FELIX HENNINGS (iD), LOVIS ANDERSON (iD),
KAI HOPPMANN (iD), MARK TURNER (iD), THORSTEN KOCH (iD)

Controlling transient gas flow in real-world pipeline intersection areas

Controlling transient gas flow in real-world pipeline intersection areas

Felix Hennings, Lovis Anderson, Kai Hoppmann, Mark Turner,
Thorsten Koch

June 10, 2021

Abstract

Compressor stations are the heart of every high-pressure gas transport network. Located at intersection areas of the network they are contained in huge complex plants, where they are in combination with valves and regulators responsible for routing and pushing the gas through the network. Due to their complexity and lack of data compressor stations are usually dealt with in the scientific literature in a highly simplified and idealized manner. As part of an ongoing project with one of Germany's largest Transmission System Operators to develop a decision support system for their dispatching center, we investigated how to automatize control of compressor stations. Each station has to be in a particular configuration, leading in combination with the other nearby elements to a discrete set of up to 2000 possible feasible operation modes in the intersection area. Since the desired performance of the station changes over time, the configuration of the station has to adapt. Our goal is to minimize the necessary changes in the overall operation modes and related elements over time, while fulfilling a preset performance envelope or demand scenario. This article describes the chosen model and the implemented mixed integer programming based algorithms to tackle this challenge. By presenting extensive computational results on real world data we demonstrate the performance of our approach.

1   Introduction

Throughout the past years, the mathematics of gas transport has been an intensively studied topic. While natural gas was, is, and will be one of the major energy sources in Germany, making the efficient and safe transport a field of high economical and political relevance [9], the task is also challenging from a mathematical point of view. One such challenge can be found in the compressors, which push the gas through the network by increasing its pressure. Compressors are typically set up as a compressor station, whereby multiple compressor units can be placed in specific configurations and dynamically adjusted to meet the current needs, allowing for different compression ratios and flow rates. At intersections of major transportation pipelines, arrangements of multiple compressor stations as well as other elements like valves or pressure regulators can be found. Such an arrangement makes it possible to choose different connections of the intersecting pipelines, and operate the system for various flow directions and pressure levels.

1

To optimize control of these areas taking all the technical restrictions into account is already combinatorially challenging. The complexity of the problem is further increased by the physics of gas flow. For pipes this physics is described by the Euler Equations [26], a set of nonlinear hyperbolic partial differential equations (PDEs), which even in simplified versions yield computationally challenging constraints.

Historically, research focused first on the simulation of gas flow, i.e., dealing with the partial differential equations given all the discrete decisions. This field has been studied for many decades already, see for example [3] and the references therein. Over recent years, the optimization of gas transport including also the combinatorial aspects have gained more and more attention. In [29] a general overview over optimization problems related to natural gas is given, which includes but is not restricted to the transport of gas. Most of the corresponding literature mentioned so far considers the stationary gas transport problem, which searches for one stable network state, making an algebraic description of the gas flow possible. An overview of state-of-the-art approaches for the stationary case can be found in [19] and [28], which consider large real-world instances and a huge amount of detail regarding the different network elements like compressors.

This article deals with the more challenging variant of the problem: The transient gas transportation problem. Here, the goal is to find a set of control decisions on the elements over a future time horizon. For this problem, research is still in early stages. One of the first publications on transient gas transport optimization was [23], who presented a mixed integer programming (MIP) model for the problem. In contrast to the structures described above, they only used a model consisting of single compressor units, whose compression capabilities are limited by a minimum and maximum power bound. This non-linear power bound as well as the non-linear pipe equations are approximated by piecewise linear functions. To solve the model for the objective of minimizing the compressor fuel costs, they used a special branching scheme for the piecewise linear functions as well as a simulated annealing heuristic described in [21]. A little later, [6] also presented a solution to the problem of minimizing the compressor fuel costs. They modeled the problem according to [23], using the same model for compressors and approximated the non-linear constraints by piecewise linear functions. However, they combined solving this MIP with solving a non-linear problem (NLP) formulation of the problem in an alternating way. From the solution of that non-linear problem they deduce a refinement of the piecewise linear approximations and repeat this procedure until finally arriving at a solution to the overall mixed-integer non-linear problem (MINLP) within a chosen approximation error. Other approaches tackle transient transport optimization problems, but neglect the discrete nature of some of the elements and therefore purely optimize over continuous variables, i.e., solve NLP problems. We mention as example the work of [33] and [22], who decide on the compression ratios of compressors, while again minimizing their fuel consumption. Very recently a few more studies on transient gas network optimization have been published. In [13] a specialized branching rule is used to solve a MINLP formulation of the problem with the objective to minimize fuel consumption. For the compressors they introduced the theoretical concept of different modes to switch between configurations, which each have separate feasible region. However, in the end they restrict to exactly one mode with nearly unrestricted compression capabilities for their experiments. Another

2

approach combining different specialized solving techniques is presented in [11], where iteratively a MIP model and a NLP model are solved for each single time step. These two models arise from the use of a special discretization of the Euler equations. For compression, both models use single compressor units featuring a linear feasible region. In contrast to the other mentioned publications, the objective function in [11] was not to minimize fuel cost, but to comply as well as possible with a set of future pressure and flow values given at the boundary nodes. Finally we mention [4], who considered maximizing the amount of temporarily stored gas in the network while maintaining a feasible transient control of the elements. They also introduce a new discretization of the Euler Equations, which results in a formulation close to the algebraic form of the stationary model. They then use this discretization to obtain globally optimal solutions. For the compressors however, again only single units restricted by upper and lower bounds in the compression ratio as well as the absolute pressure difference have been modeled. This problem is solved by alternating between solving a MIP model, which is obtained by replacing the non-linear constraints by piecewise linear functions, and a NLP model, in which the discrete decisions from the MIP are already fixed.

All approaches of the above mentioned publications on transient gas network optimization use a rather idealized model to deal with compressor stations. In contrast to this, we present in this article a transient gas network optimization problem featuring a compressor model with a so far unmatched amount of detail. The model is based on the modeling presented in [19] for the stationary case and takes the above described substructures into account. We focus on those network areas containing the compressor stations as well as additional active elements and call them *network stations*, an example station is presented in Figure 1. These areas contain the majority of active elements in the network. Regarding the number of contained elements they are comparable to, or even larger than the networks considered in the above mentioned literature on transient optimization. However, one difference to general gas network problems is the shortness of pipes due to the proximity of the elements in a network station, see Table 1 in Section 5 for an overview of different network stations. Because of their shortness, the pipes ability to store gas is negligible owing to their small volumes. Furthermore, the pressure loss induced by friction in the pipe is dependent on the its length and has therefore reduced impact. This allows us to use a linear pipe model as introduced in [16] without losing much accuracy and still producing realistic results.

For each network station we are given an initial state as well as future demands in terms of both inflow and pressure levels at the boundaries. The goal is then to find a feasible control of all the network elements over time, which can be interpreted as a recommendation for network operators on how to control the network in the future. The overall objective is to meet the future demands as best as possible while simultaneously minimizing the total number of control changes. The latter is preferable since it reduces strain on the technical elements and enables the gas network operators to understand and actually perform the desired control recommendations.

The rest of the article is structured as follows: In Section 2 we will describe the mathematical models for all used elements and formulate a corresponding MIP model. The preprocessing needed to convert the given compressor data into a linear description of the feasible operating range will be introduced in Section 3. However, solving the resulting MIP is quite challenging due to the
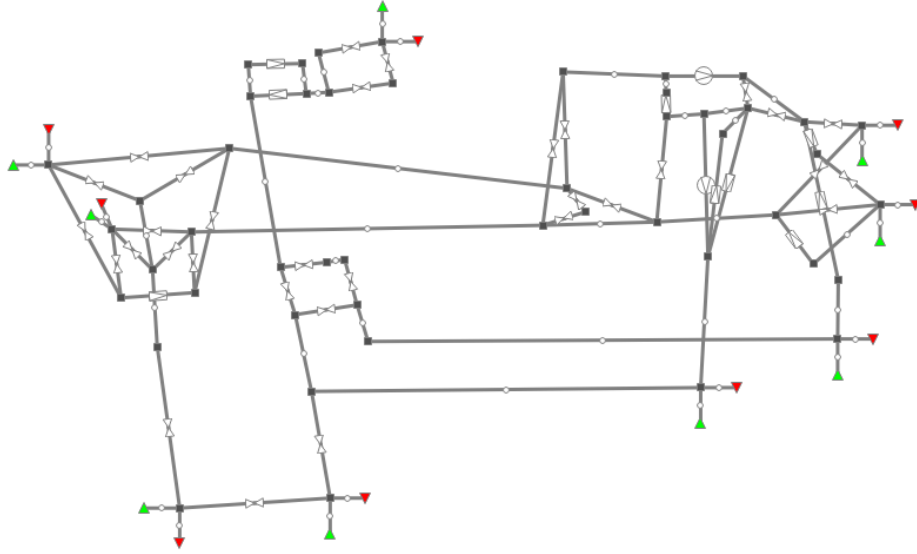
Figure 1: Example of the medium size network station E, see Table 1 for more details to its properties. The colored triangles represent the entry ▲ and exit ▼ nodes of the station. Furthermore we have denoted the single network elements by ─○─ (pipe), ─⋈─ (valve), ─▷─ (regulator), and ─⊘─ (compressor station).

complex compressor station model. We therefore propose a different solution approach in Section 4 based on solving slightly adjusted versions of the presented MIP. Section 5 then presents our results on computing solutions to a large number of real world networks situations. We finish with the conclusion in Section 6.

