focus to the modelling stage. For this reason, the present paper discusses various models for finding optimal loops in potential-driven networks.

In particular, it brings together, for the first time, the diversity of existing models in the literature and compares these, both theoretically and experimentally. For this comparison we choose a comprehensive approach that includes two different, rather unconnected strands of research on the problem that are referred to as the discrete approach and the split-pipe approach (see below), and also takes into account literature that does not explicitly focus on the looping problem, but addresses closely related problems. Moreover, the present paper also introduces two novel models for the looping problem, one of which has turned out to outperform the existing models in the literature with respect to computational time, the number of solutions found, the number of instances solved and cost savings. Finally, in discussing these models, the paper goes beyond the existing literature by including, throughout the paper, the practically highly relevant case that a particular pipeline may be looped several times.

In line with our aim to bring together the diversity of existing approaches in the literature, the remainder of this section discusses the literature on the topic rather comprehensively before presenting an outline of the structure of the paper.

During the past 40 years of research on capacity problems in potential-driven networks, three different modelling approaches have been considered for the looping problem (cf. [SS16]): (i) a direct approach where the optimal diameters for the loops are chosen from the set of commercially available diameters (discrete approach), (ii) a continuous approach where continuous diameters are

typically used to approximate the problem and (iii) an extended approach where the entire length of the pipeline to be looped is split into several segments of variable lengths each of which may have its own diameter from the discrete set of available diameters (split-pipe approach).

In the following discussion of the literature about these three approaches, we will not only look at papers addressing the looping problem (in fact, to the best of our knowlegdge, there are so far, apart from the present paper, only two papers, namely [ABC09] and [PAC08], that solely focus on the looping problem), but will also consider papers about two closely related problems: the *network design problem*, which looks for the optimal diameters of pipelines between given unconnected nodes or determines both the location of nodes and the optimal pipelines between these, and the *network expansion problem*, which is about the optimal placement of new network elements of different types (such as pipelines, compressors, valves, etc.) at pre-defined (previously connected or unconnected) locations in the network.

(i) Discrete diameters. One of the first solution approaches using mathematical optimization techniques for this NP-hard problem ([YTB84]) has been proposed by [Jac68]. The author solves a nonlinear model using a gradient approximation method where the resulting continuous diameters are finally rounded to the nearest discrete-valued pipe diameters. The approach was tested on a small water network that contains seven pipelines and two cycles. Other early work where the discrete approach is used for networks with a simple structure include [Lia71], who solved a gunbarrel system using dynamic programming, and [RFR+70], who used a serial and parallel merge algorithm to design tree shaped networks. Later, [Ges85] applied selective enumeration techniques to tackle a small sized network with two cycles. In the 1990s a class of approaches was developed that relies on meta heuristics, such as genetic algorithms, see e.g. [SDM94] and [SW97] for water networks, and [BSR94] and [CG98] for gas networks.

In the past decade, different papers applied a MINLP formulation to determine discrete pipe sizes, e.g. [ABC09], [BDL+12] and [RSS+19]. While [ABC09] solve the problem heuristically in two stages, where a first step identifies pipes to be looped by solving the continuous relaxation and a second step determines discrete-valued diameters for these selected pipes using Branch and Bound, [BDL+12] solve the model directly with an MINLP solver using a continuous reformulation of the cost function. [RSS+19] determine discrete arc sizes in the context of tree shaped networks, which allows flows on arcs to be fixed and thus simplifying the MINLP model to an Mixed Integer Programming model.

The nonlinear nature of the problem has led to a number of different MINLP models and approaches: [Rag13] presents a disjunctive program together with a convex relaxation, which is then solved to global optimality and [BSBB+16] propose a new solution approach by presenting a new model together with a second-order cone relaxation. [Hum14] formulates a model that is solved to global optimality using convex reformulation techniques, special tailored cuts.

In this paper, we investigate the different existing MINLP approaches and pro-

pose a new model for the discrete looping problem.

(ii) Continuous diameters have been considered e.g. by [BS79a], [RB82], [Bha85], [DWS96], [DWB12], [BNV12] and [AAB⁺13]. Continuous diameters typically have been used in the literature to approximate the discrete diameters that are commercially available. [HMN91], for example, use successive linear programming with a trust region strategy, where the algorithm adjusts the continuous diameters in each iteration to elements in the set of available discrete diameters. [OG95] apply sequential quadratic programming to the continuous relaxation of a gas network design problem and round the solution to the closest available diameter size. [SS16] introduce an analytic approach that calculates the optimal diameter costs for the pipe-sizing problem of a tree-shaped gas network with continuous diameters, which are then heuristically converted to discrete pipe diameters. As in the present paper is concerned with models that lead to exact solutions we will not further consider this line of research here.

(iii) Split-pipe approach. This approach, where the pipes can be split into several segments with different diameters each, combines features of both the discrete and the continuous approaches: while the diameters are chosen from the discrete set of available diameters, the option to split pipes at arbitrary points into sections of different diameters leads, as we will see in the next section, to a situation that is equivalent to choosing diameters from a continuous set. This concept was first used in designing networks with a tree structure, which allows the flows in the pipelines to be treated as constants and leads to a linear programming model (e.g. [KGM68] and [GHC72] for water networks, and, independently, [BS79b] for gas networks). The first paper to attempt the split-pipe version of the capacity expansion problem as a non-linear problem for general networks was [AS77], who introduced the linear programming gradient (LPG) method, a two-stage heuristic that alternates between solving the linear program with fixed flows from [KGM68] for obtaining the pipeline diameters and modifying these flows on the basis of a sensitivity analysis of the solution of the first stage. This idea stimulated a number of subsequent papers (e.g. [QLB81], [FK90] and [KS89]) that improved on the method. Starting with [ESBT94], a strand of genuinely NLP-based global solution methods emerged, with further contributions by [ZZ96] and [SSL01], for example. Surprisingly, despite 50 years of research on the split-pipe approach, all these contributions proceed from the very same basic model that goes back to the linear model by [KGM68]. For this reason, the present paper presents a novel, alternative model for the split-pipe approach and compares it to the split-pipe model in the literature.

We have now sketched the history of research on the three different approaches to our looping problem and closely related problems. In the light of the variety of models and optimization methods for these problems, it is astonishing that there is nearly no research that brings together the discrete and the split-pipe approaches and compares the models proposed in the literature. In fact, to the best of our knowledge, the very early paper by [BS79b] with its linear model for a tree-shaped network is the only paper to compare the split-pipe approach with the discrete approach, albeit for very small trees with 9 and 14 nodes. The present paper addresses this gap and presents comparisons of all models discussed in the literature.

Another important aspect of the looping problem that has not been addressed in the literature is multiple looping, where each pipeline may be reinforced with several parallel pipelines of different diameters. This is a practically highly relevant problem as multiple looping can first replace large diameters that are commercially not available; may second lead to cost savings by substituting several parallel pipelines with smaller diameters for one pipeline with a large diameter; may third, as we will see in Section 2.3, allow for pipe characteristics that cannot be realized with single loops; and can finally provide a tool for strategic planning where several stages of successively looping a given network are to be considered. For these reasons we will take into account multiple looping throughout the paper.

The remainder of the paper is organized as follows: In the next section we formalize the looping problem in potential-driven networks, show some of its basic properties and explain our approach of dealing with multiple loops. In Section 3, we present a new model for the discrete looping problem and contrast it with the existing models in the literature. The subsequent Section 4 turns to the split-pipe approach of the looping problem. Again we present a new model and address its relationship with the split-pipe model that can be found in the literature. Moreover, we discuss the way in which the feasible regions of all models presented so far are related to each other. In Section 5 we carry out extensive computational experiments on instances of both natural gas and water networks that allows us to compare the performance of all models and give recommendations regarding their use. The paper ends with some concluding remarks in Section 6.

2 The expansion planning problem

Let us begin by formally defining the planning problem that this paper discusses.

2.1 Problem statement

Let $\mathcal{G} = (\mathcal{V}, \mathcal{A}, r)$ be a directed multigraph with node set $\mathcal{V}$, arc set $\mathcal{A}$ and a function $r \colon \mathcal{A} \to \mathcal{V} \times \mathcal{V}$ that maps each arc to its end points. The nodes can be partioned into supply nodes (sources), consumption nodes (sinks) and and transshipment nodes. In this paper, we restrict to passive and connected networks, where the only arc type are pipes that are needed to transport the commodities. We are given a demand vector $b \in \mathbb{R}^{|\mathcal{V}|}$ where $b_v > 0$ denotes injection into the network at source $v \in \mathcal{V}$ and $b_v < 0$ withdrawal at sink $v \in \mathcal{V}$. Since we work with stationary and isothermal models, the demand vector is balanced, i.e. $\sum_{v \in \mathcal{V}} b_v = 0$. With each arc $a \in \mathcal{A}$, flow variables $x_a \in [\underline{x}_a, \overline{x}_a]$ are associated. Positive values of x_a indicate flow along the arc $a = (v, w)$ from v to w, whereas negative values indicate flow in the reversed direction. As in classical network flow problems, flow conservation is required at every node i.e. $\sum_{a \in \delta^+(v)} x_a - \sum_{a \in \delta^-(v)} x_a = b_v$ with $\delta^+(v) := \{(v, w) \in A\}$ and $\delta^-(v) := \{(w, v) \in A\}$. In potential based networks, the physical state is

additionally described by nonnegative potential variables $\pi_v \in [\underline{\pi}_v, \overline{\pi}_v]$ at each node $v \in \mathcal{V}$.

In applications such as water transport problems the nodal potentials π_v correspond to hydraulic heads, i.e. the sum of the elevation head, velocity head and pressure head ([WCS01]), while they respresent squared pressure variables in gas transport problems ([KHPSe15]).

The flow along a pipe $a = (v, w)$ depends on the potential difference at its adjacent nodes, the pipe length $L_a > 0$, a physical parameter $R_a > 0$ respresenting phenomena such as friction and density, and the diameter $d_a > 0$. The potential difference

$$\pi_v - \pi_w = \Phi(d_a, x_a) \qquad \forall a \in \mathcal{A} \text{ and } (v, w) = r(a), \tag{1}$$

is given by a function $\Phi(d_a, .) : \mathbb{R} \to \mathbb{R}$ that is strictly increasing, $\Phi(d_a, .) \in \mathcal{C}^1$ and antisymmetrical, i.e. $\Phi(d_a, -x_a) = -\Phi(d_a, x_a)$, and typically takes the form

$$\Phi(d_a, x_a) = \frac{L_a R_a}{d_a^\beta} \, \text{sgn}(x_a) |x_a|^\alpha. \tag{2}$$

with $\alpha, \beta > 0$. In line with most authors in the literature, we model R_a as a constant and assume the pipes to have zero slope, i.e. there is no influence of gravity on the potential drop $\pi_v - \pi_w$ (e.g. [ZZ96], [ABC09], [BNV12]). The value of α depends on the commodity and the type of approximation used for modelling the commodity flow. In water transport problems, the potential loss function is typically given by the equations of Darcy-Weisbach with $\alpha = 2$ or Hazen-Williams with $\alpha = 1.852$ ([WCS01]), whereas in gas transport problems Eq. (2) takes the shape of the Weymouth equation with $\alpha = 2$ ([Wey12]). In the approximations proposed by Darcy-Weisbach and Weymouth, the exponent of the diameter is $\beta = 5$, while it is $\beta = 4.87$ in the case of the approximation by Hazen-Williams.

The solution of the capacity expansion problem involves two decisions: (a) which pipelines $a \in A$ should be looped? and (b) what pipeline diameters d_a should be used for the loops? As we wish to solve the problem to global optimality and any preselection of certain pipes are looping candidates would be a heuristic procedure, we will allow all pipes to be looped.

For the diameters d_a we have in the discrete case $d_a \in \mathcal{D}_a := \{D_{a,0}, D_{a,1}, ..., D_{a,k_a}\}$, where $d_{a,0}$ refers to the diameter of the already existing pipe and d_{a,k_a} denotes the maximal possible diameter when looping. In the continuous case the domain of the diameter is given by $d_a \in \mathcal{D}_a := [D_{a,0}, D_{a,k_a}]$. While the present paper is not concerned with the continuous looping problem due to its approximative nature, we will see in Section 2.3 that a continuous interval of diameters can also be interpreted as representing diameters in the split-pipe problem.

The general looping problem then reads

$$\min_{x_a, d_a, \pi_v} \sum_a c(d_a) L_a \tag{3a}$$

$$\text{s.t.} \qquad \pi_v - \pi_w = \Phi(d_a, x_a) \qquad\qquad \forall a \in \mathcal{A} \text{ and } (v, w) = r(a) \tag{3b}$$

6

$$\sum_{a\in\delta^+(v)} x_a - \sum_{a\in\delta^-(v)} x_a = b_v \qquad\qquad \forall\, v \in V \qquad (3c)$$

$$\underline{\pi}_v \le \pi_v \le \overline{\pi}_v \qquad\qquad \forall\, v \in V \qquad (3d)$$

$$\underline{x}_a \le x_a \le \overline{x}_a \qquad\qquad \forall\, a \in A \qquad (3e)$$

$$d_a \in \mathcal{D}_a \qquad\qquad \forall\, a \in A \qquad (3f)$$

Note that even when no particular bounds $\underline{x}_a$ and $\overline{x}_a$ on the flow variables x_a are given, flow bounds are implied by the bounds on π_v and d_a by virtue of Equation (1).

To provide the reader with some intuition for the problem statement of the expansion planning problem, let us briefly demonstrate that we can always find a solution to the problem provided that there is no flow bound (3e), that the intersection of the bounds of the potential variables is non-empty, i.e. $\cap_{v\in V}\,[\underline{\pi}_v, \overline{\pi}_v] = [\underline{\pi}, \overline{\pi}]$ with $\underline{\pi} \le \overline{\pi}$ and that we can choose sufficiently large pipeline diameters.

Let $\overline{b}$ the overall network inflow, i.e. the sum of all flows that enter the network at the entry nodes, which is clearly an upper bound on any flow along any arc in the network. We now select for each arc $a \in A$ a diameter $\tilde{d}_a$ such that $L_a R_a / \tilde{d}_a^{\beta}\,\mathsf{sgn}(\overline{b})|\overline{b}|^{\alpha} \le (\overline{\pi} - \underline{\pi})/|A|$.

Let $(\pi_v, x_a)_{v\in V, a\in A}$ be a corresponding solution of problem (3a) – (3c) where the values for d_a are fixed. The solution has unique flow values and the potential values are uniquely determined up to a constant shift (cf. [CCH$^+$78], [Hum14] for the existence of such a solution), which obviously satisfies

$$\pi_v - \pi_w = \frac{L_a R_a}{\tilde{d}_a^{\beta}}\,\mathsf{sgn}(x_a)|x_a|^{\alpha} \le \frac{L_a R_a}{\tilde{d}_a^{\beta}}\,\overline{b}^{\alpha} \le \frac{\overline{\pi} - \underline{\pi}}{|A|} \quad \forall a \in \mathcal{A} \text{ and } (v,w) = r(a).$$

Summing up these inequalities along any path $\overrightarrow{v_0 v_k}$ between two connected nodes v_0 and v_k yields:

$$|\pi_{v_0} - \pi_k| = |\sum_{i=0}^{k-1} \pi_{v_i} - \pi_{v_{i+1}}| = |\sum_{i=0}^{k-1} \frac{L_{i,i+1} R_{i,i+1}}{\tilde{d}_{i,i+1}^{\beta}}\,\mathsf{sgn}(x_{i,i+1})|x_{i,i+1}|^{\alpha}|$$

$$\le \sum_{i=0}^{k-1} \frac{L_{i,i+1} R_{i,i+1}}{\tilde{d}_{i,i+1}^{\beta}}\,\overline{b}^{\alpha} \le \sum_{i=0}^{k-1} \frac{\overline{\pi} - \underline{\pi}}{|A|} \overset{k\le|A|}{\le} \overline{\pi} - \underline{\pi}.$$

We now choose a node v_0 with the highest potential in our solution and shift this node's potential by a constant $r \in \mathbb{R}$ such that it is equal to the upper bound $\overline{\pi}$, i.e. such that we have $\tilde{\pi}_{v_0} := \pi_{v_0} + r = \overline{\pi}$ for the new potential of the node v_0. If we shift the potentials π_v of all other nodes by the same constant r, all our potentials are guaranteed to satisfy (3d).

2.2 Convexity analysis

In the previous section we have seen that we can model the expansion planning problem as a Mixed-Integer Nonlinear Program (MINLP) for discrete looping decisions and we will see in the following Section 2.3 that the split-pipe approach

leads to a Nonlinear Program (NLP). We will now show that the feasible regions of the discrete and continuous (or split-pipe) capacity expansion planning problem are non-convex.

When neglecting loop expansions, i.e. $d_a = d_{a,0}$ for all pipes $a \in \mathcal{A}$, the continuous and the discrete problem (3) reduce to the same existence problem of validating a given demand scenario for feasibility. The feasible region of the resulting problem is convex ([Mau77] and [CCH$^+$78]), even though it comprises nonlinear nonconvex constraints of type (3b). However, the following proposition shows that this property does not hold for the feasible region of the expansion planning problem.

Proposition 1. *The feasible regions of the continuous and discrete expansion planning problems are nonconvex.*

Proof. The discrete expansion planning problem is trivially nonconvex. Regarding the continuous (and split-pipe) problem, we consider a network of two pipes a_1, a_2 in parallel with adjacent nodes v and w. For given $\alpha = 2$ and $\beta = 5$, let a demand situation $b_v = 10 = -b_w$, pipe properties $L_a, R_a = 1$ and diameters $d_a \in [1, 2]$ for both pipes $a \in \{a_1, a_2\}$ be given. Then the flow distribution among the parallel pipes is unique, where both pipes have the same flow direction, i.e. $\mathsf{sgn}(x_{a_1}) = \mathsf{sgn}(x_{a_2})$, and thus (2) reduces to $\pi_v - \pi_w = L_a R_a / d_{a_1}^5 x_{a_1}^2 = L_a R_a / d_{a_2}^5 x_{a_2}^2$. Then $s := (x_{a_1}, x_{a_2}, d_{a_1}, d_{a_2}, \pi_v, \pi_w) \approx (1.50, 8.50, 1, 2, 100, 97.74)$ is a solution of problem (3) and by symmetry another solution exits when swapping the diameters of both pipes, i.e. $\tilde{s} := (\tilde{x}_{a_1}, \tilde{x}_{a_2}, \tilde{d}_{a_1}, \tilde{d}_{a_2}, \tilde{\pi}_v, \tilde{\pi}_w) \approx (8.50, 1.50, 2, 1, 100, 97.74)$, where the numbers are rounded to two decimals. But there exists an infeasible convex combination of both solutions, namely $s^* := 0.5(s + \tilde{s}) = (x_{a_1}^*, x_{a_2}^*, d_{a_1}^*, d_{a_2}^*, \pi_v^*, \pi_w^*) \approx (5, 5, 1.5, 1.5, 100, 97.74)$, because s^* violates Eq. (3b), i.e.

$$\pi_v^* - \pi_w^* \neq L_a R_a / (d_{a_1}^*)^5 (x_{a_1}^*)^2 = L_a R_a / (d_{a_2}^*)^5 (x_{a_2}^*)^2$$

since $\pi_v^* - \pi_w^* \approx 2.26$ and $L_a R_a / (d_{a_1}^*)^5 (x_{a_1}^*)^2 = 1/1.5^5 * 5^2 \approx 3.29$.

$\square$

2.3 Loop diameters

After our general outline of the extension planning problem and some of its basic characteristics, we will now describe how the diameters used for looping pipelines are represented in the discrete models and the split-pipe to be discussed in the present paper. In doing so, we will go beyond the existing literature and allow that each pipeline a may be looped several times. Our starting point is a finite set $\{d_1, \dots, d_n\}$ of commercially available diameters for looping. These diameters are associated with costs $c_1 < \dots < c_n$ per unit of pipeline length, see left Figure 1.

It is well known in the literature (see [Kat59] and [ABC09] for the case of gas networks and [BDL$^+$12] for the case of water networks, for example) that two parallel pipelines with diameters d_1 and d_2 can be replaced, without changing
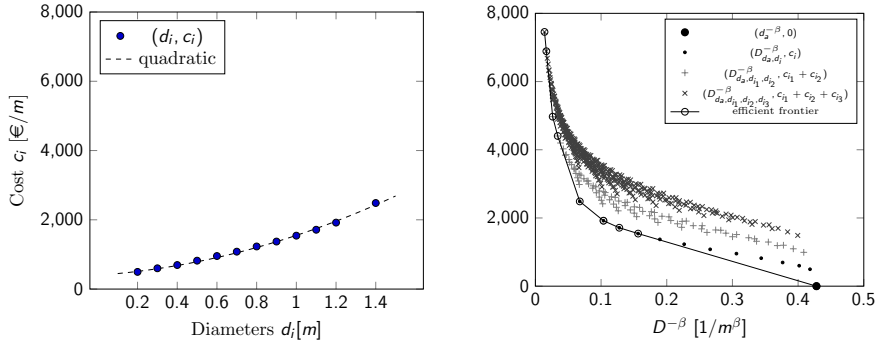
Figure 1: Available diameters $\{d_1, \ldots, d_n\}$ with cost factors (left). On the right: construction of the efficient frontier of dominating equivalent diameters. Note that the cost components of the points (d_i, c_i) on the left side correspond with those of the points $(D_{d_a,d_i}^{-\beta}, c_i)$ on the right side.

any physical properties of the network, by a single pipeline with diameter D_{d_1,d_2} (called *equivalent diameter*) by virtue of

$$D_{d_1,d_2}^{\beta/\alpha} = d_1^{\beta/\alpha} + d_2^{\beta/\alpha}.$$

It is easy to see that for looping an arc a of length L with existing diameter d_a multiple (k) times with (not necessarily distinct) diameters $d_{i_1}, d_{i_2}, \ldots, d_{i_k} \in D$, this relationship can be extended to

$$D_{d_a,d_{i_1},\ldots,d_{i_k}}^{\beta/\alpha} = d_a^{\beta/\alpha} + d_{i_1}^{\beta/\alpha} + \ldots + d_{i_k}^{\beta/\alpha}, \tag{4}$$

where this multiple loop is associated with costs

$$\sum_{j=1}^{k} c_{i_j} L. \tag{5}$$

This equation implies that when allowing for multiple loops, we can not only choose among the discrete set of commercially available diameters, but we have at our disposal the much larger discrete set of equivalent diameters that result from all possible combinations of available loops. This is the approach that we choose in our discrete models.

A type of equivalent diameter also exists for the case of two *serial* pipelines with no source or sink in between. Assume we have two pipelines, one from node v to node w and the other one from w to z, with diameters d_1 and d_2, the same physical parameter $R := R_1 = R_2$, a total length of L from v to z, and with λL being the length of the first pipeline from v to w, where $0 < \lambda < 1$. Then, due to (1) and (2), the total potential loss along the flow x from v to z is given by

$$(\pi_v - \pi_w) + (\pi_w - \pi_z) = \frac{\lambda L R_1}{d_1^\beta} \operatorname{sgn}(x)|x|^\alpha + \frac{(1-\lambda)L R_2}{d_2^\beta} \operatorname{sgn}(x)|x|^\alpha.$$

The same potential loss can be achieved with a single pipeline of length L when its diameter D is given by $\pi_v - \pi_z = LR/D^\beta \operatorname{sgn}(x)|x|^\alpha$. This and the previous equation imply that the equivalent diameter satisfies $D^{-\beta} = \lambda d_1^{-\beta} + (1-\lambda)d_2^{-\beta}$. Analogously we have for the case of k serial pipeline segments with diameters

9

$d_1, d_2, \ldots, d_k$ and lengths $\lambda_1 L, \lambda_2 L, \ldots, \lambda_k L$, where $\sum_{i=1}^{k} \lambda_i = 1$ with $\lambda_i \geq 0$:

$$D^{-\beta} = \sum_{j=1}^{k} \lambda_i d_i^{-\beta} \qquad \text{with associated costs} \qquad L \sum_{j=1}^{k} \lambda_i c_i. \qquad (6)$$

For practical purposes, relationship (6) implies that, when looping pipelines, we are not restricted to the discrete set of equivalent diameters that results from multiple loops with the commercially available diameters according to (4). Instead we can decide to split a pipeline into several segments of lengths $\lambda_i L$ and have different, possibly multiple loops, i.e. loops with different equivalent diameters, on all segments. In this way, (6) allows us to realize in the network all diameters in the (continuous) interval between the smallest and the largest possible equivalent diameters. This is the approach that our *split-pipe models* use and it enables us to benefit, despite the limited number of commercially available pipeline diameters, from having the flexibility of a continuous set of equivalent diameters at our disposal.

We will allow a single pipe a to be looped up to r times, i.e. in the capacity expansion problem we have to pick up to r diameters for each existing pipeline of the network. With $\binom{n+k-1}{k}$ being the number of k-combinations with repetitions from the diameter set $\mathcal{D}$, we obtain $\sum_{k=0}^{r} \binom{n+k-1}{k}$ equivalent diameters for each pipe. To avoid a factorial growth of the number of variables in our models, we apply a model reduction technique to our split-pipe models that has been introduced in the literature for the case of single looping several times (e.g. [BS79b], [FD87], [ZZ96]). We will briefly outline this method for the general case of multiple loops.

Our aim is to compare all potential equivalent diameters resulting from both parallel pipelines (multiple loops) and serial pipelines (split-pipe setting) with respect to their costs and to incorporate into our models only those equivalent diameters that are not dominated by equivalent diameters with lower costs. To this end, we place all pairs of equivalent diameters D generated according to (4) and their corresponding costs per unit of pipeline length c_D according to (5) into a coordinate system where the horizontal axis depicts the *exponentiated* equivalent diameter $D^{-\beta}$ and the vertical axis the cost c_D, i.e. we consider the points $(D^{-\beta}, c_D)$. In this coordinate system, the $D^{-\beta}$ coordinate of each convex combination of points $(D_i^{-\beta}, c_{D_i})$ represents, due to (6), an equivalent diameter that results from splitting a looped pipe into several sections with equivalent diameters D_i each, while the c_D coordinate of such a convex combination indicates, also due to (6), the corresponding cost of such a split-pipe arrangement. This is illustrated on the right side of Fig. 1. It shows for an arbitrary pipeline the original diameter $d_a^{-\beta}$ of the pipeline, which has zero cost (solid circle); all equivalent diameters $D_{d_a,d_i}^{-\beta}$ that result from looping the existing pipeline once at cost c_i (dots in our diagram); all equivalent diameters $D_{d_a,d_{i_1},d_{i_2}}^{-\beta}$ that result from looping the existing pipeline twice at cost $c_{i_1} + c_{i_2}$ (plus signs in the diagram); and finally, all equivalent diameters $D_{d_a,d_{i_1},d_{i_2},d_{i_3}}^{-\beta}$ that result from triple looping at cost $c_{i_1} + c_{i_2} + c_{i_3}$, which are represented by crosses in the diagram.