2 Mathematical model

We model the gas network as a directed graph $G = (\mathcal{V}, \mathcal{A})$ in which the arcs $\mathcal{A}$ represent the different network elements and nodes $\mathcal{V}$ represent the junctions of the arcs. We split $\mathcal{A}$ into individual sets $\mathcal{A} = \mathcal{A}^{\mathrm{pi}} \cup \mathcal{A}^{\mathrm{va}} \cup \mathcal{A}^{\mathrm{rs}} \cup \mathcal{A}^{\mathrm{rg}} \cup \mathcal{A}^{\mathrm{cs}}$ for the network elements considered in this paper, i.e., pipes, valves, resistors, regulators, and compressor stations respectively. Note that regulators are also often named control valves in the literature, e.g. see [8] or [19]. In a similar fashion we split the node set $\mathcal{V} = \mathcal{V}^{\mathrm{b}} \cup \mathcal{V}^{0}$ into boundary nodes and inner nodes respectively. Here, boundary nodes $\mathcal{V}^{\mathrm{b}}$ represent those having inflow and pressure level demand values for future time steps. We define the set of considered time steps as $\mathcal{T}_0 := \{0, \ldots, k\}$ where $\mathcal{T} := \mathcal{T}_0 \setminus \{0\}$ are the future time steps having demand conditions at boundary nodes. Associated with each time step t is a value $\tau(t)$ representing the time difference in seconds from t to the initial state time step 0.

The most important quantities we will consider to describe the gas flow are pressure and mass flow. We have pressure variables $p_{v,t}$ at each node $v \in \mathcal{V}$ and time $t \in \mathcal{T}_0$ as well as variables $q_{a,t}$ for the flow from l to r on each non-pipe arc $(l, r) = a \in \mathcal{A} \setminus \mathcal{A}^{\mathrm{pi}}$ and time $t \in \mathcal{T}_0$. For a pipe $(l, r) = a \in \mathcal{A}^{\mathrm{pi}}$, we have two

4

flow variables $q_{l,a,t}$ representing the inflow into the pipe at end node l and $q_{r,a,t}$ representing the outflow out of the pipe at end node r, similar to the model of [8] on pipes or the one of [6] on all arcs. Our final flow quantity is the inflow at a boundary node $d_{v,t}$ entering the network for each $v \in \mathcal{V}^{\mathrm{b}}$ and $t \in \mathcal{T}_0$. Although we are given flow demands for the future, we allow deviations from them, and hence have to have a variable capturing the actual inflow value.

We assume that there are bounds on all stated quantities for each point in time $t \in \mathcal{T}_0$, so upper and lower bounds $\bar{p}_{v,t}$ and $\underline{p}_{v,t}$ on the pressure at each node $v \in \mathcal{V}$, upper and lower bounds $\bar{q}_{v,t}$ and $\underline{q}_{v,t}$ on the flow on each non-pipe arc $a \in \mathcal{A} \setminus \mathcal{A}^{\mathrm{pi}}$ respectively the inflow and outflow of each pipe $a \in \mathcal{A}^{\mathrm{pi}}$, as well as upper and lower bounds $\bar{d}_{v,t}$ and $\underline{d}_{v,t}$ on the inflow at each boundary node $v \in \mathcal{V}^{\mathrm{b}}$. Note that while pressure is always positive, flow itself can be negative, as it can represent flow in the opposite direction. For example, negative inflow at a boundary node represents flow out of the network at this node.

In the following we will describe each of the elements and at the same time introduce corresponding variables and constraints to create a MIP formulation. This MIP will not be solved directly, but is the basis for the three variants used in our overall solution algorithm described in Section 4.

Note that a list of all used variables can be found in the appendix in Table A.1.

2.1 Compressor stations

Compressor stations are responsible for increasing the pressure in the network and thereby the most important in controlling the flow of gas. They are also the most complex elements, having their own substructure and a large amount of operational restrictions. Our model of the compressor station is based on the description in [10], where the elements are called *compressor groups* instead.

Structure of a compressor station A compressor station $(l,r) = a \in \mathcal{A}^{\mathrm{cs}}$ has three different modes: Bypass, closed and active mode. In bypass mode the element is bypassed and therefore allows unrestricted gas flow without changing the pressure level. For the closed mode, the element is closed and thereby blocks the gas flow, which disconnects the network between the its end nodes. Finally, in active mode the gas is compressed and pressure is increased along the direction of the flow.

When compressing, the compressor station can use a set of associated compressor units $\mathcal{U}_a$. These are the actual pressure increasing elements, each with a separate operating range. In the compressor station, these compressor units are combined in series and/or parallel to allow proper reactions to different compression requirements. The set of all allowed serial-parallel compressor unit combinations is called the set of configurations $\mathcal{C}_a$ for a compressor station $a \in \mathcal{A}^{\mathrm{cs}}$, from which exactly one active configuration has to be chosen if the compressor station is in active mode. For each of these configurations $c \in \mathcal{C}_a$, we create a polytope in the space (p_l, p_r, q) describing the feasible operating range of the compressor station using configuration c. This polytope is described as the intersection of a set of half spaces $\mathcal{H}_c = \{(w, x, y, z) \in \mathbb{R}^4\}$ encoding inequalities of the form $w \cdot p_l + x \cdot p_r + y \cdot q + z \leq 0$. The creation of the feasible operating range of the configurations of the compressor stations is described in Section 3.

Compressor station model To model the above described constraints we use a disjunctive formulation. This is the most compact formulation in terms of number of constraints and variables for which holds that its LP relaxation is equal to the convex hull of its feasible points [1][2]. For this we introduce binary "selection" variables $m_{c,a,t}^{\text{cf}}$ for each configuration $c \in \mathcal{C}_a$ of a compressor station $(l,r) = a \in \mathcal{A}^{\text{cs}}$, as well as $m_{a,t}^{\text{by}}$ and $m_{a,t}^{\text{cl}}$ for bypass and closed mode respectively. In addition we introduce corresponding sets of pressure and flow variables. The binary variables will force the pressure and flow variables of all non-selected configurations or modes to be zero and only enforce the constraints of the selected configuration or mode. The introduced pressure and flow variables are:

$$
\begin{array}{lll}
p_{a,t}^{\text{by}} \quad q_{a,t}^{\text{by}} & & \text{bypass mode variables} \\
p_{a,t}^{\text{l}-\text{cl}} \quad p_{a,t}^{\text{r}-\text{cl}} & & \text{closed mode variables} \\
p_{c,a,t}^{\text{l}-\text{cf}} \quad p_{c,a,t}^{\text{r}-\text{cf}} \quad q_{c,a,t}^{\text{cf}} \quad \forall c \in \mathcal{C}_a & & \text{configuration variables}
\end{array}
$$

Note that we need only one p value for bypass mode, since here $p_l = p_r$ holds. Also there is no q variable for the closed mode, since $q = 0$ holds in this case anyway. Furthermore, all introduced variables have bounds equal to the corresponding original pressure and flow bounds $\bar{p}_{l,t}$, $\underline{p}_{l,t}$, $\bar{p}_{r,t}$, $\underline{p}_{r,t}$, $\bar{q}_{a,t}$, $\underline{q}_{a,t}$ of the compressor station $(l,r) = a \in \mathcal{A}^{\text{cs}}$, enlarged to zero if necessary. We indicate these bounds by the variable symbol combined with an overscore respectively underscore.

We are now able to state the constraints for all $(l,r) = a \in \mathcal{A}^{\text{cs}}$ and $t \in \mathcal{T}$:

$$1 = \sum_{c \in \mathcal{C}_a} m_{c,a,t}^{\text{cf}} + m_{a,t}^{\text{by}} + m_{a,t}^{\text{cl}} \tag{1}$$

$$p_{l,t} = p_{a,t}^{\text{by}} + p_{a,t}^{\text{l}-\text{cl}} + \sum_{c \in \mathcal{C}_a} p_{c,a,t}^{\text{l}-\text{cf}} \tag{2}$$

$$p_{r,t} = p_{a,t}^{\text{by}} + p_{a,t}^{\text{r}-\text{cl}} + \sum_{c \in \mathcal{C}_a} p_{c,a,t}^{\text{r}-\text{cf}} \tag{3}$$

$$q_{a,t} = q_{a,t}^{\text{by}} + \sum_{c \in \mathcal{C}_a} q_{c,a,t}^{\text{cf}} \tag{4}$$

$$\underline{p}_{c,a,t}^{\text{l}-\text{cf}} m_{c,a,t}^{\text{cf}} \leq p_{c,a,t}^{\text{l}-\text{cf}} \leq \bar{p}_{c,a,t}^{\text{l}-\text{cf}} m_{c,a,t}^{\text{cf}} \quad \forall c \in \mathcal{C}_a \tag{5}$$

$$\underline{p}_{c,a,t}^{\text{r}-\text{cf}} m_{c,a,t}^{\text{cf}} \leq p_{c,a,t}^{\text{r}-\text{cf}} \leq \bar{p}_{c,a,t}^{\text{r}-\text{cf}} m_{c,a,t}^{\text{cf}} \quad \forall c \in \mathcal{C}_a \tag{6}$$

$$\underline{q}_{c,a,t}^{\text{cf}} m_{c,a,t}^{\text{cf}} \leq q_{c,a,t}^{\text{cf}} \leq \bar{q}_{c,a,t}^{\text{cf}} m_{c,a,t}^{\text{cf}} \quad \forall c \in \mathcal{C}_a \tag{7}$$

$$\underline{p}_{a,t}^{\text{by}} m_{a,t}^{\text{by}} \leq p_{a,t}^{\text{by}} \leq \bar{p}_{a,t}^{\text{by}} m_{a,t}^{\text{by}} \tag{8}$$

$$\underline{q}_{a,t}^{\text{by}} m_{a,t}^{\text{by}} \leq q_{a,t}^{\text{by}} \leq \bar{q}_{a,t}^{\text{by}} m_{a,t}^{\text{by}} \tag{9}$$

$$\underline{p}_{a,t}^{\text{l}-\text{cl}} m_{a,t}^{\text{cl}} \leq p_{a,t}^{\text{l}-\text{cl}} \leq \bar{p}_{a,t}^{\text{l}-\text{cl}} m_{a,t}^{\text{cl}} \tag{10}$$

$$\underline{p}_{a,t}^{\text{r}-\text{cl}} m_{a,t}^{\text{cl}} \leq p_{a,t}^{\text{r}-\text{cl}} \leq \bar{p}_{a,t}^{\text{r}-\text{cl}} m_{a,t}^{\text{cl}} \tag{11}$$

$$w \cdot p_{c,a,t}^{\text{l}-\text{cf}} + x \cdot p_{c,a,t}^{\text{r}-\text{cf}} + y \cdot q_{c,a,t}^{\text{cf}} + z m_{c,a,t}^{\text{cf}} \leq 0 \quad \forall (w,x,y,z) \in \mathcal{H}_c \quad \forall c \in \mathcal{C}_a \tag{12}$$