This setting allows us to identify those equivalent diameters among the equivalent diameters generated by (4) that are crucial for our models, namely those

equivalent diameters that are represented by the extreme points of the "efficient frontier" on the right side of Fig. 1 (circled in the diagram). All points on the line that represent the efficient frontier in our figure correspond to convex combinations of two equivalent diameters, i.e. a split-pipe setting where the pipeline is split into two segments, and these points represent the cost minimal ways of equipping a particular arc of the network with a certain equivalent diameter. As a consequence, for finding an optimal solution to the capacity extension problem, it suffices to incorporate into our split-pipe models only the equivalent diameters that correspond to the extreme points of the efficient frontier and to discard all others. As we will see when introducing the models this will greatly reduce the number of variables.

In the case of the discrete capacity extension problem, we cannot use convex combinations of equivalent diameters and have to get along with the equivalent diameters given by (4). Unfortunately however, we cannot possibly use all equivalent diameters depicted on the right of Fig. 1 for all arcs of the network as this would make the problem computationally intractable. For this reason, we will restrict our models to the equivalent diameters that result from looping an existing pipeline once and, additionally, to those equivalent diameters that result from looping an existing pipeline several times *and* are extreme points of the efficient frontier described above. In this way we may have to do without some potentially cost-minimal diameters, but will keep our models at a reasonable size and will, due to having incorporated the option of multiple looping, still achieve better results than the approaches presented in the literature.

In the following we denote the set of equivalent extension diameters of pipe *a* that we use in our models by $\{D_{a,0}, ..., D_{a,k_a}\}$ (sorted in ascending order), where $D_{a,0}$ represents the (unlooped) original diameter d_a of the existing pipe, and the corresponding cost factors by $c_{a,0} < ... < c_{a,k_a}$. Moreover, for convenience' sake we write $[k_a] = \{0, 1, ..., k_a\}$.

3 Discrete Loop Expansions

In this section, we present different approaches to model discrete loop expansions. We will begin with the model that is closest to our generic formulation of the capacity expansion problem (3).

3.1 Discrete looping with potential function constraints (Model A)

To formulate a MINLP on the basis of our generic formulation (3) of the looping problem, which can be found, in its discrete version, in several recent papers (e.g. [ABC09, BDL$^+$12, RSS$^+$19]), we have to specify the way in which the discrete values of the diameter variables are selected. Here this happens in a straight forward way by means of binary variables $\lambda_{a,i}$ each of which represents one diameter $D_{a,i} \in \{D_{a,0}, ..., D_{a,k_a}\}$. Constraint (7c) ensures that exactly one

diameter is chosen for each arc $a \in \mathcal{A}$, and in constraint (7b) the potential function has been rewritten in a way such that the chosen diameter is used for modelling the potential loss. With these variables $\lambda_{a,i}$ the objective function is a direct consequence of (5) in conjunction with the generic objective function (3a), while all other constraints are identical with the generic model.

$$\underset{\lambda,x,\pi}{\text{minimize}} \quad \sum_{a\in\mathcal{A}} L_a \sum_{i=0}^{k_a} \lambda_{a,i} c_{a,i} \tag{7a}$$

$$\text{s.t.} \quad \pi_v - \pi_w = L_a R_a \left(\sum_{i=0}^{k_a} \frac{\lambda_{a,i}}{D_{a,i}^{\beta}} \right) \text{sgn}(x_a)|x_a|^{\alpha} \quad \forall a \in \mathcal{A},\, (v,w) = r(a) \tag{7b}$$

$$\sum_{i=0}^{k_a} \lambda_{a,i} = 1 \qquad\qquad\qquad\qquad \forall a \in \mathcal{A} \tag{7c}$$

$$\sum_{a\in\delta^+(v)} x_a - \sum_{a\in\delta^-(v)} x_a = b_v \qquad\qquad \forall v \in \mathcal{V} \tag{7d}$$

$$\underline{\pi}_v \leq \pi_v \leq \overline{\pi}_v \qquad\qquad\qquad\qquad \forall v \in \mathcal{V} \tag{7e}$$

$$\underline{x}_a \leq x_a \leq \overline{x}_a \qquad\qquad\qquad\qquad \forall a \in \mathcal{A} \tag{7f}$$

$$\lambda_{a,i} \in \{0,1\} \qquad\qquad\qquad\qquad \forall a \in \mathcal{A}\; \forall i \in [k_a] \tag{7g}$$

3.2 Discrete looping with flow direction variables (Model B)

This model is based on an approach by [Rag13] to tackle the network design problem. The main difference to the previous model is that we split the flow variable x_a into two sets of variables $x_{a,i}^+$ and $x_{a,i}^-$, which represent the flow on arc a with diameter $D_{a,i}$ in forward and backward direction, respectively (see constraint (8e)), and correspondingly, variables Δ_a^+, Δ_a^- that describe the potential drop along the arcs according to the flow direction (see constraints (8b) and (8c)). The idea behind this formulation is that constraint (7b) uses two non-linear functions, namely the power function and the sign function, and by splitting the variable x_a into a part for a forward flow and a part for a backward flow, we are able to do without the latter function. This comes at a price, however, as we have to introduce additional binary variables z_a to choose between the two flow directions (8g) and (8h).

In his paper [Rag13] develops his own solution algorithm to the network design problem, where he uses a convex relaxation of the problem by relaxing the potential constraint function (1) to $\Delta_a^+ \leq \Phi(D_{a,i}, x_{a,i}^+)$ and $\Delta_a^- \leq \Phi(D_{a,i}, x_{a,i}^-)$ $\forall a \in \mathcal{A}, \forall i \in [k_a]$. In our context here, to model the expansion problem in an exact way, our aim is to enforce these equations with equality. In general, this can be done in different ways, such as using big-M constraints for modelling the disjunctions $\vee_{i=0}^{k_a} \Delta_a^+ = \Phi(D_{a,i}, x_{a,i}^+)$ and $\vee_{i=0}^{k_a} \Delta_a^- = \Phi(D_{a,i}, x_{a,i}^-)$ or introducing the following constraints: $\Delta_a^+ = \sum_{i=0}^{k_a} \Phi(D_{a,i}, x_{a,i}^+)$ and $\Delta_a^- = \sum_{i=0}^{k_a} \Phi(D_{a,i}, x_{a,i}^-)$ for all $a \in \mathcal{A}$. Preliminary computational tests by the authors of the present pa-

per revealed that the best performance is achieved with Balas's convex hull formulation [Bal85], i.e. when resolving these disjunctions by explicitly modelling the potential drop along arc a for diameter $D_{a,i}$ with the variables $\Delta_{a,i}^+, \Delta_{a,i}^-$. The model then reads as follows:

$$\underset{\lambda,x,\pi,\Delta,z}{\text{minimize}} \quad \sum_{a\in\mathcal{A}} L_a \sum_{i=0}^{k_a} \lambda_{a,i} c_{a,i} \tag{8a}$$

$$\text{s.t.} \qquad \pi_v - \pi_w = \sum_{i\in[k_a]} \left(\Delta_{a,i}^+ - \Delta_{a,i}^-\right) \qquad \forall a\in\mathcal{A}, (v,w)=r(a) \tag{8b}$$

$$\Delta_{a,i}^+ = \frac{L_a R_a}{D_{a,i}^\beta}(x_{a,i}^+)^\alpha \qquad \Delta_{a,i}^- = \frac{L_a R_a}{D_{a,i}^\beta}(x_{a,i}^-)^\alpha \quad \forall a\in\mathcal{A} \, \forall i\in[k_a] \tag{8c}$$

$$\sum_{i=0}^{k_a} \lambda_{a,i} = 1 \qquad \forall a\in\mathcal{A} \, \forall i\in[k_a] \tag{8d}$$

$$\sum_{a\in\delta^+(v)} \sum_{i\in[k_a]} (x_{a,i}^+ - x_{a,i}^-) - \sum_{a\in\delta^-(v)} \sum_{i\in[k_a]} (x_{a,i}^+ - x_{a,i}^-) = b_v \qquad \forall v\in\mathcal{V} \tag{8e}$$

$$0 \le x_{a,i}^+ \le \overline{x}_{a,i} \lambda_{a,i} \qquad 0 \le x_{a,i}^- \le \overline{x}_{a,i}\lambda_{a,i} \qquad \forall a\in\mathcal{A} \, \forall i\in[k_a] \tag{8f}$$

$$x_{a,i}^+ \le \overline{x}_{a,i} z_a \qquad x_{a,i}^- \le \overline{x}_{a,i}(1-z_a) \qquad \forall a\in\mathcal{A} \, \forall i\in[k_a] \tag{8g}$$

$$0 \le \Delta_{a,i}^+ \le \overline{\Delta}_a^+ z_a \qquad 0 \le \Delta_{a,i}^- \le \overline{\Delta}_a^-(1-z_a) \quad \forall a\in\mathcal{A} \, \forall i\in[k_a] \tag{8h}$$

$$\underline{\pi}_v \le \pi_v \le \overline{\pi}_v \qquad \forall v\in\mathcal{V} \tag{8i}$$

$$\lambda_{a,i} \in \{0,1\} \qquad \forall a\in\mathcal{A} \, \forall i\in[k_a] \tag{8j}$$

with $\overline{\Delta}_a^+ := \overline{\pi}_v - \underline{\pi}_w$ and $\overline{\Delta}_a^- := \overline{\pi}_w - \underline{\pi}_v$. Equations (8f) - (8h) model that non-negative flow of arc a takes only place along a given direction for the selected diameter $D_{a,i}$ inducing the corresponding potential drop. To provide another upper bound on the potential loss on the arcs, [Rag13] introduces further constraints, which in our case read $\Delta_{a,i}^- \le L_a R_a/D_{a,i}^\beta (\overline{x}_{a,i})^{\alpha-1} x_{a,i}^-$ and $\Delta_{a,i}^+ \le L_a R_a/D_{a,i}^\beta (\overline{x}_{a,i})^{\alpha-1} x_{a,i}^+$.

3.3 Discrete looping with potential difference variables (Model C)

The model presented in this section is a novel approach to tackle the non-linearities of the problem. Instead of introducing variables for the flow direction as in the previous model, the idea here is to reduce the two-dimensional function $\Phi(d_a, x_a)$ to a one-dimensional function $\phi(x_a)$. This is achieved by shifting the loop diameters from the term that represents the flow towards the potential loss term (see Eq. (9b)). To this end, we introduce variables $\Delta_{a,i}$ to model the potential loss that corresponds to the loop diameter chosen (constraint (9c)) and use the binary variables $\lambda_{a,i}$ not to choose the diameters, but to select the potential loss. As a trade-off we have to introduce big-M constraints (9d) to determine the potential loss variables $\Delta_{a,i}$.

$$\underset{\lambda,x,\pi,\Delta}{\text{minimize}} \quad \sum_{a\in\mathcal{A}} L_a \sum_{i=0}^{k_a} \lambda_{a,i} c_{a,i} \tag{9a}$$

$$\text{s.t.} \quad \operatorname{sgn}(x_a)|x_a|^\alpha - \sum_{i=0}^{k_a} \frac{D_{a,i}^\beta}{L_a R_a} \Delta_{a,i} = 0 \qquad \forall\, a \in \mathcal{A}, \quad (9\text{b})$$

$$\sum_{i=0}^{k_a} \Delta_{a,i} = \pi_v - \pi_w \qquad \forall a \in \mathcal{A},\ (v,w) = r(a), \quad (9\text{c})$$

$$M_{a,i}^{(1)} \lambda_{a,i} \leq \Delta_{a,i} \leq M_{a,i}^{(2)} \lambda_{a,i} \qquad \forall a \in \mathcal{A}\ \forall i \in [k_a], \quad (9\text{d})$$

$$\sum_{i=0}^{k_a} \lambda_{a,i} = 1 \qquad \forall\, a \in \mathcal{A}, \quad (9\text{e})$$

$$\sum_{a \in \delta^+(v)} x_a - \sum_{a \in \delta^-(v)} x_a = b_v \qquad \forall\, v \in \mathcal{V}, \quad (9\text{f})$$

$$\underline{\pi}_v \leq \pi_v \leq \overline{\pi}_v \qquad \forall\, v \in \mathcal{V}, \quad (9\text{g})$$

$$\underline{\pi}_v - \overline{\pi}_w \leq \Delta_{a,i} \leq \overline{\pi}_v - \underline{\pi}_w \qquad \forall a \in \mathcal{A}\ \forall i \in [k_a] \quad (9\text{h})$$

$$\underline{x}_a \leq x_a \leq \overline{x}_a \qquad \forall\, a \in \mathcal{A}, \quad (9\text{i})$$

$$\lambda_{a,i} \in \{0,1\} \qquad \forall a \in \mathcal{A}\ \forall i \in [k_a], \quad (9\text{j})$$

where $M_{a,i}^{(1)} := \Phi(D_{a,i}, \underline{x}_a)$, $M_{a,i}^{(2)} := \Phi(D_{a,i}, \overline{x}_a)$.

3.4 Further approaches for discrete looping in the literature

There are two further models in the literature for the network design problem, both of which are not intended as "stand-alone-models", but as starting points for solution algorithms. While we will not go into the details of the models here because preliminary computational experiments have demonstrated that they do not perform particularly well (see Section 5.2), the approaches still deserve mentioning.

[BSBB+16] present an MINLP formulation that is similar to the model developed by [Rag13] (Section 3.1) in the sense that it distinguishes between forward and backward flow directions. Here, however, this is achieved by binary variables z_a^+, z_a^- that are then used to model the potential loss function in product form $\lambda_{a,i}(z_a^+ - z_a^-)(\pi_v - \pi_w) = L_a R_a / D_{a,i}^\beta x^\alpha$.

[Hum14] uses indicator constraints to select arcs a in the network design problem (or diameters $D_{a,i}$ in the context of our expansion problem), i.e. constraints of the form $\lambda_{a,i} = 1 \Rightarrow \pi_v - \pi_w = \Phi(d_{a,i}, x_a)$ and $\lambda_{a,i} = 0 \Rightarrow x_{a,i} = 0$, which may be represented by big-M constraints in a MINLP model.

3.5 Equivalence of the discrete models

To finish our section on discrete models for the capacity expansion problem, we show that the models presented so far are equivalent. In this context we denote the feasible regions of the discrete models as: X_A for Model A, X_B for Model B and X_C for Model C. Note that when we use the subset relation "$\subseteq$" for these feasible regions in the following, this relation is not meant to be understood literally since the different model formulations do not share the same variables.

Instead, we understand the subset relation to mean that a suitable mapping of feasible points exists.

Proposition 2. *The discrete Models A, B and C are equivalent, i.e.* $X_A = X_C = X_B$.

Proof. "$X_A \subseteq X_C$":

Let $(\tilde{x}_a, \tilde{\pi}_v, \tilde{\lambda}_{a,i})_{a \in A, v \in V, i \in [k_a]} \in X_A$ be a solution of Model A. We set for all $a \in A$, $v \in V$ and $\forall i \in [k_a] : x_a := \tilde{x}_a$, $\pi_v := \tilde{\pi}_v$ and $\lambda_{a,i} := \tilde{\lambda}_{a,i}$. Then (9e)-(9g) and (9i), (9j) follow right away. For pipe a let $\check{i}_a \in [k_a]$ be such that $\tilde{\lambda}_{a,\check{i}_a} = 1$. We define

$$\Delta_{a,i} := \begin{cases} 0 & \forall i \in [k_a] \setminus \{\check{i}_a\} \\ \frac{L_a R_a}{D_{a,i}^\beta} \operatorname{sgn}(x_a)|x_a|^\alpha & i = \check{i}_a. \end{cases}$$

Then (9b) follows by construction, (9c) equals (7b), (9d) holds by virtue of $M_{a,i}^{(1)} = \Phi(D_{a,\check{i}_a}, \underline{x}_a)$, $M_{a,i}^{(2)} = \Phi(D_{a,\check{i}_a}, \overline{x}_a)$ and (9h) follows from (9c) and (9g).

"$X_C \subseteq X_A$":

Let $(x_a, \pi_v, \lambda_{a,i}, \Delta_{a,i})_{a \in A, v \in V, i \in [k_a]} \in X_C$ be a solution of Model C. We set for all $a \in A$, $v \in V$ and $\forall i \in [k_a] : \tilde{x}_a := x_a$, $\tilde{\pi}_v := \pi_v$ and $\tilde{\lambda}_{a,i} := \lambda_{a,i}$. Then Equation (7b) follows from (9b) – (9d) and (7c) - (7g) equals (9e) - (9g), (9j) - (9i).

"$X_A \subseteq X_B$":

Let $(\tilde{x}_a, \tilde{\pi}_v, \tilde{\lambda}_{a,i})_{a \in A, v \in V, i \in [k_a]} \in X_A$ be a solution of Model A. We set for all $a \in A$, $v \in V$ and $\forall i \in [k_a] : \pi_v := \tilde{\pi}_v$ and $\lambda_{a,i} := \tilde{\lambda}_{a,i}$. For pipe a let $\check{i}_a \in [k_a]$ be such that $\tilde{\lambda}_{a,\check{i}_a} = 1$. We set $x_{a,i}^+, x_{a,i}^-, \Delta_{a,i}^+, \Delta_{a,i}^- := 0 \,\forall i \in [k_a] \setminus \{\check{i}_a\}$.

Case 1: if $\tilde{x}_a \geq 0$, define $x_{a,\check{i}_a}^+ := \tilde{x}_a$, $x_{a,\check{i}_a}^- = 0$ and $\Delta_{a,\check{i}_a}^+ := \tilde{\pi}_v - \tilde{\pi}_w$.

Case 2: if $\tilde{x}_a < 0$, define $x_{a,\check{i}_a}^+ = 0$, $x_{a,\check{i}_a}^- := -\tilde{x}_a$ and $\Delta_{a,\check{i}_a}^- := \tilde{\pi}_w - \tilde{\pi}_v$. Then in both cases (8b) - (8j) hold.

"$X_B \subseteq X_A$":

Let $\left(x_{a,i}^+, x_{a,i}^-, z_a, \pi_v, \lambda_{a,i}, \Delta_{a,i}^+, \Delta_{a,i}^-\right)_{a \in A, v \in V, i \in [k_a]}$ be a solution of Model B. Set for all $a \in A$, $v \in V$, $i \in [k_a] : \tilde{\pi}_v := \pi_v$, $\tilde{\lambda}_{a,i} := \lambda_{a,i}$ and $\tilde{x}_a := \sum_{i \in k_a}(x_{a,i}^+ - x_{a,i}^-)$ then the equations of Model A are satisfied. $\square$

4 Split-Pipe Loop Expansions

In this section we present two equivalent modelling approaches for continuous loop expansions. The first one is a model common in the literature (e.g. [AS77], [ZZ96]), while the second one is a novel approach.

4.1 Split-pipe looping with length variables (Model D)

This split-pipe model is identical to the first discrete Model A, except that the indicator variables $\lambda_{a,i} \in \{0, 1\}$ have been relaxed to continuous length variables $\lambda_{a,i} \in [0, 1]$ here, turning the MINLP of Section 3.1 into a NLP. As explained in Section 2.3, these length variables denote the proportion of pipe a that is looped with diameter $D_{a,i}$, i.e. for each arc a, the pipeline consists of segments of those equivalent diameters $D_{a,i}$ for which $\lambda_{a,i} > 0$.

$$\underset{\lambda, x, \pi}{\text{minimize}} \quad \sum_{a \in \mathcal{A}} L_a \sum_{i=0}^{k_a} \lambda_{a,i} c_{a,i} \tag{10a}$$

$$\text{s.t.} \quad \pi_v - \pi_w = L_a R_a \left(\sum_{i=0}^{k_a} \frac{\lambda_{a,i}}{D_{a,i}^{\beta}} \right) \text{sgn}(x_a) |x_a|^{\alpha} \quad \forall a \in \mathcal{A}, (v, w) = r(a) \tag{10b}$$

$$\sum_{i=0}^{k_a} \lambda_{a,i} = 1 \qquad \forall a \in \mathcal{A} \tag{10c}$$

$$\sum_{a \in \delta^+(v)} x_a - \sum_{a \in \delta^-(v)} x_a = b_v \qquad \forall v \in \mathcal{V} \tag{10d}$$

$$\underline{\pi}_v \leq \pi_v \leq \overline{\pi}_v \qquad \forall v \in \mathcal{V} \tag{10e}$$

$$\underline{x}_a \leq x_a \leq \overline{x}_a \qquad \forall a \in \mathcal{A} \tag{10f}$$

$$\lambda_{a,i} \in [0, 1] \qquad \forall a \in \mathcal{A} \ \forall i \in [k_a] \tag{10g}$$

4.2 Split-pipe looping with efficient-frontier constraints (Model E)

In the previous section we modelled the efficient frontier of equivalent diameters using length variables $\lambda_{a,i}$ to express convex combinations of the equivalent diameters $D_{a,i}$ that are the extreme points of the frontier. In the present section, the efficient frontier is modelled explicitly by using linear constraints.

As the efficient frontier consists of points $(D^{-\beta}, c)$ (cf. Section 2.3), we introduce new continuous variables y_a to model the exponentiated diameter (see constraints (11g) and (11b)) and variables c_a that represent the costs per unit length of pipe a for the equivalent diameters (see (11h) and the objective function). On this basis the efficient frontier can be represented by linear constraints (11d), each of which models the frontier between a pair of adjacent extreme points (cf. the right side of Fig. 1). The parameters $s_{a,i}$ and $t_{a,i}$ can be calculated in advance as $s_{a,i} = (c_{a,i} - c_{a,i+1})/(D_{a,i}^{-\beta} - D_{a,i+1}^{-\beta})$ and $t_{a,i} = -s_{a,i} D_{a,i+1}^{-\beta} + c_{a,i+1}$.

$$\underset{y, c, x, \pi}{\text{minimize}} \quad \sum_{a \in \mathcal{A}} L_a c_a \tag{11a}$$

$$\text{s.t.} \quad \pi_v - \pi_w = L_a R_a y_a \ \text{sgn}(x_a) |x_a|^{\alpha} \qquad \forall a \in \mathcal{A}, (v, w) = r(a) \tag{11b}$$

$$\sum_{a \in \delta^+(v)} x_a - \sum_{a \in \delta^-(v)} x_a = b_v \qquad \forall v \in V \tag{11c}$$

$$c_a \geq s_{a,i} y_a + t_{a,i} \qquad \forall a \in A \ \forall i \in [k_a - 1] \tag{11d}$$

$$\underline{\pi}_v \leq \pi_v \leq \overline{\pi}_v \qquad\qquad \forall v \in V \qquad (11e)$$

$$\underline{x}_a \leq x_a \leq \overline{x}_a \qquad\qquad \forall a \in A \qquad (11f)$$

$$\underline{y}_a \leq y_a \leq \overline{y}_a \qquad\qquad \forall a \in A \qquad (11g)$$

$$c_a \geq 0 \qquad\qquad \forall a \in A \qquad (11h)$$

Note that the bounds in (11g) can be calculated as $\underline{y}_a = D_{a,k_a}^{-\beta}$ and $\overline{y}_a = D_{a,0}^{-\beta}$.

4.3 Equivalence of the split-pipe models

The feasible region of Model D is a subset of the feasible region of Model E due to the inequality constraints (11d). But since the objective is to minimize the cost for building loops, solutions of both models are forced to be on the efficient frontier. The proof of the following proposition formalizes this argument.

Proposition 3. *Each optimal solution to Model D is an optimal solution to Model E, and vice versa.*

Proof. (i) Let $\left(x_a^*, \pi_v^*, \lambda_{a,i}^*\right)_{a \in A, v \in V, i \in [k_a]}$ be an optimal solution of Model D. According to [FD87], an optimal solution has the property that each arc consists of at most two pipe segments, where the corresponding equivalent diameters correspond to adjacent extreme points of the efficient-frontier $\{D_{a,i}^{-\beta}, c_{a,i}\}_{i=0}^{k_a}$. Hence, for all $a \in \mathcal{A}$ there exists a unique $i_0 \in [k_a - 1]$ such that $0 \leq \lambda_{a,i_0}^*, \lambda_{a,i_0+1}^* \leq 1$ with $\lambda_{a,i_0}^* + \lambda_{a,i_0+1}^* = 1$ and $\lambda_{a,i}^* = 0 \; \forall i \in [k_a] \setminus \{i_0, i_0 + 1\}$. Define

$$\tilde{y}_a := \lambda_{a,i_0}^* D_{a,i_0}^{-\beta} + \lambda_{a,i_0+1}^* D_{a,i_0+1}^{-\beta} \text{ and } \tilde{c}_a := \lambda_{a,i_0}^* c_{a,i_0} + \lambda_{a,i_0+1}^* c_{a,i_0+1}.$$

Set $\tilde{x}_a := x_a^*, \tilde{\pi}_v := \pi_v^*$, then $(\tilde{x}_a, \tilde{\pi}_v, \tilde{y}_a, \tilde{c}_a)_{a \in A, v \in V, i \in [k_a]}$ obviously fulfills the variable bounds (11e) - (11g) and Equation (11c). From (10b) and the definition of $\tilde{y}$ follows (11b). We now show that (11d) holds: From our definition of $\tilde{y}_a$ we obtain

$$\tilde{y}_a = \lambda_{a,i_0}(D_{a,i_0}^{-\beta} - D_{a,i_0+1}^{-\beta}) + D_{a,i_0+1}^{-\beta} \qquad (12)$$

$$\Leftrightarrow \lambda_{a,i_0}(c_{a,i_0} - c_{a,i_0+1}) = \frac{c_{a,i_0} - c_{a,i_0+1}}{D_{a,i_0}^{-\beta} - D_{a,i_0+1}^{-\beta}}(\tilde{y}_a - D_{a,i_0+1}^{-\beta}) \qquad (13)$$

$$\Leftrightarrow \lambda_{a,i_0}c_{a,i_0} + \lambda_{a,i_0+1}c_{a,i_0+1} = \frac{c_{a,i_0} - c_{a,i_0+1}}{D_{a,i_0}^{-\beta} - D_{a,i_0+1}^{-\beta}}(\tilde{y}_a - D_{a,i_0+1}^{-\beta}) + c_{a,i_0+1} \qquad (14)$$

$$\Leftrightarrow \tilde{c}_a = s_{a,i_0}\tilde{y}_a + t_{a,i_0}. \qquad (15)$$

Since the objective is to minimize and the slopes $s_{a,i}$ in (11d) are decreasing for increasing i, we have $\tilde{c}_a = s_{a,i_0}\tilde{y}_a + t_{a,i_0} = \max_{i \in [k_a-1]} s_{a,i}\tilde{y}_a + t_{a,i}$ which implies $\tilde{c}_a \geq s_{a,i}\tilde{y}_a + t_{a,i}$ for all $a \in A$ and for all $i \in [k_a - 1]$, and hence the solution of Model D satisfies (11d). Therefore, all optimal solutions of Model D are feasible for Model E. The optimal objective function value of Model D is $\sum_{a \in \mathcal{A}} L_a(\lambda_{a,i_0}^* c_{a,i_0} + \lambda_{a,i_0+1}^* c_{a,i_0+1})$ and equals $\sum_{a \in \mathcal{A}} L_a \tilde{c}_a$ by construction.