2.2 Pipes

Gas flow in pipelines is for operational purposes modeled as one-dimensional flow through a straight cylindrical pipe. When assuming a constant gas temperature

T, the flow can be described by the isothermal *Euler Equations*[26] consisting of the *Continuity Equation* and the *Momentum Equation*. The Continuity Equation describes the conservation of mass, i.e., guaranteeing that mass may neither be created nor destroyed. On the other hand, the Momentum Equation reflects the equality between the force acting on the gas particles and their corresponding rate of change of momentum. For a pipe $a = (l, r)$ we can state this pair of equations as

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0 \tag{13}$$

$$\frac{\partial (\rho v)}{\partial t} + \frac{\partial p}{\partial x} + \frac{\partial (\rho v^2)}{\partial x} + \frac{\lambda_a}{2D_a} |v| v \rho + g s_a \rho = 0. \tag{14}$$

Here x denotes the position in the pipe by its distance from the source node l, t the current time, ρ the density of the gas, v its velocity, D_a the diameter of the pipe, g the gravitational acceleration, and λ_a the friction factor of the pipe, which we assume to depend on pipe characteristics only, see Section 2.2.1. With $s_a \in [-1, 1]$ we denote the slope of the pipe, i.e., the quotient of the elevation increase between the pipes endpoints and the length L_a of the pipe.

In order to complete the system of equations describing the state variables p, ρ, and v, we add the *equation of state for real gases* to establish the connection between p and ρ as

$$p = \rho R_{\mathrm{s}} T z_a. \tag{15}$$

The two new quantities that arise are the compressibility factor z_a and the specific gas constant R_{s}. We assume both values to be constant parameters, which for the compressibility factor is a common assumption in the gas transport literature, see for example [26][4]. For the specific gas constant this follows from its dependence on the molar mass, which in turn is determined by the gas mixture which we assume to be constant.

In the following, we will drop the terms $\partial_t(\rho v)$ and $\partial_x(\rho v^2)$ as they contribute only little to the equation under normal operating conditions [8][26]. In addition, we reformulate the pipe flow equations in terms of the quantities we are interested in, i.e., pressure p and mass flow q, where q is defined using the cross sectional area $A_a = D_a^2 \frac{\pi}{4}$ of the cylindric pipe a as

$$q = A_a \rho v. \tag{16}$$

Then we can write (13) and (14) as

$$\frac{\partial p}{\partial t} + \frac{R_{\mathrm{s}} T z_a}{A_a} \frac{\partial q}{\partial x} = 0$$

$$\frac{\partial p}{\partial x} + \frac{\lambda_a R_{\mathrm{s}} T z_a}{2 D_a A_a^2} \frac{|q| q}{p} + \frac{g s_a}{R_{\mathrm{s}} T z_a} p = 0.$$

Since the spatial pressure change now mainly depends on the friction term including the friction factor λ_a this model variant is often referred to as the *friction dominated* model, see model (FD1) in [3] respectively model (ISO3) in [7].

For the discretization, we use the implicit box scheme introduced by [6], respectively [20]. Here, the spacial domain is the length L_a of pipe $a = (l, r)$ and the time domain is the set of time steps $\mathcal{T}_0$, which we defined in the beginning of

Section 2. Using the notation of flow into and out of a pipe, again defined in beginning of Section 2, as well as the function τ, we are able to write the discretized model for two adjacent time points t_0 and t_1 as

$$p_{l,t_1} + p_{r,t_1} - p_{l,t_0} - p_{r,t_0} + \frac{2R_sT z_a(\tau(t_1) - \tau(t_0))}{L_a A_a}(q_{r,a,t_1} - q_{l,a,t_1}) = 0 \quad (17)$$

$$p_{r,t_1} - p_{l,t_1} + \frac{\lambda_a R_s T z_a L_a}{4 D_a A_a^2}\left(\frac{|q_{l,a,t_1}|q_{l,a,t_1}}{p_{l,t_1}} + \frac{|q_{r,a,t_1}|q_{r,a,t_1}}{p_{r,t_1}}\right)$$
$$+ \frac{g s_a L_a}{2 R_s T z_a}(p_{l,t_1} + p_{r,t_1}) = 0. \quad (18)$$

As a final step, we will linearize the Momentum Equation (18) as proposed in [16] by fixing the absolute velocity $|v|$ to a predefined constant in the friction term, where the absolute velocity is according to (15) and (16) defined as $|v| = |\frac{R_s T z_a}{A_a}\frac{q}{p}| = \frac{R_s T z_a}{A_a}\frac{|q|}{p}$. We found that the relative error stated in [16] is less relevant for the overall accuracy of the model, if the pipes are contained in network stations. The reason for this is, that the elements of a network station are usually clustered in a small geographic area. Therefore the contained pipes are relatively short in comparison to the rest of the network, see also Table 1 with statistics over different network stations. Since the friction based pressure reduction depends on the pipe length, the corresponding term and therefore the also stated relative error have much less impact than usual. This linearization allows us to model the pipe flow in a MIP context. The final equations for each pipe $(l, r) = a \in \mathcal{A}^{\mathrm{pi}}$ and all adjacent time points $t_0, t_1 \in \mathcal{T}_0$ are:

$$p_{l,t_1} + p_{r,t_1} - p_{l,t_0} - p_{r,t_0} + \frac{2R_sT z_a(\tau(t_1) - \tau(t_0))}{L_a A_a}(q_{r,a,t_1} - q_{l,a,t_1}) = 0 \quad (19)$$

$$p_{r,t_1} - p_{l,t_1} + \frac{\lambda_a L_a}{4 D_a A_a}(|v_{l,a}|q_{l,a,t_1} + |v_{r,a}|q_{r,a,t_1})$$
$$+ \frac{g s_a L_a}{2 R_s T z_a}(p_{l,t_1} + p_{r,t_1}) = 0 \quad (20)$$

The constant $|v_{x,a}|$ for one of the end nodes $x \in \{l, r\}$ of pipe $a = (l, r)$ is determined based on the flow and pressure values of the given initial state, i.e.,

$$|v_{x,a}| = \frac{R_s T z_a}{A_a}\frac{|q_{x,a,0}|}{p_{x,0}}.$$

2.2.1 Friction and compressibility factor

The *Darcy–Weisbach friction factor* λ, describes the pressure drop on a pipe a caused by frictional forces and depends on the diameter D_a and integral roughness k_a of the pipe, as well as the current flow q and the dynamic viscosity η of the gas. For turbulent gas flow the most accurate description is given by the implicit Colebrook-White equation [5][10]. There exist a series of different explicit approximation formulas, typically depending on the *Reynolds Number* which describes the amount of turbulence of the flow. We use the formula of Nikuradse [24][10], which assumes infinite turbulence and makes the friction factor dependent only on the constant diameter D_a and integral roughness k_a of the pipe:

$$\lambda_a = \left(2\log_{10}\left(\frac{D_a}{k_a}\right) + 1.138\right)^{-2}.$$

For the compressibility factor z we use the approximation formula developed by Papay [27][31], which is valid up to 150 bar and thus fits our considered pressure range of 1 bar to 100 bar well. It is given as

$$z(p) = 1 - 3.52 \frac{p}{p^{\mathrm{c}}} e^{-2.26 \frac{T}{T^{\mathrm{c}}}} + 0.247 \left(\frac{p}{p^{\mathrm{c}}} \right)^2 e^{-1.878 \frac{T}{T^{\mathrm{c}}}}.$$

Apart from the pressure p and gas temperature T Papay's formula depends on the gas mixture dependent pseudo-critical pressure p^{c} and temperature T^{c}, which we assume to be given constants. The constant compressibility factor z_a per pipe $a = (l, r)$ is then determined as an average of the corresponding values derived from the initial state pressure values of the pipes end nodes, i.e., by $z_a = (z(p_{l,0}) + z(p_{r,0}))/2$.

2.3   Resistors

Resistors are artificial elements created to model points of high friction in the network, which can be caused by all sorts of special elements. Some examples of these elements are measuring equipment and complex local piping, which both are not captured by the other considered element types but need to be accounted for. The pressure drop caused by a resistor arc $(l, r) = a \in \mathcal{A}^{\mathrm{rs}}$ for time $t \in \mathcal{T}$ is defined by the *Darcy-Weisbach equation* [10]:

$$p_{l,t} - p_{r,t} = \frac{\zeta_a R_{\mathrm{s}} T z_a}{2 A_a^2} \left(\frac{|q_{a,t}| q_{a,t}}{p_{\mathrm{in},t}} \right)$$

Here the friction factor of the resistor is called the *drag factor*, and is represented by ζ_a, a parameter of the element. The compressibility factor z_a is determined in the same way as described for pipes, see Section 2.2.1. The formula is flow direction dependent, where $p_{\mathrm{in},t}$ is either $p_{l,t}$ or $p_{r,t}$ depending on $q_{a,t}$ being positive or negative (for $q_{a,t} = 0$ it holds that $p_{l,t} = p_{r,t}$).

As we did for pipes in Equation (20) we linearize the model by assuming a constant velocity $|v| = \frac{R_{\mathrm{s}} T z_a}{A_a} \frac{|q|}{p}$, which also includes the flow direction dependent pressure value. The equations for each arc $(l, r) = a \in \mathcal{A}^{\mathrm{rs}}$ and time $t \in \mathcal{T}$ then reads as

$$p_{l,t} - p_{r,t} = \frac{\zeta_a |v_a|}{2 A_a} q_{a,t}. \tag{21}$$

The constant velocity value is again calculated based on the initial element state, and is defined as an average of the two velocities using the pressure from the corresponding resistors end nodes as

$$|v_l| = \frac{R_{\mathrm{s}} T z_a}{A_a} \frac{|q_{a,0}|}{p_{l,0}}$$

$$|v_r| = \frac{R_{\mathrm{s}} T z_a}{A_a} \frac{|q_{a,0}|}{p_{r,0}}$$

$$|v_a| := \frac{|v_l| + |v_r|}{2}$$

2.4 Valves

Valves are active elements that dynamically connect or disconnect two nodes by being open or closed respectively, and thereby change the network topology. Closed valves work exactly as the closed mode of a compressor station, while open valves imply the same behavior as the corresponding stations bypass mode. The mode of a valve is captured by the binary variable $m_{a,t}^{\mathrm{op}}$. For a valve arc $(l,r) = a \in \mathcal{A}^{\mathrm{va}}$ and time $t \in \mathcal{T}$ we write

$$p_{l,t} - p_{r,t} \leq (1 - m_{a,t}^{\mathrm{op}})(\bar{p}_{l,t} - \underline{p}_{r,t}) \tag{22}$$

$$p_{l,t} - p_{r,t} \geq (1 - m_{a,t}^{\mathrm{op}})(\underline{p}_{l,t} - \bar{p}_{r,t}) \tag{23}$$

$$q_{a,t} \leq (m_{a,t}^{\mathrm{op}})\bar{q}_{a,t} \tag{24}$$

$$q_{a,t} \geq (m_{a,t}^{\mathrm{op}})\underline{q}_{a,t}. \tag{25}$$

2.5 Regulators

A regulator or control valve is a valve with variable opening degree, used to reduce the pressure along the direction of the flow. Regulators can reduce the pressure in active mode and also be bypassed or closed like a compressor station. For each mode there is a binary variable, from which exactly one is equal to 1 at any time, i.e., the regulator always has to have a unique mode

$$1 = m_{a,t}^{\mathrm{cl}} + m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{ac}} \qquad \forall a \in \mathcal{A}^{\mathrm{rg}} \quad \forall t \in \mathcal{T} \tag{26}$$

The implications of each of the modes can be modeled by the following constraints for each arc $a \in \mathcal{A}^{\mathrm{rg}}$ and all times $t \in \mathcal{T}$

$$p_{l,t} - p_{r,t} \leq +(1 - m_{a,t}^{\mathrm{by}})(\bar{p}_{l,t} - \underline{p}_{r,t}) \tag{27}$$

$$p_{l,t} - p_{r,t} \geq +(1 - m_{a,t}^{\mathrm{by}} - m_{a,t}^{\mathrm{ac}})(\underline{p}_{l,t} - \bar{p}_{r,t}) \tag{28}$$

$$q_a \leq (1 - m_{a,t}^{\mathrm{cl}})\bar{q}_{a,t} \tag{29}$$

$$q_a \geq 0. \tag{30}$$

Note that the flow is always positive, even in bypass mode. This occurs because all regulators have a flap trap which prevents flow going against the topological orientation.

2.6 Nodes

The nodes don't represent technical elements but rather establish the connections between them. The pressure coupling is realized by using the pressures at a node in the constraints of all its incident arcs. To connect the mass flow between arcs we have flow conservation constraints in each node. This means that the sum of incoming flows should match the sum of outgoing flows, resulting in the following

constraints for all $t \in \mathcal{T}$:

$$\sum_{(l,v)=a\in\mathcal{A}^{\mathrm{pi}}} q_{v,a,t} - \sum_{(v,r)=a\in\mathcal{A}^{\mathrm{pi}}} q_{v,a,t}$$

$$+ \sum_{(l,v)=a\in\mathcal{A}\backslash\mathcal{A}^{\mathrm{pi}}} q_{a,t} - \sum_{(v,r)=a\in\mathcal{A}\backslash\mathcal{A}^{\mathrm{pi}}} q_{a,t} + d_{v,t} = 0 \qquad \forall v \in \mathcal{V}^{\mathrm{b}} \qquad (31)$$

$$\sum_{(l,v)=a\in\mathcal{A}^{\mathrm{pi}}} q_{v,a,t} - \sum_{(v,r)=a\in\mathcal{A}^{\mathrm{pi}}} q_{v,a,t}$$

$$+ \sum_{(l,v)=a\in\mathcal{A}\backslash\mathcal{A}^{\mathrm{pi}}} q_{a,t} - \sum_{(v,r)=a\in\mathcal{A}\backslash\mathcal{A}^{\mathrm{pi}}} q_{a,t} = 0 \qquad \forall v \in \mathcal{V}^{0} \qquad (32)$$

2.7 Network station

In addition to the constraints imposed by the single elements used in the network, the network station itself has a set of associated constraints. The model described below is an extension of the one described in [10] for similar structures. There, network stations are called *compressor stations* and the elements we call compressor stations are called *compressor groups*, as already mentioned in Section 2.1.

Network station structure Most important are the *operation modes* $\mathcal{O}$ of the network station from which exactly one has to be selected at each time point and which determine the modes and configurations of all valves and compressor stations. Note that not all possible mode combinations of the different elements have to be valid operation modes of the network station. In addition to the operation mode a *flow direction* has to be chosen for the network station from the set of possible flow directions $\mathcal{F}$. The restrictions on each flow direction of the station prescribe the flow patterns in terms of inflow, outflow or no-flow over the boundary nodes of the station. The flow direction itself also has to fit to the selected operation mode of the network station, where the set of feasible operation mode and flow direction pairs is $\mathcal{Q}$.

Operation modes model We introduce binary variables $m_{o,t}^{\mathrm{om}}$ for each $o \in \mathcal{O}$ and $t \in \mathcal{T}_0$. They represent whether the operation mode has been selected, or respectively if the network station is in the given operation mode at this point in time. Furthermore, we define the function $M(o, a)$, which maps operation modes to the individual modes and configurations of valves and compressor stations:

$$M(o, a) := x \text{ where } x \text{ is the mode or configuration of arc } a$$
$$\text{in operation mode } o \quad \forall o \in \mathcal{O} \quad \forall a \in \mathcal{A}^{\mathrm{va}} \cup \mathcal{A}^{\mathrm{cs}}$$
$$\text{with} \quad x \in \{\mathrm{op}, \mathrm{cl}\} \text{ if } a \in \mathcal{A}^{\mathrm{va}}$$
$$x \in \{\mathrm{by}, \mathrm{cl}\} \cup \mathcal{C}_a \text{ if } a \in \mathcal{A}^{\mathrm{cs}}$$

Note that we assume that all valves and compressor stations are determined by the network station operation modes. In general however, there are valves which are not controlled by the operation modes, and whose mode is already given as a fixed decision over time and cannot be changed. We can handle these valves by

11

pre-processing, and either contracting the associated valve in the open case, or simply removing the valve in the closed case.

Using $M(o, a)$ we can then state the operation mode related constraints for all $t \in \mathcal{T}$:

$$\sum_{o \in \mathcal{O}} m_{o,t}^{\mathrm{om}} = 1 \tag{33}$$

$$m_{a,t}^{\mathrm{op}} = \sum_{o \in \mathcal{O}: M(o,a)=\mathrm{op}} m_{o,t}^{\mathrm{om}} \quad \forall a \in \mathcal{A}^{\mathrm{va}} \tag{34}$$

$$m_{a,t}^{\mathrm{by}} = \sum_{o \in \mathcal{O}: M(o,a)=\mathrm{by}} m_{o,t}^{\mathrm{om}} \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \tag{35}$$

$$m_{a,t}^{\mathrm{cl}} = \sum_{o \in \mathcal{O}: M(o,a)=\mathrm{cl}} m_{o,t}^{\mathrm{om}} \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \tag{36}$$

$$m_{c,a,t}^{\mathrm{cf}} = \sum_{o \in \mathcal{O}: M(o,a)=c} m_{o,t}^{\mathrm{om}} \quad \forall c \in \mathcal{C}_a \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \tag{37}$$

Note that either Equation (35), Equation (36) or one of the constraints of Equation (37) for one of the configurations $c \in \mathcal{C}_a$ can be omitted, because it follows from the remaining constraints combined with Equation (1).

Operation mode unavailability   Certain operation modes are not available at specific points in time. The basis for this is the non-availability of compressor units over time, which is part of the model input data.

As explained in Section 2.1, a configuration $c \in \mathcal{C}_a$ of some compressor station $a \in \mathcal{A}^{\mathrm{cs}}$ represents the serial and/or parallel combination of a subset of the compressor stations compressor units. Hence, the unavailability of a certain compressor unit at time t results in the unavailability of all configurations which use this unit. On the next level, each network station operation mode defines the mode and (for the active mode) the configuration of each compressor station in the network station. Hence, all network station operation modes using a configuration for a compressor station which is unavailable for time t will be unavailable for t, too. To implement this in the model, we just fix the variables $m_{o,t}^{\mathrm{om}}$ for the corresponding operation mode o and time points t to zero, i.e., remove them from the model.

The unavailability of a compressor unit may not be aligned with the set of discrete time points $\mathcal{T}_0$, i.e., the unavailability period may start or stop in between two adjacent time points. To be able to tell which of the two points is then effected by this, we have to establish an interpretation for the operation mode of a network station between two time steps. Therefore, we define that if a network station has the operation mode A at the discrete time point t, then we also assume the station to have operation mode A in the following time interval up to the next discrete time point $t + 1$. From this definition it follows that if a network station operation mode is unavailable for some $k \in (t, t+1)$ with $t, t+1 \in \mathcal{T}_0$, then the operation mode is unavailable for time t but potentially available for time $t + 1$.

Operation mode transition times   If the operation mode of a network station is changed from mode A to some other mode B, the transition takes a given amount of time $\theta(A, B)$. A transition time is given for every possible combination

of operation modes and is part of the input data. While in transition between the two modes, the network station acts as follows: Assume the transition starts at time t_0, then for $t \in \left[t_0, t_0 + \frac{\theta(A,B)}{2} \right)$ the station uses mode A while for $t \in \left[t_0 + \frac{\theta(A,B))}{2}, t_0 + \theta(A,B) \right)$ the station uses mode B. In other words, the station stays in mode A until it reaches the middle of the transition period and then changes to mode B. While being in transition, the whole transition period is blocked for other changes, i.e., two transition periods cannot overlap. Since we are only able to change network station operation modes at discrete time points, we assume that for each transition the middle point is in $\mathcal{T}_0$. This is also in line with our interpretation of network station operation modes in between discrete time points.