(ii) Let $(x_a^*, \pi_v^*, y_a^*, c_a^*)_{a \in A, v \in V}$ be an optimal solution of Model E. Consider $i_0 \in [k_a-1]$ such that $D_{a,i_0+1}^{-\beta} \leq y_a^* \leq D_{a,i_0}^{-\beta}$. Then there exist unique $0 \leq \lambda_{a,i_0}, \lambda_{a,i_0+1} \leq 1$ with $\lambda_{a,i_0} + \lambda_{a,i_0+1} = 1$ such that $y_a^* = \lambda_{a,i_0} D_{a,i_0}^{-\beta} + \lambda_{a,i_0+1} D_{a,i_0+1}^{-\beta}$ for all $a \in \mathcal{A}$. We
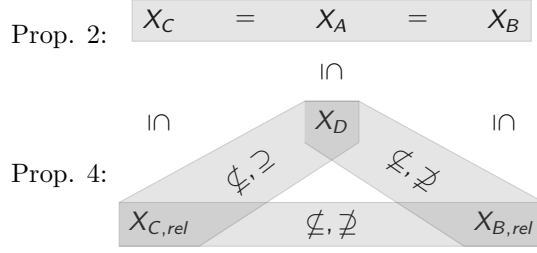
Figure 2: Relation of feasible regions. The highlighted relations in gray are proven as propositions. The other relations $X_A \subseteq X_D$, $X_C \subseteq X_{C, \text{rel}}$ and $X_B \subseteq X_{B, \text{rel}}$ are the canonical continuous relaxations of the corresponding discrete models. Proposition 2 also implies $X_C, X_B \subseteq X_D$. Thus, the split-pipe model can be seen as a relaxation of Models A, B and C.

set $\lambda_{a,i} := 0 \; \forall i \in [k_a] \setminus \{i_0, i_0 + 1\}$, and $x_a := x_a^*$, $\pi_v := \pi_v^* \; \forall a \in \mathcal{A}, \forall v \in \mathcal{V}$, then $(x_a, \pi_v, \lambda_a)_{a \in A, v \in V, i \in [k_a]}$ fulfills (10b) - (10g). Therefore, all optimal solutions of Model E are feasible for Model D.

From (11d) we have $c_a^* = s_{a,i_0} y_a^* + t_{a,i_0}$. However, rearranging $y_a^* = \lambda_{a,i_0} D_{a,i_0}^{-\beta} + \lambda_{a,i_0+1} D_{a,i_0+1}^{-\beta}$ as in (12) to (14) yields $\lambda_{a,i_0} c_{a,i_0} + \lambda_{a,i_0+1} c_{a,i_0+1} = s_{a,i_0} y_a^* + t_{a,i_0}$, hence $\sum_{a \in \mathcal{A}} L_a(\lambda_{a,i_0} c_{a,i_0} + \lambda_{a,i_0+1} c_{a,i_0+1}) = \sum_{a \in \mathcal{A}} L_a c_a^*$. $\qquad \square$

4.4 Comparison of relaxations

In this section, we consider the continuous relaxations of the discrete models. As solvers relax the combinatorial part during the solution procedure, the tightness of the continuous relaxation plays a major role for the performance of the models.

In the following, we denote the feasible region of the Model D by X_D and the feasible regions of the continuous relaxations of the discrete Models B and C by $X_{B, \text{rel}}$ and $X_{C, \text{rel}}$, respectively. Clearly, X_D is the feasible region of the continuous relaxation of Model A.

Proposition 4. *The following relationships hold for the continuous relaxation of Models A, B and C:*

1. *$X_D \subseteq X_{C, \text{rel}}$ but $X_{C, \text{rel}} \neq X_D$, i.e. the continuous relaxation of Model C is weaker than the continuous Model D, but the reverse is not true.*

2. *$X_D \not\subseteq X_{B, \text{rel}}$ and $X_{B, \text{rel}} \not\subseteq X_D$*

3. *$X_{C,\text{rel}} \not\subseteq X_{B, \text{rel}}$ and $X_{B, \text{rel}} \not\subseteq X_{C,\text{rel}}$*

Proof. 1: "$X_D \subseteq X_{C, \text{rel}}$":

18

Let $\left(\tilde{x}_a, \tilde{\pi}_v, \tilde{\lambda}_{a,i}\right)_{a \in A, v \in V, i \in [k_a]} \in X_D$ be a solution of Model D. We show, that it can be transformed to a point in $X_{\text{C, rel}}$. If we set for all pipes $a \in A$: $x_a := \tilde{x}_a$, $\pi_v := \tilde{\pi}_v$ and $\lambda_{a,i} := \tilde{\lambda}_{a,i} \; \forall i \in [k_a]$, Equations (9e) - (9g) and (9i) - (9j) hold. By defining

$$\Delta_{a,i} := \lambda_{a,i} \, \text{sgn}(x_a)|x_a|^{\alpha} L_a R_a D_{a,i}^{-\beta} \quad \forall a \in A \; \forall i \in [k_a] \tag{16}$$

$$\Rightarrow \sum_{i=0}^{k_a} \Delta_{a,i} \stackrel{(16)}{=} \text{sgn}(x_a)|x_a|^{\alpha} \sum_{i=0}^{k_a} \lambda_{a,i} L_a R_a D_{a,i}^{-\beta} \stackrel{(10b)}{=} \pi_v - \pi_w \quad \forall a \in A,$$

the potential loss constraint (9c) and the bounds of $\Delta_{a,i}$ (9h) are satisfied. With $M_{a,i}^{(1)} = \Phi(D_{a,i}, \underline{x}_a)$, $M_{a,i}^{(2)} = \Phi(D_{a,i}, \overline{x}_a)$, the big M-formulation (9d) holds by construction. Equation (9b) also holds, since

$$\stackrel{(16)}{\Rightarrow} \lambda_{a,i} \, \text{sgn}(x_a)|x_a|^{\alpha} = \frac{D_{a,i}^{\beta}}{L_a R_a} \Delta_{a,i} \qquad \forall a \in A \; \forall i \in [k_a],$$

$$\Rightarrow \sum_{i=0}^{k_a} \lambda_{a,i} \, \text{sgn}(x_a)|x_a|^{\alpha} = \sum_{i=0}^{k_a} \frac{D_{a,i}^{\beta}}{L_a R_a} \Delta_{a,i} \stackrel{(9e)}{\Rightarrow} \text{sgn}(x_a)|x_a|^{\alpha} = \sum_{i=0}^{k_a} \frac{D_{a,i}^{\beta}}{L_a R_a} \Delta_{a,i} \; \forall a \in A,$$

hence $X_D \subseteq X_{\text{C, rel}}$.

The remaining relations between the continuous relaxations of the discrete models are shown by counter examples. Throughout we consider a single pipe $a = (v, w)$ with $L_a R_a = 1, D_{a,0}^{\beta} = 1, D_{a,1}^{\beta} = 2$ for given $\alpha > 0$ and $\beta > 0$.

"$X_{\text{C, rel}} \nsubseteq X_D, X_{\text{B, rel}} \nsubseteq X_D$":

Let $b_v = \sqrt[\alpha]{100}$, $b_w = -\sqrt[\alpha]{100}$ and $\pi_v, \pi_w \in [0, 3600]$ be given. Then $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w, x_a) := (0.5, 0.5, 3600, 3500, \sqrt[\alpha]{100}) \notin X_D$, but $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w, x_a, \Delta_{a,0}, \Delta_{a,1})$ $:= (0.5, 0.5, 3600, 3500, \sqrt[\alpha]{100}, 100, 0) \in X_{\text{C, rel}}$ and $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w, x_{a,0}^+, x_{a,1}^+, x_{a,0}^-,$ $x_{a,1}^-, z_a, \Delta_{a,0}^+, \Delta_{a,1}^+, \Delta_{a,0}^-, \Delta_{a,1}^-) := (0.5, 0.5, 3600, 3500, \sqrt[\alpha]{100}, 0, 0, 0, 1, 100, 0, 0, 0) \in X_{\text{B, rel}}$.

"$X_D \nsubseteq X_{\text{B, rel}}, X_{\text{C, rel}} \nsubseteq X_{\text{B, rel}}$":

Now slightly change the above example to $b_v = \sqrt[\alpha]{1000} =: -b_w$ and $\pi_v, \pi_w \in [0, 10000]$. Then $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w, x_a) := (0.5, 0.5, 7500, 0, \sqrt[\alpha]{1000}) \in X_D$, but $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w) := (0.5, 0.5, 7500, 0) \notin X_{\text{B, rel}}$, and $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w, x_a, \Delta_{a,0}, \Delta_{a,1}) :=$ $(0.5, 0.5, 7500, 0, \sqrt[\alpha]{1000}, 5000, 2500) \in X_{\text{C, rel}}$.

"$X_{\text{B, rel}} \nsubseteq X_{\text{C, rel}}$":

We now choose $b_v := \sqrt[\alpha]{75} =: -b_w$ and $\pi_v, \pi_w \in [0, 3600]$. Then $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w,$ $x_{a,0}^+, x_{a,1}^+, x_{a,0}^-, x_{a,1}^-, z_a, \Delta_{a,0}^+, \Delta_{a,1}^+, \Delta_{a,0}^-, \Delta_{a,1}^-) := (0.75, 0.25, 3600, 375, \sqrt[\alpha]{55}, \sqrt[\alpha]{20}, 0, 0, 1,$ $3025, 200, 0, 0) \in X_{\text{B, rel}}$, but $(\lambda_{a,0}, \lambda_{a,1}, \pi_v, \pi_w, x_a) := (0.75, 0.25, 3600, 375, 75) \notin$ $X_{\text{C, rel}}$. $\qquad \square$

5 Computational Study

In this computational study we first investigate the performance of the discrete models and then compare the split-pipe models. In a third step we analyze the difference in the performance of the discrete versus the split-pipe models. Especially from the point of view of practical applications, it is important to see whether the split-pipe approach can lead to significant cost-savings when compared to the discrete looping approach.

5.1 Experimental Setup

The experiments were conducted on a cluster of 64-bit Intel Xeon CPU E5-2670 v2 CPUs at 2.5GHz with 25MB cache and 128GB main memory. In order to safeguard against a potential mutual slowdown of parallel processes, we bind the processes to specific cores and run at most 4 jobs per node. We solve the split-pipe and discrete expansion planning problems to global optimality using the nonconvex MINLP solver SCIP version 5.0.1 with CPLEX 12.7.1 as LP solver and Ipopt 3.12.6 as NLP solver. In all experiments, we ran SCIP with default settings and set a time limit of four hours. This timelimit seems reasonable for practial purposes since expansion planning is carried out with a long-term perspective and decisions on short notice are not required as in daily operations. We solve the instances to ϵ-global optimality with a gap limit of $\epsilon = 10^{-4}$. As performance measure, we use the number of solved instances, the runtime and the number of Branch-and-Bound nodes. In order to reduce the impact of very easy or hard instances in the mean values, we report the shifted geometric mean $\left(\sqrt[n]{\prod (t_i + s)} - s \right)$ for values $t_1, ..., t_n$ with a shift of $s = 10$ for the runtime in [sec] and $s = 100$ for the number of branch and bound nodes. Additionally, we report the arithmetic mean of the computation time to indicate the time required for applications in practice.

The computational study was carried out on different data sets that vary in size and structure and are based on two major types of potential-driven networks: gas networks and water networks. As the starting point we used the *Belgian* gas network from the GAMS model library, which has a simple, almost tree-shaped structure (with the only exception being 5 arcs that have 2 parallel pipelines) and is of a rather smaller size. We continued our computational experiments on the *GasLib-40* network ([SAB$^+$17]), to test the models on a network that is considerably larger and more complex. As the basis for our water instances, the well-known *New York* network (library of instances at the operations research group in Bologna) was employed. Finally, to focus not only on computational difficulties that arise from the size of the network, we systematically investigate the impact of cycles on the model performance and generated extended versions of the *Belgian* networks by adding 2 to 10 additional arcs. The details about the different networks can be found in Table 1 and Figure 3. We note that originally the *Belgian* and the *GasLib-40* network contain a small number of compressor stations the treatment of which is outside the scope of this paper. In the case of the *Belgian* network we used data available in the GAMS model library to model these as normal pipelines, while arcs representing compressor

Table 1: Used networks in the computational study. The diameters describe an aggregated number of the extension candidates for all pipes in the network.

	# Nodes $|\mathcal{V}|$	# Sources	# Sinks	# Arcs $|\mathcal{A}|$	Circuit rank $|\mathcal{A}| - |\mathcal{V}| + 1$	# Diameters	
						Split-Pipe models	Discrete models
Belgian	20	5	9	24	5	308	418
GasLib-40	34	3	29	39	6	488	678
New York	20	1	20	21	2	191	327
Belgian + (2, 4, 6, 8, 10) arcs	20	5	9	26, 28, 30, 32, 34	7, 9, 11, 13, 15	332, 356, 380, 404, 428	452, 486, 520, 554, 588

stations had to be contracted in the case of the *GasLib-40* network due to lack of available data.