In Figure 2 we see an example of an operation mode sequence with corresponding transition times. The time point t_X represents the first $t \in \mathcal{T}_0$ in which operation mode X is active, which according to the interpretation in the previous paragraph is also the first discrete time point t in which X is active. In this example, there would be a conflict in the transition times $\theta(C,D)$ from mode C to mode D and $\theta(D,E)$ from mode D to mode E, since the two time periods overlap. Note that this is true, although for each single transition period the network station has the correct operation mode for each point in time, i.e. first mode in the first half of the period and the second mode in the second half.
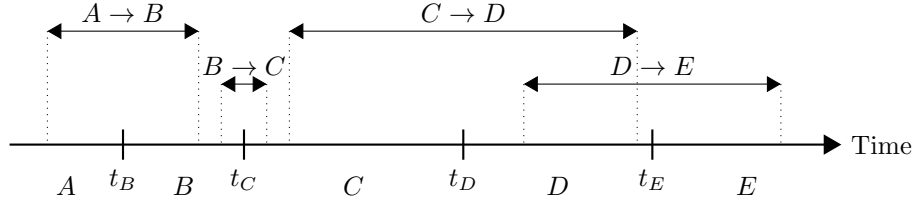

Figure 2: Transition time example with 5 operation modes and 4 transitions, from which two are in conflict. The time point t_X represent the first discrete time point in $\mathcal{T}_0$ in which mode X is active.

We will *not* cover the transition time restrictions in the MIP model, but will make sure our solutions respect them in a different way, see Section 4.2. For this we only need a way to check for a given sequence of operation modes if all the transition times are valid, i.e., the corresponding periods do not overlap, which we do as follows: For each network station operation mode in the sequence, we check if the time in which the operation mode is active is at least as big as the sum of adjacent transition time components, i.e., the sum of half the time of the transition into that mode and half of the time of the transition out of that mode. In the given example, we would have to check for mode C if $\tau(t_D) - \tau(t_C) \geq \frac{\theta(B,C)}{2} + \frac{\theta(C,D)}{2}$ holds, where $\tau(t)$ represents the time difference of a time step t from the initial state time. If the above holds for all operation modes in the sequence, then the transition periods can never overlap since they are centered around the time points in which the mode change happens.

Note, that for the last mode in the sequence, we do not need to do any checks at all. We do so as we cannot determine how long into the future the operation mode will be active for. Hence, we assume it is active long enough to comply

with the given transition time, allowing the possibility to apply operation mode changes up to the very last time step. For the first mode, a similar assumption would be too optimistic, since the desired operation mode change would have then already been triggered before the start our time horizon. However, there is no known transition into this first mode, so we need only check if half the transition time into the next mode fits into the first modes active period.

Flow directions model   Similar to the way we handle operations modes, we introduce binary variables $m_{f,t}^{\text{fd}}$ representing the selection of flow direction $f \in \mathcal{F}$ at time $t \in \mathcal{T}_0$. To be able to state the connection between the chosen flow direction and the actual boundary node inflow pattern, we represent each flow direction f as a tuple of the set of boundary nodes having inflow into the station f^+ and the set of boundary nodes having outflow out of the station f^-. Hence, using the power set $\mathcal{P}$, a flow direction f is defined as

$$(f^+, f^-) = f \in \mathcal{F} \subseteq \mathcal{P}(\mathcal{V}^{\text{b}}) \times \mathcal{P}(\mathcal{V}^{\text{b}}) \text{ where } f^+ \cap f^- = \emptyset$$

Note that if $v \notin f^+$ and $v \notin f^-$ for flow direction (f^+, f^-), then the inflow of node v is zero.

Using the inflow variable $d_{v,t}$ for boundary node v and time t we can define the flow direction constraints for each $t \in \mathcal{T}$ as:

$$\sum_{f \in \mathcal{F}} m_{f,t}^{\text{fd}} = 1 \tag{38}$$

$$m_{o,t}^{\text{om}} \leq \sum_{(o,f) \in \mathcal{Q}} m_{f,t}^{\text{fd}} \quad \forall o \in \mathcal{O} \tag{39}$$

$$d_{v,t} \geq (1 - \sum_{f=(f^+,f^-) \in \mathcal{F}: v \notin f^-} m_{f,t}^{\text{fd}}) \underline{d}_{v,t} \quad \forall v \in \mathcal{V}^{\text{b}} \tag{40}$$

$$d_{v,t} \leq (1 - \sum_{f=(f^+,f^-) \in \mathcal{F}: v \notin f^+} m_{f,t}^{\text{fd}}) \bar{d}_{v,t} \quad \forall v \in \mathcal{V}^{\text{b}} \tag{41}$$

Flow direction exit pressures   Apart from the consequences that the flow direction choice has on the corresponding boundary node inflows, it may also influence the upper pressure bounds on some boundary nodes. Each node $v \in \mathcal{V}^{\text{b}-\text{ex}} \subseteq \mathcal{V}^{\text{b}}$ is given an upper pressure bound $\bar{p}_v^{\text{exit}}$, which is only active if the node is in the outflow set of the currently active flow direction, i.e., serving as exit of the network station. The corresponding constraint is the following for each time $t \in \mathcal{T}$

$$p_{v,t} \leq \bar{p}_v^{\text{exit}} + (1 - \sum_{f=(f^+,f^-) \in \mathcal{F}: v \in f^-} m_{f,t}^{\text{fd}})(\bar{p}_{v,t} - \bar{p}_v^{\text{exit}}) \quad \forall v \in \mathcal{V}^{\text{b}-\text{ex}} \tag{42}$$

Flow direction conditions   As the final constraints concerning flow directions, there exist in some network stations a special set of conditions $\mathcal{W}$, which concern the amount of flow over sets of boundary nodes. These conditions must be met for the flow direction which they are associated with to be active. Each condition $w = (f, \mathcal{V}^{w_1}, \mathcal{V}^{w_2}) \in \mathcal{W}$ states that the flow over a set of boundary nodes $\mathcal{V}^{w_1}$ has to be smaller than the flow over a second set of boundary nodes $\mathcal{V}^{w_2}$ if f is

selected. Note, that the flow over a set of boundary nodes $\mathcal{V}^w$ for time t is defined as $\sum_{v \in \mathcal{V}^w} |d_{v,t}|$, which is potentially a non-linear expression due to the absolute value. However, each boundary node set $\mathcal{V}^w$ which is part of some $w \in \mathcal{W}$ is known to be either a subset of f^+ or a subset of f^- of the corresponding flow direction $f = (f^+, f^-)$. For this reason, we always know the sign of the flow over $\mathcal{V}^w$ in advance and hence using the following function definition for easier notation

$$\text{sgn} : \mathcal{F} \times \mathcal{V}^{\text{b}} \to \{-1, 1\}, \ ((f^+, f^-), v) \to \begin{cases} 1 & \text{if } v \in f^+ \\ -1 & \text{if } v \in f^- \end{cases}$$

we can write the constraints for each $t \in \mathcal{T}$ as

$$\sum_{v \in \mathcal{V}^{w_1}} \text{sgn}(f, v) d_{v,t} - \sum_{v \in \mathcal{V}^{w_2}} \text{sgn}(f, v) d_{v,t}$$

$$\leq (1 - m^{\text{fd}}_{f,t}) C_1 \qquad \forall (f, \mathcal{V}^{w_1}, \mathcal{V}^{w_2}) \in \mathcal{W}. \qquad (43)$$

Here C_1 denotes a big-M constant, which can be set as follows:

$$C_1 = \sum_{v \in \mathcal{V}^{w_1} : v \in f^+} \max(0, \bar{d}_{v,t}) - \sum_{v \in \mathcal{V}^{w_1} : v \in f^-} \min(0, \underline{d}_{v,t})$$

$$- \left(\sum_{v \in \mathcal{V}^{w_2} : v \in f^+} \max(0, \underline{d}_{v,t}) - \sum_{v \in \mathcal{V}^{w_2} : v \in f^-} \min(0, \bar{d}_{v,t}) \right)$$

2.8 Scenario and initial state

For the future we are given scenario values for the boundaries of the station in terms of pressure and inflow. While we are given one pressure value $\hat{p}_{v,t}$ per boundary node $v \in \mathcal{V}^{\text{b}}$ for each future time point $t \in \mathcal{T}$, the flow demands are only given for sets of boundary nodes, which are called the *fence groups* of the network station forming the set $\mathcal{G}$. For each set $g \in \mathcal{G}$, which can also consist of only a single boundary node, and each future time point $t \in \mathcal{T}$, the sum of inflows should be equal to the given demand value $\hat{d}_{g,t}$.

We do not require strict obedience of the given demand values $\hat{p}_{v,t}$ and $\hat{d}_{g,t}$, but instead allow deviations from them, which will be punished in the objective function. These deviations are captured in the *slack variables* σ. For pressure we have $\sigma^{p+}_{v,t}$ and $\sigma^{p-}_{v,t}$ which capture the positive and respectively negative difference between the pressure values of boundary node v at future time t and the given demand $\hat{p}_{v,t}$. We then have the slack variables $\sigma^{d+}_{v,t}$ and $\sigma^{d-}_{v,t}$ associated with the inflow, which capture the positive and respectively negative contribution to the difference between the inflow demand $\hat{d}_{g,t}$ of fence group $g \in \mathcal{G}$ at each future time step t and the sum of the corresponding inflow variables of each boundary node v in the fence group g.

The described relations can be modeled for each future time step $t \in \mathcal{T}$ as:

$$\hat{p}_{v,t} = p_{v,t} - \sigma^{p+}_{v,t} + \sigma^{p-}_{v,t} \quad \forall v \in \mathcal{V}^{\text{b}} \qquad (44)$$

$$\hat{d}_{g,t} = \sum_{v \in g} \left(d_{v,t} - \sigma^{d+}_{v,t} + \sigma^{d-}_{v,t} \right) \quad \forall g \in \mathcal{G} \qquad (45)$$

Note that by using inflow slacks, it is possible to choose a flow direction for the station that does not fit the inflow demand values $\hat{d}_{g,t}$ in terms of the constraints (40)-(41).

The second set of prescribed values are those of the initial state. A complete list of given values are as follows: The initial pressures $p_{v,0}$ for each node $v \in \mathcal{V}$, the in- and outflow values $q_{l,a,0}$ and $q_{r,a,0}$ for each pipe $(l,r) = a \in \mathcal{A}^{\mathrm{pi}}$, the flow values $q_{a,0}$ for each non-pipe arc $a \in \mathcal{A} \setminus \mathcal{A}^{\mathrm{pi}}$, the operation mode of the network station fixing $m_{o,0}^{\mathrm{om}}$ for each $o \in \mathcal{O}$, the thereby determined values of the variables $m_{a,0}^{\mathrm{op}}$, $m_{a,0}^{\mathrm{by}}$, $m_{a,0}^{\mathrm{cl}}$, $m_{a,0}^{\mathrm{ac}}$ for each corresponding valve and compressor station arc $a \in \mathcal{A}^{\mathrm{va}} \cup \mathcal{A}^{\mathrm{cs}}$, the values of $m_{c,a,0}^{\mathrm{cf}}$ for each compressor station $a \in \mathcal{A}^{\mathrm{cs}}$ and corresponding configuration $c \in \mathcal{C}_a$, and finally the modes $m_{a,0}^{\mathrm{by}}$, $m_{a,0}^{\mathrm{cl}}$, $m_{a,0}^{\mathrm{ac}}$ of all regulators $a \in \mathcal{A}^{\mathrm{rg}}$. All these variables are actually parameters of the model and fixed to the corresponding value.

2.9 Objective

As already describe above, the objective function should punish deviation from the given future scenario, while simultaneously favoring those solutions with a stable control of the single elements. While the first part can easily be described by using the slack variables introduced in Section 2.8, we still need to define a measure for the stability. To do so, we first quantify the discrete changes of the control in binary variables, i.e., the change of the network station into a new operation mode at time t in variable δ_t^{om} as well as the change into a new mode of regulator a at time t in variable $\delta_{a,t}^{\mathrm{rg}}$. Furthermore, we capture the start of compressor unit u at time t in the variable $\delta_{u,t}^{\mathrm{us}}$, since starting a compressor is a very time and energy intensive action and should therefore be avoided if possible. The mode changes of valves and compressor stations are not tracked separately, as the start-up of compressor stations is already penalized and the change of valves only happens for multiple elements at once in the context of operation mode changes, see the Equation (34). This described variable behavior is realized by the following constraints for each future time step $t \in \mathcal{T}$, where we denote the set of configurations of the containing compressor station $a \in \mathcal{A}^{\mathrm{cs}}$ which use compressor unit $u \in \mathcal{U}_a$ by $\mathcal{C}_u a$.

$$\delta_t^{\mathrm{om}} \geq m_{o,t}^{\mathrm{om}} - m_{o,t-1}^{\mathrm{om}} \quad \forall o \in \mathcal{O} \tag{46}$$

$$\delta_t^{\mathrm{om}} \leq 2 - m_{o,t}^{\mathrm{om}} - m_{o,t-1}^{\mathrm{om}} \quad \forall o \in \mathcal{O} \tag{47}$$

$$\delta_{a,t}^{\mathrm{rg}} \geq m_{a,t}^{\mathrm{cl}} - m_{a,t-1}^{\mathrm{cl}} \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \tag{48}$$

$$\delta_{a,t}^{\mathrm{rg}} \leq 2 - m_{a,t}^{\mathrm{cl}} - m_{a,t-1}^{\mathrm{cl}} \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \tag{49}$$

$$\delta_{a,t}^{\mathrm{rg}} \geq m_{a,t}^{\mathrm{by}} - m_{a,t-1}^{\mathrm{by}} \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \tag{50}$$

$$\delta_{a,t}^{\mathrm{rg}} \leq 2 - m_{a,t}^{\mathrm{by}} - m_{a,t-1}^{\mathrm{by}} \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \tag{51}$$

$$\delta_{a,t}^{\mathrm{rg}} \geq m_{a,t}^{\mathrm{ac}} - m_{a,t-1}^{\mathrm{ac}} \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \tag{52}$$

$$\delta_{a,t}^{\mathrm{rg}} \leq 2 - m_{a,t}^{\mathrm{ac}} - m_{a,t-1}^{\mathrm{ac}} \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \tag{53}$$

$$\delta_{u,t}^{\mathrm{us}} \geq \sum_{c \in \mathcal{C}_u a} m_{c,a,t}^{\mathrm{cf}} - \sum_{c \in \mathcal{C}_u a} m_{c,a,t-1}^{\mathrm{cf}} \quad \forall u \in \mathcal{U}_a \forall a \in \mathcal{A}^{\mathrm{cs}} \tag{54}$$

In order to obtain a smooth operation for network situations without discrete mode switching, we add variables tracking the change of the operation point of

16

single elements, i.e., their corresponding changes in flow, incoming pressure and outgoing pressure. We do this for all elements with an actual operation point, i.e., regulators and compressor stations in active mode, while ignoring times in which the network station operation mode or regulator mode has just been changed. The variables $\delta_{a,t}^{\mathrm{rg-pl}}$, $\delta_{a,t}^{\mathrm{rg-pr}}$, $\delta_{a,t}^{\mathrm{rg-q}}$ representing changes of the incoming pressure, outgoing pressure and flow of an active regulator a respectively $\delta_{a,t}^{\mathrm{cs-pl}}$, $\delta_{a,t}^{\mathrm{cs-pr}}$, $\delta_{a,t}^{\mathrm{cs-q}}$ representing the corresponding value changes of an active compressor station a can be established using the following constraints for each $(l,r) = a \in \mathcal{A}^{\mathrm{rg}}$ and each $t \in \mathcal{T}$

$$p_{l,t} - p_{l,t-1} \leq \delta_{a,t}^{\mathrm{rg-pl}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{a,t}^{\mathrm{rg}})(\bar{p}_{l,t} - \underline{p}_{l,t-1}) \tag{55}$$

$$p_{l,t-1} - p_{l,t} \leq \delta_{a,t}^{\mathrm{rg-pl}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{a,t}^{\mathrm{rg}})(\bar{p}_{l,t-1} - \underline{p}_{l,t}) \tag{56}$$

$$p_{r,t} - p_{r,t-1} \leq \delta_{a,t}^{\mathrm{rg-pr}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{a,t}^{\mathrm{rg}})(\bar{p}_{r,t} - \underline{p}_{r,t-1}) \tag{57}$$

$$p_{r,t-1} - p_{r,t} \leq \delta_{a,t}^{\mathrm{rg-pr}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{a,t}^{\mathrm{rg}})(\bar{p}_{r,t-1} - \underline{p}_{r,t}) \tag{58}$$

$$q_{a,t} - q_{a,t-1} \leq \delta_{a,t}^{\mathrm{rg-q}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{a,t}^{\mathrm{rg}})(\bar{q}_{a,t} - \underline{q}_{a,t-1}) \tag{59}$$

$$q_{a,t-1} - q_{a,t} \leq \delta_{a,t}^{\mathrm{rg-q}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{a,t}^{\mathrm{rg}})(\bar{q}_{a,t-1} - \underline{q}_{a,t}) \tag{60}$$

respectively for each $(l,r) = a \in \mathcal{A}^{\mathrm{cs}}$ and each $t \in \mathcal{T}$

$$p_{l,t} - p_{l,t-1} \leq \delta_{a,t}^{\mathrm{cs-pl}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{t}^{\mathrm{om}})(\bar{p}_{l,t} - \underline{p}_{l,t-1}) \tag{61}$$

$$p_{l,t-1} - p_{l,t} \leq \delta_{a,t}^{\mathrm{cs-pl}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{t}^{\mathrm{om}})(\bar{p}_{l,t-1} - \underline{p}_{l,t}) \tag{62}$$

$$p_{r,t} - p_{r,t-1} \leq \delta_{a,t}^{\mathrm{cs-pr}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{t}^{\mathrm{om}})(\bar{p}_{r,t} - \underline{p}_{r,t-1}) \tag{63}$$

$$p_{r,t-1} - p_{r,t} \leq \delta_{a,t}^{\mathrm{cs-pr}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{t}^{\mathrm{om}})(\bar{p}_{r,t-1} - \underline{p}_{r,t}) \tag{64}$$

$$q_{a,t} - q_{a,t-1} \leq \delta_{a,t}^{\mathrm{cs-q}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{t}^{\mathrm{om}})(\bar{q}_{a,t} - \underline{q}_{a,t-1}) \tag{65}$$

$$q_{a,t-1} - q_{a,t} \leq \delta_{a,t}^{\mathrm{cs-q}} + (m_{a,t}^{\mathrm{by}} + m_{a,t}^{\mathrm{cl}} + \delta_{t}^{\mathrm{om}})(\bar{q}_{a,t-1} - \underline{q}_{a,t}). \tag{66}$$

Note that we needed to define the upper bound constraints for the discrete change variables δ_{t}^{om} and $\delta_{a,t}^{\mathrm{rg}}$ to allow them to be equal to one only if there really is a discrete change. Otherwise it might have been possible to set the change variable to 1 although there is no actual discrete change and thereby avoid high costs imposed by the continuous change variables, which is not a desired behavior.