For each network, we generated instances that cover a wide range of possible network demands. It is known in practice that increasing the overall network demand typically results in more complex transport situations that stress the networks. Practicioners therefore use the so-called "transport moment" to detect severe demand situations by approximating the transport load in the network (cf. [HKS^{+}18]). It can even be observed in many computational experiments that higher network demands tend to slow down the model performance, provided that the transport situations are not trivially infeasible. Hence, to test the performance of the models in diverse and severe situations, we consider different demand loads that represent the whole spectrum from "easier" instances with lower demands up to "harder" instances with higher demands for each network.

Throughout, we randomly generated scenarios for each given total network demand T according to a random uniform distribution of the demand at sources and sinks. More precisely speaking, for all sources and for all sinks we generated uniformly distributed independent random numbers $r_v \in [0, 1)$ and then normalize the source and sink flows to calculate the demand as $\mathcal{X}_v = T r_v / \sum_{v \in W} r_v$. For each given total demand and for each network we generated 500 scenarios because preliminary tests indicated that this sample size leads to sufficiently stable results for the given distribution with respect to model performances. Including preliminary tests, we calculated about 300000 instances. Due to this large number and variety of instances we can assume that the results provide us with a representative picture for model comparisons. Here, we present a selection of 5500 different demand scenarios and a total of 40000 instances that represent the major effects observed.

Throughout the data sets, the expansion capacities of all instances are given by the diameter candidates $\{D_{a,0}, ..., D_{a,k_a}\}$ resulting from at most triple loops, i.e. in the discrete case each arc of the network may consist of up to three pipelines in parallel, while in the split-pipe case up to three parallel pipelines are possible for each arc segment.

In our experiments, we use a quadratic cost function as this is the most common form in practical applications, (see e.g. [DWS96], [Par05]). The data points for the cost function used for the gas related instances are shown on the left in Fig. 1 and were provided to us by practicioners, and for the water instances it was part of the *New York* network data. (Cost functions from linear to cubic can also be found in the literature, see [OG95] and [BNV12], for example, and
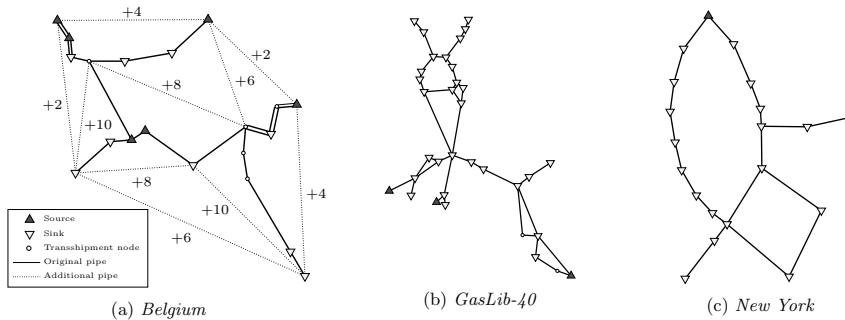
Figure 3: *Belgian* (a) and *GasLib-40* (b) and *New York* networks (c) in schematic view, being illustrated by the bold lines. Dotted lines indicate the *Belgian* network with $+2, +4, +6, +8, +10$ additional arcs.

were used as part of our preliminary tests.)

Summarizing, with the maximum number of parallel loops fixed to 3 and the quadratic cost function being given, one instance is characterized by the choice of network and demand scenario.

The computational results for the comparison of the discrete models can be found in Tables 2 – 5 and for the comparsion of the split-pipe models in Tables 6 – 8. The tables are structured as follows: each column contains the results of 500 instances with the only variable parameter being the total network demand T. The computational results within a column are grouped into three categories: (i) In the section *All instances*, we provide a summary of the results of all 500 instances and indicate the number of solved instances ($\sum \#$ solved), which includes the optimal solved instances (# opt) and the instances that were detected as infeasible (# infeas). Moreover, we state the number of instances, where at least one feasible solution was found within the time limit (# sol found), no matter whether optimality was shown for this solution. For all remaining instances no feasible solution was found within the time limit nor was infeasibility proved. (ii) The section *All opt* reports data for all instances for which all models under comparison have found an optimal solution. (iii) In the section *Only opt by* we compare the models with respect to the instances that they alone were able to solve to global optimality.

In all three cases we report computational time by means of the shifted geometric mean (sgm) and the arithmetic mean (amm). For instances that are solved to optimality by all models, we additionally present the sgm of the number of Branch-and-Bound nodes.

5.2 Comparison of discrete models (Models A, B and C)

We begin our computational experiments with the *Belgian* gas network, which has frequently been used as a tool in research about network optimization (see e.g. [DWS96], [DWS00] and [BNV12]). Our computational results for the net-

22

work can be found in Table 2. Demands are given in 10^6 m^3/day and computational time in seconds. We first consider the performance of our models as depending on the demand. For the lowest demand $T = 100$ about all instances are solved to optimality within a couple of minutes. With increasing demand, computational time increases and the number of instances solved to optimality decreases. However, at a demand of $T = 1000$, we have reached a point where an optimum has been found for less than a quarter of instances and more of 60 % of the instances are detected as infeasible. As a consequence of the high number of infeasibilities, the number of solved instances has increased and computational time decreased. This pattern suggests that the range of demand used here is sufficient to obtain an overview of the general behaviour of the models and that it is not necessary to test the models for a demand of less than $T = 100$ and greater than $T = 1000$. Due to this typical pattern, which was frequently observed also during our preliminary computational experiments, we will generally use a test range from a low demand to a high demand with a large number of infeasibilities.

Table 2: Comparison of discrete models. *Belgian* network with different demand scenarios.

Test Set	demand 100			demand 200			demand 500			demand 1000		
Models	A	B	C	A	B	C	A	B	C	A	B	C
All instances												
# opt	499	500	493	492	496	467	344	365	218	84	111	55
# infeas	0	0	0	0	0	0	5	5	5	310	310	309
$\sum$ # solved	499	500	493	492	496	467	349	370	223	394	421	364
# sol found	500	500	500	500	500	500	495	495	489	189	189	188
time (amm)	80.4	69.7	446.9	552.9	552.2	1652.4	6093.5	5962.8	9187.8	3480.6	3307.6	4319.1
time (sgm)	9.1	39.1	20.3	41.4	159.9	106.7	1424.6	2380.3	3691.8	88.4	151.0	113.2
All opt												
# opt		493			464			209			48	
time (amm)	29.8	64.4	248.7	136.8	336.0	741.2	582.5	1632.7	2548.0	784.9	2783.5	3602.1
time (sgm)	7.8	37.6	17.7	27.0	134.3	71.4	155.4	634.0	673.9	301.5	1248.8	1055.0
nodes (sgm)	293	305	564	2047	2140	5015	33296	26614	151517	99209	60268	364978
Only opt by												
# opt	0	1	0	1	8	0	45	73	0	9	33	0
time (amm)	14400.0	363.7	14400.0	13034.9	3784.9	14400.0	11433.0	8876.4	14400.0	12937.6	8447.2	14400.0
time (sgm)	14400.0	363.7	14400.0	11638.7	1880.6	14400.0	9591.8	6634.3	14400.0	12117.4	7048.4	14400.0

We observe that, with respect to the sgm, Model A is the fastest model throughout, up to a factor 4.3 faster than Model B for a demand of T = 100, while B is the fastest with respect to the amm. This is due to the fact that Model B has a high number of instances that were solved fast (see e.g. those instances that were only solved by B). Model B also solves the largest number of instances. We note though that here Model B does not stand out with respect to the infeasible instances discovered and instances for which a feasible solution has been found.

Let us remark in passing that during our preliminary tests of the two models mentioned in Section 3.4, which were carried out on the rather simple *Belgian* network, the model in [Hum14] ran into the time limit at a demand of $T = 200$ with about 2/3 of all instances, while the model in [BSBB+16] exceeded the time limit in all cases for the same demand. (In constrast to this, the three models considered here ran into the time limit only with 0.8% to 6.6% of the same instances.)

We now turn our attention to *GasLib-40* (see Table 3). Again we can see that

23

Table 3: Comparison of discrete models. *Gaslib-40* network.

Test Set	demand 50			demand 100			demand 500			demand 1000		
Models	A	B	C	A	B	C	A	B	C	A	B	C
All instances												
# opt	500	500	496	302	283	9	0	0	0	0	2	0
# infeas	0	0	0	0	0	0	0	0	0	144	158	139
$\sum$ # solved	500	500	496	302	283	9	0	0	0	144	160	139
# sol found	500	500	500	500	499	498	474	491	491	148	242	112
time (amm)	9.9	119.7	203.4	7781.9	9681.5	14245.3	14400.0	14400.0	14400.0	10264.6	9932.3	10391.5
time (sgm)	4.0	17.2	12.8	3923.0	7777.7	14068.8	14400.0	14400.0	14400.0	1794.8	3176.8	2085.0
All opt												
# opt		496			9			0			0	
time (amm)	6.9	76.2	88.9	55.7	1168.0	5802.7	-	-	-	-	-	-
time (sgm)	3.7	16.1	11.7	45.0	1024.7	3949.2	-	-	-	-	-	-
nodes (sgm)	60	159	284	2689	14905	415265	-	-	-	-	-	-
Only opt by												
# opt	0	0	0	45	26	0	0	0	0	0	2	0
time (amm)	-	-	-	9814.8	12517.3	14400.0	-	-	-	14400.0	8849.7	14400.0
time (sgm)	-	-	-	7990.3	12100.7	14400.0	-	-	-	14400.0	8815.3	14400.0

Model A is the fastest model, with B only having an advantage at a demand of $T = 1000$ with respect to the amm. This time Model A also tends to be better with respect to the number of instances solved. Finally note, that a demand of $T = 500$ is to difficult for all three models, i.e. for unfavourable demand situations we have reached the limits of what is computational tractable. (But obviously, this does not apply that expansion problems on more complex networks cannot be solved for favourable demand situations.)

Table 4: Comparison of discrete models. *New York* network.

Test Set	demand 100			demand 200			demand 500		
Models	A	B	C	A	B	C	A	B	C
All instances									
# opt	0	142	0	9	412	58	0	0	0
# infeas	0	0	0	36	79	73	500	500	500
$\sum$ # solved	0	142	0	45	491	131	500	500	500
# sol found	500	500	500	384	421	348	0	0	0
time (amm)	14400.0	12781.4	14400.0	13227.5	1768.0	11639.1	0.0	0.3	0.0
time (sgm)	14400.0	12183.0	14400.0	8759.6	1000.0	7465.9	0.0	0.3	0.0
All opt									
# opt		0			8			0	
time (amm)	-	-	-	2917.1	734.3	1470.1	-	-	-
time (sgm)	-	-	-	563.3	537.6	881.1	-	-	-
nodes (sgm)	-	-	-	126220	8918	109148	-	-	-
Only opt by									
# opt	0	142	0	0	355	0	0	0	0
time (amm)	14400.0	8903.2	14400.0	14400.0	2190.2	14400.0	-	-	-
time (sgm)	14400.0	8185.9	14400.0	14400.0	1570.5	14400.0	-	-	-

The famous *New York* network is of particular interest for the capacity expansion problem as it is the only network in the water library where the arcs are already laid out with given diameters. Introduced by [SL69], this instances has been used extensively (see e.g. [QLB81], [Bha85], [BDL+12]). On this network, Model B clearly performs best, both regarding the number of instances solved and the runtime (Table 4). We note that we have instances here where Model C performs better than Model A in terms of both solved instances and runtime ($T = 200 \ m^3/s$).

It is well known that optimization problems on potential-driven networks get

Table 5: Comparison of discrete models. Increasing the circuit rank of the *Belgian* network.

Test Set		+0 Arcs			+2 Arcs			+4 Arcs			+6 Arcs			+8 Arcs			+10 Arcs	
Models	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C
All instances																		
# opt	499	500	493	393	416	227	251	364	127	307	297	214	326	262	245	372	248	299
# infeas	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$\sum$ # solved	499	500	493	393	416	227	251	364	127	307	297	214	326	262	245	372	248	299
# sol found	500	500	500	500	500	500	498	479	500	500	381	500	499	357	500	500	316	500
time (amm)	80.4	69.7	446.9	4688.8	4726.2	9094.5	8489.5	7306.5	11352.2	7018.2	9003.2	9463.2	6719.1	9817.5	8613.4	5123.6	9969.2	7151.8
time (sgm)	9.1	39.1	20.3	1368.6	2069.0	4766.3	3492.2	4329.8	6399.7	2327.2	5812.4	4357.7	2489.2	7343.8	3805.4	1443.4	6824.2	2262.1
All opt																		
# opt		493			221			117			183			192			207	
time (amm)	29.8	64.4	248.7	545.8	859.6	2632.2	753.0	1440.6	2148.4	809.8	3632.4	2276.1	1381.0	4934.4	2254.0	969.5	4870.5	1899.4
time (sgm)	7.8	37.6	17.7	249.5	585.5	1213.8	258.0	787.3	495.3	278.2	1879.5	663.9	489.2	3291.6	814.1	326.4	2712.0	510.2
nodes (sgm)	294	305	564	33817	8509	146470	20679	7216	25610	18061	16784	33299	30406	27272	37800	19120	20619	15523
Only opt by																		
# opt	0	1	0	22	39	0	24	129	0	48	36	1	62	25	6	67	3	8
time (amm)	14400.0	363.7	14400.0	11918.7	9747.6	14400.0	13283.2	8441.5	14400.0	9907.2	12030.4	14358.8	8811.9	12824.1	13859.5	5925.5	14154.0	13321.2
time (sgm)	14400.0	363.7	14400.0	10717.3	8217.9	14400.0	12488.9	7165.3	14400.0	8101.0	11101.1	14352.9	6849.8	12245.5	12989.9	3866.6	13979.7	11744.4

more difficult to solve the more cycles they contain because the existence of cycles leads to more complex patterns of flow directions (cf. the literature review in [SS16]). To systematically test our models also with respect to this type of difficulty, we successively increase the circuit rank by adding up to ten new pipelines to the *Belgian* network, which results in five new networks *Belgium* + (2, 4, 6, 8, 10) arcs, see Fig. 3a. Here, the intension was to add arcs that connect different regions of the network and have a high impact on the network topology. For the new pipelines we use in our data set an original diameter $D_{a,0}$ that is equal to the average diameter of the existing pipelines in the network and use moderate demand scenarios with $T = 200$.