Finally, we are able to state our objective function, which minimizes the weighted sum of the change variables and the slack variables defined in Section 2.8

as

$$\text{min objective} :=$$

$$\sum_{t \in \mathcal{T}} \Bigg(\big(\tau(t) - \tau(t-1)\big) \sum_{v \in \mathcal{V}^{\mathrm{b}}} \big(w^{\sigma-\mathrm{p}} \cdot (\sigma_{v,t}^{p+} + \sigma_{v,t}^{p-}) + w^{\sigma-\mathrm{d}} \cdot (\sigma_{v,t}^{d+} + \sigma_{v,t}^{d-}) \big)$$

$$+ w^{\mathrm{om}} \cdot \delta_t^{\mathrm{om}}$$

$$+ \sum_{a \in \mathcal{A}^{\mathrm{rg}}} w^{\mathrm{rg}} \cdot \delta_{a,t}^{\mathrm{rg}} \tag{67}$$

$$+ \sum_{u \in \mathcal{U}_a, a \in \mathcal{A}^{\mathrm{cs}}} w^{\mathrm{us}} \cdot \delta_{u,t}^{\mathrm{us}}$$

$$+ \sum_{a \in \mathcal{A}^{\mathrm{rg}}} \big(w^{\mathrm{rg-pl}} \cdot \delta_{a,t}^{\mathrm{rg-pl}} + w^{\mathrm{rg-pr}} \cdot \delta_{a,t}^{\mathrm{rg-pr}} + w^{\mathrm{rg-q}} \cdot \delta_{a,t}^{\mathrm{rg-q}} \big)$$

$$+ \sum_{a \in \mathcal{A}^{\mathrm{cs}}} \big(w^{\mathrm{cs-pl}} \cdot \delta_{a,t}^{\mathrm{cs-pl}} + w^{\mathrm{cs-pr}} \cdot \delta_{a,t}^{\mathrm{cs-pr}} + w^{\mathrm{cs-q}} \cdot \delta_{a,t}^{\mathrm{cs-q}} \big) \Bigg),$$

where the w^* parameters denote the corresponding positive weights given to the single quantities. Note that the weights for pressure and flow slack variables are additionally multiplied by the length of the corresponding time interval. We do this, since the slacks track deviation amounts over time and we therefore need to take the time interval length into account for their objective function value coefficients.

2.10 Final model

Putting everything together, we can formulate our problem in the following transient gas flow model $\mathscr{P}$:

$$
\begin{aligned}
&\min && (67) \\
&\text{s.t.} \quad \forall t \in \mathcal{T} && (1)-(4),(8)-(11),(35)-(36),(61)-(66) \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \\
& && (5)-(7),(37) \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \quad \forall c \in \mathcal{C}_a \\
& && (12) \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \quad \forall c \in \mathcal{C}_a \quad \forall (w,x,y,z) \in \mathcal{H}_c \\
& && (19)-(20) \quad \forall a \in \mathcal{A}^{\mathrm{pi}} \\
& && (21) \quad \forall a \in \mathcal{A}^{\mathrm{rs}} \\
& && (22)-(25),(34) \quad \forall a \in \mathcal{A}^{\mathrm{va}} \\
\mathscr{P} && && (26)-(30),(48)-(53),(55)-(60) \quad \forall a \in \mathcal{A}^{\mathrm{rg}} \\
& && (31),(40)-(41),(44) \quad \forall v \in \mathcal{V}^{\mathrm{b}} \\
& && (32) \quad \forall v \in \mathcal{V}^0 \\
& && (33),(38) \\
& && (39),(46)-(47) \quad \forall o \in \mathcal{O} \\
& && (42) \quad \forall v \in \mathcal{V}^{\mathrm{b-ex}} \\
& && (43) \quad \forall w \in \mathcal{W} \\
& && (45) \quad \forall g \in \mathcal{G} \\
& && (54) \quad \forall a \in \mathcal{A}^{\mathrm{cs}} \quad u \in \mathcal{U}_a
\end{aligned}
$$

Note that we apply the constraints only starting from time step 1 explicitly excluding the initial time step 0. We do this since the initial pressure and flow values as well as the initial modes described in Section 2.8 are not guaranteed to fit our model, and simply serve as a starting point for the calculations.

3 Feasible operating range of compressor station configurations

As already explained in Section 2.1 each compressor station arc $(l, r) = a \in \mathcal{A}^{\mathrm{cs}}$ has an inherent substructure. It represents a set of compressor units $\mathcal{U}_a$, which are the actual compressing elements and can be combined in a serial and/or parallel fashion, to either allow for a higher compression ratio, a higher flow rate or a mixture of both. The set of all feasible serial-parallel combinations is called the set of configurations $\mathcal{C}_a$ of a compressor station. For each of these configurations, we will describe in this section how to obtain their polytope description given as intersection of the set of half spaces $\mathcal{H}_c$, which is used in the model as described in Section 2.1. The polytope of a configuration is created based on the polytopes of the compressor units, which are created from combining the corresponding feasible operation ranges with a maximum power restriction. Note that we drop the time index in this section for the ease of notation.

3.1 Feasible operating range for a single compressor unit

A compressor unit is a combination of a single compressor machine (or just a compressor), which increases the gas pressure in flow direction, and a corresponding drive, providing the power needed to run the compressor. For each compressor machine we are given a feasible operating range as polytope in the space $(\frac{p_r}{p_l}, Q)$, where the volumetric flow rate Q is given as

$$Q = q/\rho_l, \quad \text{with } \rho_l = \frac{p_l}{R_\mathrm{s} T z_l}.$$

Note, that since we explicitly consider density and pressure at the incoming node l of the compressor, we use the compressibility factor $z_l := z(p_{l,0})$ instead of z_a, here as well as in the rest of this Section.

Usually, the feasible operating range, sometimes also called "characteristic diagram" or "performance curve", is given as area in the dimensions (H_ad, Q) restricted by a set of possibly concave quadratic curves, see e.g. [10][25]. The quantity H_ad denotes the *specific change in adiabatic enthalpy* and is defined as

$$H_\mathrm{ad} = R_\mathrm{s} T z_l \frac{\kappa}{\kappa - 1} \left[\left(\frac{p_r}{p_l} \right)^{\frac{\kappa - 1}{\kappa}} - 1 \right], \tag{68}$$

using for the *isentropic exponent* κ the constant value 1.296, as stated in [10]. The transformation of such a feasible operating range using H_ad into our format is easily doable, since there is a unique transformation from H_ad to $\frac{p_r}{p_l}$ obtained by simply rearranging (68). The diagram then just has to be linearized by approximation or relaxation to obtain the polytope description in the desired space.

In addition to the feasible operating range polytope, each compressor machine is given an upper bound on the absolute pressure increase $\bar{\Delta}_p \geq p_r - p_l$ and an upper bound on the maximum power to use $\bar{P}$ based on the power of the compressor drive. The power needed for compression depends on the above defined H_{ad} as well as the mass flow and is given as

$$P = \frac{qH_{\mathrm{ad}}}{\eta_{\mathrm{ad}}} = \frac{q}{\eta_{\mathrm{ad}}} R_{\mathrm{s}} T z_l \frac{\kappa}{\kappa - 1} \left[\left(\frac{p_r}{p_l} \right)^{\frac{\kappa - 1}{\kappa}} - 1 \right]. \tag{69}$$

Here η_{ad} denotes the *adiabatic efficiency* of the compression, which in theory depends on the actual point of operation in the feasible region but is here assumed to be a given constant per compressor unit. An example of a feasible operating range of a compressor unit is given in Figure 3a, where different levels of the maximum pressure difference bound and the maximum power bound are given based on different values for the incoming pressure p_l.

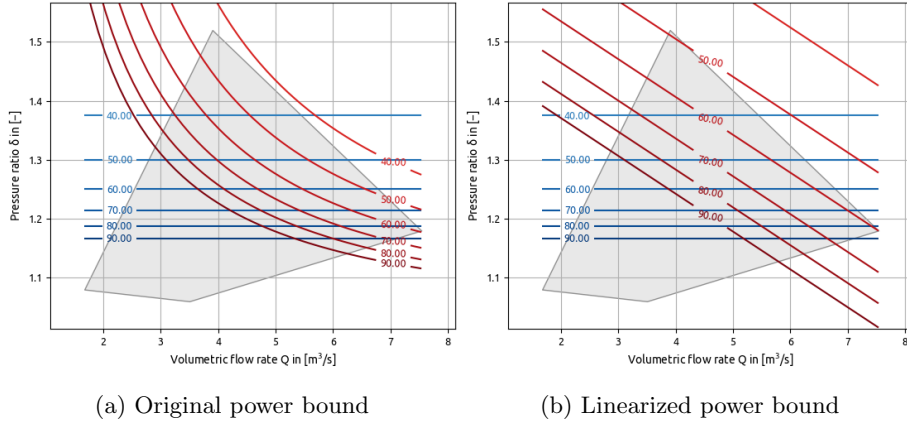(a) Original power bound   (b) Linearized power bound

Figure 3: The feasible operating range of a compressor unit. The grey region shows the operating range given as a polytope. For different values of the incoming pressure p_l the blue lines represent the upper bound on the absolute pressure increase $\bar{\Delta}_p$ and the red lines illustrate the power bound $\bar{P}$. While the left picture shows the original non-linear non-convex power bound, the right pictures shows the linearized version, see Section 3.2
.

Ignoring the power bound for a moment, we will now lift the feasible operating range into the (p_l, p_r, q) space we are interested in. Therefore, we first transform each of the faces $a_0 + a_1 Q + a_2 \frac{p_r}{p_l} \leq 0$ of the original polytope into constraints $\tilde{a}_0 p_l + \tilde{a}_1 p_r + \tilde{a}_2 q \leq 0$ of the higher dimensional space using the equation of state for real gases (15):

$$a_0 + a_1 Q + a_2 \frac{p_r}{p_l} \leq 0$$
$$\Leftrightarrow \quad a_0 + a_1 \frac{q}{\rho_L} + a_2 \frac{p_r}{p_l} \leq 0$$
$$\Leftrightarrow \quad a_0 + a_1 \frac{q R_{\mathrm{s}} T z_l}{p_l} + a_2 \frac{p_r}{p_l} \leq 0$$
$$\Leftrightarrow \quad a_0 p_l + a_1 R_{\mathrm{s}} T z_l q + a_2 p_r \leq 0$$
$$\Leftrightarrow \quad \tilde{a}_0 p_l + \tilde{a}_1 p_r + \tilde{a}_2 q \leq 0.$$

20

To bound the polyhedron described by the new constraints, we add the restriction of the absolute pressure difference as well as two pressure bounds of the end nodes of the compressor station arc $(l, r) = a \in \mathcal{A}^{cs}$ containing this machine

$$p_r - p_l \leq \bar{\Delta}_p$$
$$p_l \geq \underline{p}_l$$
$$p_r \leq \bar{p}_r.$$

A picture of the three dimensional polytope resulting from the feasible operating range of Figure 3 can be seen in Figure 4a.