We observe (see Table 5) that for increasing circuit rank the number of instances solved to optimality at first decreases and then increases again for Models A and C, while it continues to decrease for Model B. A similar pattern applies with respect to computational time, where the runtime at first increases for Models A and C and then decreases for the instances with 6 or more additional arcs, while the runtime of Model B increases nearly throughout. This leads to a situation where Model A turns out to have the best sgm runtime for all circuit ranks, B solves the most instances for smaller circuit ranks, and A solves the most instances for larger circuit ranks. Finally, while Model C shows poor results for small circuit ranks, it gains ground with increasing circuit ranks and performs second best for the instances with *Belgium* + 10 arcs.

In sum, we can state that overall Models B and A perform better than Model C, with Model A clearly outperforming Model B on problems with a higher circuit rank (including *GasLib-40*). The performance of Model C may partly be a consequence of the fact that Model A, as we have shown in Section 4.4, has a tighter relaxation than Model C, even though there is a smaller number of entire data sets and a larger number of individual instances where Model C outperforms both Model A and Model B, in particular instances with a higher circuit rank. Model C is therefore a valuable alternative in cases where Models A and B require too much computational time to prove infeasibility or solve an instance to optimality.

5.3 Comparison of split-pipe models (Models D and E)

Again we start with the *Belgian* network (Table 6) and proceed from easily solved instances to scenarios with a large number of infeasible instances. It turns out that both models solve a comparable number of instances, with the new Model E being somewhat faster. It is worth noting that for a demand of $T = 500$, i.e. for more difficult instances, 10% of the instances are solved to optimality by only one of the two models, i.e. in this case both models complement each other well.

Table 6: Comparison of split-pipe models. *Belgian* network with different demand scenarios.

	Test Set	demand 100		demand 200		demand 500		demand 1000	
	Models	D	E	D	E	D	E	D	E
All instances	# opt	500	500	494	496	318	315	77	82
	# infeas	0	0	0	0	5	5	310	310
	$\sum$ # solved	500	500	494	496	323	320	387	392
	# sol found	500	500	500	500	495	495	189	189
	time (amm)	1.0	0.7	256.2	169.9	5850.9	5891.4	3655.4	3533.2
	time (sgm)	1.0	0.6	7.0	4.8	676.1	640.9	80.3	79.1
All opt	# opt	500		492		269		60	
	time (amm)	1.0	0.7	84.0	42.0	841.4	951.2	1903.4	1657.4
	time (sgm)	1.0	0.6	5.5	3.8	80.0	78.3	205.0	196.8
	nodes (sgm)	24	21	240	197	16381	15153	50809	53529
Only opt. by	# opt	0	0	2	4	49	46	17	22
	time (amm)	-	-	9662.2	5916.3	8554.7	8457.3	10335.6	9148.2
	time (sgm)	-	-	3106.3	415.1	4037.7	3234.6	6748.7	6067.2

To compare the performance on gas networks of larger sizes with a more complex cyclic structure, we turn to the *GasLib-40* network (Table 7 on the left). Here, Model E performs clearly better: it solves up to 11 % more instances (demand $T = 100, 1000$) and requires less runtime (up to a factor of 2.5 in the case of $T = 50$). To further test the performance of the split-pipe models, we consider the *New York* water network again (Table 7 on the right). Here the results are balanced: both models solve the majority of instances in at most 3 seconds (sgm).

Finally, we investigate the performance of the models for increasing circuit rank. We observe (Table 8) that our novel Model E consistently outperforms D, with the ratio of the runtimes of the model becoming the more favourable for E the higher the circuit rank, up to a factor of 13.9 for 10 additional arcs. Similarly, Model E performs significantly better with respect to the number of solved instances, with the best ratios of solved instances reached at 16.2 for *Belgium* + 8 arcs and 10.0 for *Belgium* + 10 arcs.

On the basis of our computational experiments for the split-pipe case, we can clearly recommend the use of our novel model instead of the model from the literature, particularly for networks with a more complex cyclic structure, even though one will, of course, encounter instances where the other model is more successful.

Table 7: Comparison of the split-pipe models. *GasLib-40* and *New York* networks.

	Test Set	GasLib-40								New York					
		demand 50		demand 100		demand 500		demand 1000		demand 100		demand 200		demand 500	
	Models	D	E	D	E	D	E	D	E	D	E	D	E	D	E
All instances	# opt	499	500	371	413	237	244	206	243	500	500	421	421	0	0
	# infeas	0	0	0	0	0	0	160	160	0	0	79	79	500	500
	∑ # solved	499	500	371	413	237	244	366	403	500	500	500	500	500	500
	# sol found	500	500	500	500	500	500	340	340	500	500	421	421	0	0
	time (amm)	32.4	3.2	4045.7	3071.0	9273.6	9002.8	6154.6	5477.9	3.1	2.9	1.9	2.9	0.0	0.0
	time (sgm)	3.5	1.4	215.7	164.3	4832.4	4452.5	756.7	687.8	3.0	2.8	1.8	2.8	0.0	0.0
All opt	# opt	499		354		214		186		490		417		0	
	time (amm)	3.6	3.2	375.3	341.7	3291.6	2738.2	5434.6	4247.7	3.1	2.9	1.3	2.4	-	-
	time (sgm)	3.3	1.4	40.4	37.1	1300.6	1050.9	2842.7	2226.8	3.0	2.8	1.3	2.4	-	-
	nodes (sgm)	42	16	4270	4040	157062	33984	338695	285031	698	677	182	176	-	-
Only opt by	# opt	0	1	17	59	23	30	20	57	0	0	0	0	0	0
	time (amm)	14400.0	4.5	11605.1	5349.1	10890.4	10570.0	12434.0	10905.4	-	-	-	-	-	-
	time (sgm)	14400.0	4.5	5266.3	1317.9	7837.6	8512.7	11240.0	10117.4	-	-	-	-	-	-

Table 8: Comparison of split-pipe models. Increasing the circuit rank of the *Belgian* network.

	Test Set	+0 Arcs		+2 Arcs		+4 Arcs		+6 Arcs		+8 Arcs		+10 Arcs	
	Models	D	E	D	E	D	E	D	E	D	E	D	E
All instances	# opt	500	500	155	235	58	136	26	170	12	194	30	308
	# infeas	0	0	0	0	0	0	0	0	0	0	0	0
	∑ # solved	500	500	155	235	58	136	26	170	12	194	30	308
	# sol found	500	500	500	500	500	500	499	500	500	500	500	500
	time (amm)	1.0	0.7	10188.8	7963.9	12928.2	10819.0	13766.4	9781.5	14087.2	9346.4	13634.8	6224.3
	time (sgm)	1.0	0.6	3192.4	1173.1	9611.2	3952.4	11843.6	2790.4	12944.1	2675.5	11243.6	844.2
All opt	# opt	500		101		35		22		12		29	
	time (amm)	1.0	0.7	761.7	477.2	1087.5	1015.0	2047.6	246.8	1364.9	172.1	1275.8	23.4
	time (sgm)	1.0	0.6	91.2	34.4	258.5	37.2	237.8	14.5	160.3	32.0	194.0	10.9
	nodes (sgm)	24	21	21956	4642	71439	4495.4	29888	1023	19818	2046	17299	581.7
Only opt by	# opt	0	0	54	134	23	101	4	148	0	182	1	279
	time (amm)	-	-	10526.8	4762.4	12222.9	3738.5	14103.6	1256.1	14400.0	1454.7	14392.9	1289.6
	time (sgm)	-	-	3777.5	407.2	8689.6	387.3	13637.5	155.7	14400.0	199.6	14392.4	172.6

5.4 Comparison of discrete and split-pipe models

To conclude our computational experiments, we compare the discrete with the split-pipe approaches. As mentioned in the introduction, this is the first time that such a comparison is carried out for networks of a practically relevant size and complexity. Of course, one may expect that the split-pipe model will perform better on average, but as our computational experiments show (compare the results for *Belgium* +4, +6, and +8 arcs in Tables 5 and 8, for example), there are data sets where the discrete approach yields more optimal solutions within the time limit of 4 hours than the split-pipe approach. For this reason it is unclear what size the performance gap may actually take.

In the following, we will look at two criteria: (a) potential cost savings and (b) computational time.

Concerning cost savings, Table 9 shows the average gain of the best known solutions of the split-pipe models over the discrete models. Here, we consider

only those instances where an expansion actually takes place and where either at least one split-pipe and at least one discrete model have been solved to optimality (column *Optimal*) or where both approaches provide at least one solution (column *Feasible*). For these instances, we then consider the average gain of the best known split-pipe over the best-known discrete one. The column *Feasible* additionaly shows the percentage of instances where the best-known solution is provided by the split-pipe models.

Table 9: Gain of Split-Pipe over Discrete Problem.

|		Optimal		Feasible		
Network	Test Set	Gain [%]	# instances	Gain [%]	# instances	SP better on # instances [%]
Belgium	demand: 100	2.2	500	2.2	500	100.0
Belgium	demand: 200	1.6	498	1.6	500	100.0
Belgium	demand: 500	1.1	332	1.2	495	100.0
Belgium	demand: 1000	1.3	74	1.4	189	100.0
GasLib-40	demand: 50	321.5	147	321.5	147	100.0
GasLib-40	demand: 100	2.1	326	3.7	500	100.0
GasLib-40	demand: 500	-	0	10.7	498	100.0
GasLib-40	demand: 1000	0.7	2	8.0	175	100.0
New York	demand: 100	1.0	142	1.1	500	100.0
New York	demand: 200	0.8	413	0.8	415	100.0
New York	demand: 500	-	0	-	0	-
Belgium+2 arcs	Cycle Rank	1.6	268	1.5	500	99.8
Belgium+4 arcs	Cycle Rank	8.2	151	5.3	499	96.4
Belgium+6 arcs	Cycle Rank	14.1	169	12.1	498	98.8
Belgium+8 arcs	Cycle Rank	17.4	185	14.5	498	99.6
Belgium+10 arcs	Cycle Rank	71.0	284	50.7	490	100.0
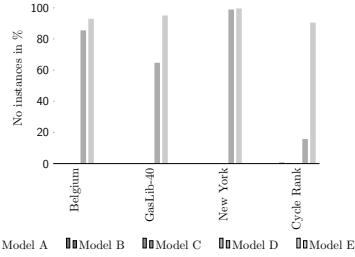
Figure 4: Percentage of instances for which the best solution was found.

As Table 9 demonstrates, the split-pipe approach yields cost savings on all data sets. While the results for the comparably simple networks of *Belgium* and *New York* are rather low, the benefit of realizing a split-pipe solution can be considerable for networks with a complex cyclical structure. Moreover, Figure 4 shows that for all feasible instances with non-zero objective function value, the best solutions were practically always found by the split-pipe models, with our novel Model E delivering the best results. Even more, further analysis of the data presented in the Tables 2 - 8 reveals that the split-pipe models optimally solve or detect (in)feasibibilty in all cases except 1 (Model E) or 2 (Model D) instances, whereas the number of instances with unknown status is much higher for the discrete models, namely 318 (Model A), 578 (Model B) and 348 (Model C) out of the 8000 instances we have calculated per model. Therefore, in view of the much larger number of feasible solutions by the split-pipe models, the economic benefit of these models goes well beyond the cost savings and the percentage of instances depicted in Table 9 and Figure 4.

To compare the overall runtime performance of all discrete and split-pipe models, we use a performance profile ([DM02]). It is based on the performance ratio, i.e. the runtime of a particular data set with the model under consideration divided by the best runtime for that data set with any of the five models. The performance profile describes on the y-axis the fraction of instances among all solved (i.e. optimal or infeasible) instances that the model could solve with a performance ratio of up to the corresponding number on the x-axis. Clearly, models are to be preferred when their profile shows higher y-values for fixed x-values and lower x-values for fixed y-values. To exclude trivial cases, we disregard instances that were not solved by any model and those that were solved by all models in less than 1 second CPU time.
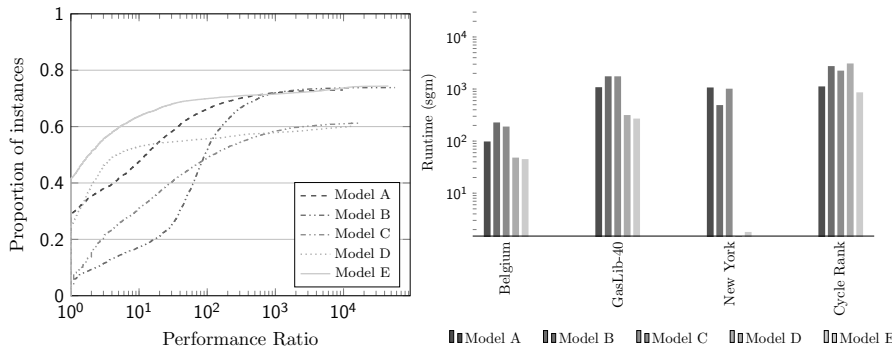
Figure 5: Performance Profile.　Figure 6: Performance on different data sets.

As we can see, the runtime of Model E dominates the runtime of all other models across the spectrum of performance ratios (Fig. 5). Further insights about the model performances for the different data sets can be gathered from Figure 6, which depicts the sgm of the runtime of all instances for different network types and confirms the dominance of the split-pipe models. Again Model E turns out to perform best.

5.5　Recommendations

To conclude this section, our computational experiments suggest the following recommendations for practitioners:

1. In view of the advantages of the split-pipe approach discussed in the previous subsection, practitioners should use the split-pipe approach whenever possible.

2. Our novel Model E should be the model of choice as it leads to more instances that are solved, to more instances for which feasible solutions are found, to a higher percentage of instances with a better objective function value and cost savings, and performs better with respect to computational time.

3. When the split-pipe approach is not possible, Models A or B are likely to be the most useful.

4. Among these two models, Model A should be tried out first on networks with a more complex cyclical structure.

5. Finally, Models C and D may be useful complements when dealing with instances that happen to be particularly hard for the other models.

6 Conclusion

We studied the problem of capacity expansion of potential-driven networks using loops. We showed properties of the looping problem, such as its nonconvexity, and, building on an existing method of selecting cost-minimal loop diameters a priori, we presented a model reduction approach for multiple loops. On this basis we introduced new models for both discrete and split-pipe looping and contrasted these with existing models for the looping problem and related problems in the literature, both theoretically and empirically. This was also the first time that discrete and split-pipe approaches were compared for networks of a practically relevant size and complexity. The performance of the models was analyzed in an extensive computational study with a large set of demand vectors and diverse networks of different sizes and topologies, including network variations for different circuit ranks, and led to recommendations regarding the use of the different models. In particular, our experiments showed that overall our novel Model E outperforms the existing models with respect to computational time, the number of solutions found, the number of instances solved and cost savings.

An interesting avenue for further research would be to improve our split-pipe Model E by generating the efficient frontier dynamically. Instead of adding all constraints for the frontier globally to the model, they could be generated dynamically during the solving process whenever the frontier is violated by the current LP solution. This could drastically reduce the model size, depending on the number of arcs and extreme points of the efficient frontier.

Moreover, while we allowed all arcs to be looped, we could reduce the model size by a heuristic pre-selection of the arcs that are most likely to be looped. Such an approach could gather the pipes that have a higher expected potential loss along the pipe, which could be determined by simple auxiliary models.

Finally, further work could also include loop expansion planning in the context of time-dynamic commodity transport.

7 Acknowledgements

References

[AAB⁺13]   Jean André, Stéphane Auray, Jean Brac, Daniel De Wolf, Guy Maisonnier, Mohamed-Mahmoud Ould-Sidi, and Antoine Simon-

net. Design and dimensioning of hydrogen transmission pipeline networks. *European Journal of Operational Research*, 229(1):239–251, 2013.

[ABC09]   Jean André, Frédéric Bonnans, and Laurent Cornibert. Optimization of capacity expansion planning for gas transportation networks. *European Journal of Operational Research*, 197(3):1019–1027, 2009.

[AS77]   Elyahu Alperovits and Uri Shamir. Design of optimal water distribution systems. *Water Resources Research*, 13(6):885–900, 1977.

[Bal85]   Egon Balas. Disjunctive programming and a hierarchy of relaxations for discrete optimization problems. *SIAM Journal on Algebraic Discrete Methods*, 6(3):466–486, 1985.

[BD56]   Garrett Birkhoff and Joaquín B Diaz. Non-linear network problems. *Quarterly of Applied Mathematics*, 13(4):431–443, 1956.

[BDL$^+$12]   Cristiana Bragalli, Claudia D'Ambrosio, Jon Lee, Andrea Lodi, and Paolo Toth. On the optimal design of water distribution networks: a practical minlp approach. *Optimization and Engineering*, 13(2):219–246, 2012.

[Bha85]   Pramod R Bhave. Optimal expansion of water distribution systems. *Journal of Environmental Engineering*, 111(2):177–197, 1985.

[BNV12]   Frédéric Babonneau, Yurii Nesterov, and Jean-Philippe Vial. Design and operations of gas transmission networks. *Operations Research*, 60(1):34–47, 2012.

[BS79a]   Sita Bhaskaran and Franz JM Salzborn. Optimal design of gas pipeline networks. *Journal of the Operational Research Society*, 30(12):1047–1060, 1979.

[BS79b]   Sita Bhaskaran and Franz JM Salzborn. Optimal diameter assignment for gas pipeline networks. *The ANZIAM Journal*, 21(2):129–144, 1979.

[BSBB$^+$16]   Conrado Borraz-Sánchez, Russell Bent, Scott Backhaus, Hassan Hijazi, and Pascal Van Hentenryck. Convex relaxations for gas expansion planning. *INFORMS Journal on Computing*, 28(4):645–656, 2016.

[BSR94]   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.

[CCH$^+$78]   M. Collins, L. Cooper, R. Helgason, Kennington, and L. LeBlan. Solving the pipe network analysis problem using optimization techniques. *Management Science*, 24(7):747–760, 1978.

[CG98]   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.

[DM02]   Elizabeth D Dolan and Jorge J Moré. Benchmarking optimiza-
           tion software with performance profiles. *Mathematical program-
           ming*, 91(2):201–213, 2002.

[DWB12]   Daniel De Wolf and Bouchra Bakhouya. Optimal dimensioning of
           pipe networks: the new situation when the distribution and the
           transportation functions are disconnected. In *Operations Research
           Proceedings 2011*, pages 369–374. Springer, 2012.

[DWS96]   Daniel De Wolf and Yyes Smeers. Optimal dimensioning of pipe
           networks with application to gas transmission networks. *Operations
           Research*, 44(4):596–608, 1996.

[DWS00]   Daniel De Wolf and Yves Smeers. The gas transmission problem
           solved by an extension of the simplex algorithm. *Management Sci-
           ence*, 46(11):1454–1465, 2000.

[ESBT94]   Gideon Eiger, Uri Shamir, and Aharon Ben-Tal. Optimal de-
           sign of water distribution networks. *Water Resources Research*,
           30(9):2637–2646, 1994.

[Fas99]   Hans-Georg Fasold. Technisch/wirtschaftliche Gesichtspunkte bei
           der Leitungsparallelisierung. *Das Gas- und Wasserfach / Gas,
           Erdgas*, 140(12):871–876, 1999.

[FD87]   Okitsugu Fujiwara and Debashis Dey. Two adjacent pipe diameters
           at the optimal solution in the water distribution network models.
           Water Resources Research, 23(8):1457–1460, 1987.

[FK90]   Okitsugu Fujiwara and Do Ba Khang. A two-phase decomposition
           method for optimal design of looped water distribution networks.
           Water Resources Research, 26(4):539–549, 1990.

[Ges85]   Johannes Gessler. Pipe network optimization by enumeration. In
           Computer Applications in Water Resources, pages 572–581. ASCE,
           1985.

[GHC72]   Ishwar Gupta, M Zia Hassan, and John Cook. Linear programming
           analysis of a water supply system with multiple supply points. *AIIE
           Transactions*, 4(3):200–204, 1972.

[HKS+18]   Benjamin Hiller, Thorsten Koch, Lars Schewe, Robert Schwarz,
           and Jonas Schweiger. A system to evaluate gas network capacities:
           Concepts and implementation. *European Journal of Operational
           Research*, 270(3):797–808, 2018.

[HMN91]   Claus Thorp Hansen, Kaj Madsen, and Hans Bruun Nielsen. Opti-
           mization of pipe networks. *Mathematical Programming*, 52(1-3):45–
           58, 1991.

[Hum14]   Jesco Humpola. *Gas network optimization by MINLP*. PhD thesis,
           Technische Universität zu Berlin, 2014.

[Jac68]   Shmuel LS Jacoby. Design of optimal hydraulic networks. *Journal
           of the Hydraulics Division*, 94(3):641–662, 1968.

[Kat59]   Donald La Verne Katz. *Handbook of Natural Gas Engineering.* McGraw-Hill, 1959.

[KGM68]   David Karmeli, Yoram Gadish, and Selwyn Meyers. Design of optimal water distribution networks. *Journal of the Pipeline Division*, 94(1):1–10, 1968.

[KHPSe15]   Thorsten Koch, Benjamin Hiller, Marc E. Pfetsch, and Lars Schewe (eds.). *Evaluating Gas Network Capacities.* SIAM-MOS Series on Optimization, 2015.

[KS89]   Avner Kessler and Uri Shamir. Analysis of the linear programming gradient method for optimal design of water supply networks. *Water Resources Research*, 25(7):1469–1480, 1989.

[Lia71]   Tung Liang. Design conduit system by dynamic programming. *Journal of the Hydraulics Division*, 97(3):383–393, 1971.

[Mau77]   JJ Maugis. Etude de réseaux de transport et de distribution de fluide. *RAIRO - Operations Research - Recherche Opérationnelle*, 11(2):243–248, 1977.

[OG95]   A. Osiadacz and M. Górecki. Optimization of pipe sizes for distribution gas network design. In *PSIG Conference, 27th Annual Meeting*. Pipeline Simulation Interest Group, 1995.

[PAC08]   S. Pietrasz, M. Antunes, and L. Cornibert. The problem of reinforcing regional gas transmission networks : A performance study of three optimization methods. In *EngOpt 2008 - International Conference on Engineering Optimization*, pages 1–10, 2008.

[Par05]   N. Parker. Using natural gas transmission pipeline costs to estimate hydrogen pipeline costs. Technical Report UCD-ITS-RR-04-3, Institute of Transportation Studies, University of California, Davis, 2005.

[QLB81]   Gerald E Quindry, Jon C Liebman, and E Downey Brill. Optimization of looped water distribution systems. *Journal of the Environmental Engineering Division*, 107(4):665–679, 1981.

[Rag13]   Arvind U Raghunathan. Global optimization of nonlinear network design. *SIAM Journal on Optimization*, 23(1):268–295, 2013.

[RB82]   William F Rowell and J Barnes. Obtaining the layout of water distribution systems. *Journal of the Hydraulics Division*, 108(1):137–148, 1982.

[RFR+70]   B Rothfarb, H Frank, DM Rosenbaum, Kenneth Steiglitz, and Daniel J Kleitman. Optimal design of offshore natural-gas pipeline systems. *Operations Research*, 18(6):992–1020, 1970.

[RSS+19]   Martin Robinius, Lars Schewe, Martin Schmidt, Detlef Stolten, Johannes Thürauf, and Lara Welder. Robust optimal discrete arc sizing for tree-shaped potential networks. *Computational Optimization and Applications*, 73(3):791–819, 2019.

[SAB⁺17]   Martin Schmidt, Denis Aßmann, Robert Burlacu, Jesco Humpola, Imke Joormann, Nikolaos Kanelakis, Thorsten Koch, Djamal Oucherif, Marc E. Pfetsch, Lars Schewe, Robert Schwarz, and Mathias Sirvent. Gaslib – a library of gas network instances. *Data*, 2(4), 2017.

[SDM94]   Angus R Simpson, Graeme C Dandy, and Laurence J Murphy. Genetic algorithms compared to other techniques for pipe optimization. *Journal of Water Resources Planning and Management*, 120(4):423–443, 1994.

[SL69]   John C Schaake and Fu Hsiung Lai. *Linear programming and dynamic programming application to water distribution network design.* MIT Hydrodynamics Laboratory, Department of Civil Engineering, Report No. 116, 1969.

[SS16]   Naoshi Shiono and Hisatoshi Suzuki. Optimal pipe-sizing problem of tree-shaped gas distribution networks. *European Journal of Operational Research*, 252(2):550–560, 2016.

[SSL01]   Hanif D Sherali, Shivaram Subramanian, and GV Loganathan. Effective relaxations and partitioning schemes for solving water distribution network design problems to global optimality. *Journal of Global Optimization*, 19(1):1–26, 2001.

[SW97]   Dragan A Savic and Godfrey A Walters. Genetic algorithms for least-cost design of water distribution networks. *Journal of Water Resources Planning and Management*, 123(2):67–77, 1997.

[WCS01]   Thomas M Walski, Donald V Chase, and Dragan A Savic. Water distribution modeling. Technical report, University of Dayton, Civil and Environmental Engineering and Engineering Mechanics Faculty Publications, Paper 17, 2001.

[Wey12]   T. R. Weymouth. Problems in natural gas engineering. *Transactions of the American Society of Mechanical Engineers*, 34(1349):185–231, 1912.

[YTB84]   DF Yates, AB Templeman, and TB Boffey. The computational complexity of the problem of determining least capital cost designs for water supply networks. *Engineering Optimization*, 7(2):143–155, 1984.

[ZZ96]   Jianzhong Zhang and Detong Zhu. A bilevel programming method for pipe network optimization. *SIAM Journal on Optimization*, 6(3):838–857, 1996.