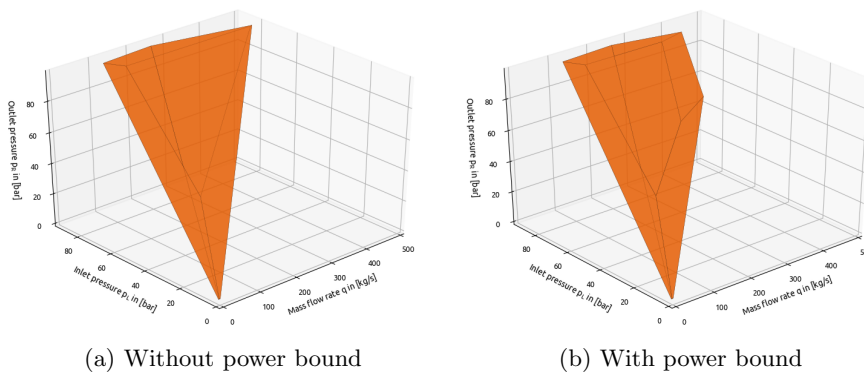(a) Without power bound

(b) With power bound

Figure 4: The feasible operating range of a compressor unit in the space (p_l, p_r, q), computed from the two dimensional operating range shown in Figure 3. While the left picture shows the lifted polytope based on the original feasible operating range, the maximum absolute pressure difference and the end nodes pressure bounds, the right pictures also includes the linearized power bound, see Section 3.2

.

3.2 Power bound linearization

Until now, we have ignored the maximum power constraint in the three dimensional feasible operation range polytope, that restricts the value of the available power P for compression. Figure 3a shows the constraint $P \leq \bar{P}$ cuts into the original two dimensional feasible operating range in a non-linear and non-convex fashion. The same holds for the feasible operating range representation in (p_l, p_r, q). In the following, we are going to derive a linear approximation to this constraint that can then be added to the operating range polytope.

Therefore, we generate a set of N random sample points from within the three dimensional operating range polytope, which we represent by the vectors $\mathbf{p_l}, \mathbf{p_r}, \mathbf{q} \in \mathbb{R}^N$. We used a rejection-free sampling method, which is based on the sampling from a 3D tetrahedron, see [30]. The idea behind this method is to sample uniformly from a through the tetrahedron generated parallelepiped and then project the sample back to the tetrahedron. To extend this method from tetrahedra to general polytopes, we first compute a triangulation of the polytope. We compute the volumes of the tetrahedra and use their relative volume as a probability function. This allows us to first randomly select a tetrahedron and

then sample from it using the in [30] described method. Our approach has high preprocessing costs of computing the triangulation tetrahedra and their volume but then it is very cheap to generate samples. On the contrary for rejection based sampling there is no lower bound for how many samples we need to generate for a single sample point, since the polytope can be arbitrarily small in comparison to the space we sample from (typically the enclosing axis-parallel cube). For the operating range polytopes we observed a significant speed-up while generating 50000 samples.

For each point, we then determine the corresponding compressor power using Equation (69) and store these values again in a vector $\mathbf{P} \in \mathbb{R}^N$. The goal is now to obtain a linear approximation of the power function, i.e., an approximation of the form

$$\mathbf{P} \approx a_0 + a_1\mathbf{p_l} + a_2\mathbf{p_r} + a_3\mathbf{q}.$$

We achieve this by applying an ordinary least-squares method in order to determine the coefficients of the linear function as

$$\min_{a_0,a_1,a_2,a_3} \| \mathbf{P} - (a_0 + a_1\mathbf{p_l} + a_2\mathbf{p_r} + a_3\mathbf{q}) \|^2 .$$

Finally, we can formulate the new linearized power bound constraint based on the obtained solution values $(\bar{a}_0, \bar{a}_1, \bar{a}_2, \bar{a}_3)$ as

$$\bar{a}_0 + \bar{a}_1 p_l + \bar{a}_2 p_r + \bar{a}_3 q \leq \bar{P},$$

which is then added to the three dimensional polytope to create the final three description of the feasible operating range of a compressor unit. An example the final polytope is illustrated in Figure 4b, while the linearized power bound projected to the original two dimensional operating range can be seen in Figure 3b.

3.3 Feasible operating range for a compressor station configuration

As the final step, we will create the polytope description for each configuration by combining the polytopes of the used compressor units. The procedure was originally described in [18], while we exactly follow the steps of the variant described in [32].

Each configuration c is given as a serial sequence $s_1, \ldots, s_{n_c}$ of parallel compressor machine arrangements, where combining compressors in series allows for higher output pressures by multi-step compression while parallel compression increases the throughput in terms of flow. We call such a parallel machine arrangement *stage* and denote by $\mathcal{U}_s$ the set of compressor units combined in stage $s \in \{s_1, \ldots, s_{n_c}\}$.

We will now start with the definition of the feasible operation range polytope P_s of such a stage $s \in \{s_1, \ldots, s_{n_c}\}$. For each compressor unit $u \in \mathcal{U}_s$ the corresponding polytope is denoted by P_u, and using this we can describe P_s as

$$\begin{aligned}
P_s := \{ \quad (p_l, p_r, q) \quad | \quad &\forall u \in \mathcal{U}_s \quad \exists (p_{l,u}, p_{r,u}, q_u) \in P_u \\
&\text{with } p_l = p_{l,u'} \quad \forall u' \in \mathcal{U}_s, \\
&p_r = p_{r,u'} \quad \forall u' \in \mathcal{U}_s, \\
&q = \sum_{u' \in \mathcal{U}_s} q_{u'} \quad \}.
\end{aligned}$$

In other words, a valid operation point of the stage is represented by each unit operating at the same incoming and outgoing pressures, while the the mass flow through the stage is the sum of mass flow through the individual units.

In a similar fashion, we can then define the polytope P_c for the overall configuration. Here, the mass flow through all the stages stays the same, while the outgoing pressure of some stage s_i in the sequence has to match the incoming pressure of the subsequent stage s_{i+1}. Using this logic P_c can be defined as

$$
\begin{aligned}
P_c := \{ \quad (p_l, p_r, q) \quad | \quad & \forall s \in \{s_1, \dots, s_{n_c}\} \quad \exists (p_{l,s}, p_{r,s}, q_s) \in P_s \\
& \text{with } p_l = p_{l,s_1}, \\
& \quad p_r = p_{r,s_{n_c}}, \\
& \quad p_{r,s_i} = p_{l,s_{i+1}} \quad \forall i \in \{1, \dots, n_c - 1\}, \\
& \quad q = q_{s'} \quad \forall s' \in \{s_1, \dots, s_{n_c}\} \quad \}.
\end{aligned}
$$

Finally, the set of half spaces $\mathcal{H}_c$ used in Section 2.1 to define the feasible operating range of the configuration c are simply the facets of the polytope P_c representing the feasible operating range.

Note, that due to the symmetric polytope creation the parallel compressor units of a stage do not need a specific order. In contrast, the serial stage sequence is indeed important since in general two sequences using the same stages but in a different order will have different operating ranges.

4 Specialized network station algorithm

The MIP model $\mathscr{P}$ presented in Section 2.10 turns out to be quite challenging according to our experiments even though we are only considering parts of a larger gas network by restricting ourselves to network stations. We therefore created a specialized algorithm to solve the problem for network stations. The baseline insight of it is that the elements in network stations are all very close to each other, making the corresponding pipes inside the station relatively short, see also Table 1 in the computational Section 5. This means that their capability to store gas, which is often referred to as *linepack*, is insignificant in comparison to the long pipelines in between the network stations. Thus there is no possibility to "prepare for the future", i.e., use linepack to pre-transport gas to handle upcoming critical demand situations inside the network station itself.

This led us to the idea of splitting the time coupled model $\mathscr{P}$ into individual stationary models to determine the best operation mode and flow direction for each individual time step. We then use this information to find a solution in terms of operation modes and flow directions over the whole time horizon. This solution will respect the transition time conditions, which have not been modeled in $\mathscr{P}$ explicitly, see the corresponding part of Section 2.7.

Since our goal is to find a feasible solution for the presented model $\mathscr{P}$ of Section 2.10, we will not only have to determine the network station operation modes and flow directions, but also all other involved quantities. Hence, after determining these values we still need to calculate a transient version of $\mathscr{P}$, which for example also takes care of minimizing the differences in the operation points of the single elements, see Section 2.9. We do so by prescribing the operation modes and flow directions determined in the stationary calculations to make the

model tractable. The whole algorithm is summarized in Algorithm 1, in which the creation of the operation mode sequence is already split into the two steps described in Section 4.2.

1 opModes, flowDirs ← initialSolutionCreation() `// see Section 4.2.1`
2 opModes, flowDirs ← improvementHeuristic(opModes, flowDirs)   `// see Section 4.2.2`
3 return transientSmoothing(opModes, flowDirs)   `// see Section 4.3`
Algorithm 1: Specialized network station algorithm

Note, that our algorithm does not have the guarantee of finding globally optimal solutions, which would have been the case when directly solving $\mathscr{P}$ enhanced by a proper description of the operation mode transition time constraints. However, it has the potential to obtain good overall solutions for $\mathscr{P}$, since we assume the operation mode depending objective function weights w^{om} for changing to a new operation mode and w^{us} for starting a new compressor unit to dominate the other objective weights, see also the used objective function values in our computational experiments stated in Section 5.1.

In this section we will first introduce in 4.1 the three different variants of the model $\mathscr{P}$ that we will use in our algorithm. Afterwards, we will present the algorithm itself, that determines operation modes in 4.2 and finish with the so-called smoothing procedure in 4.3, in which we will find the feasible transient solution to our problem.

4.1 Model variants

As mentioned above, we will not solve the complete transient model $\mathscr{P}$ directly, but use the following variants of it in our overall solution approach.

$\mathscr{P}^{\mathrm{s}}$ – **Stationary model** For the first variant we solve a <u>s</u>tationary version of the model to determine the best operation mode and flow direction for one independent time step t. The necessary changes in $\mathscr{P}$ mainly affect the pipe model. In the stationary case a pipe no longer has the possibility to store gas, since the incoming and outgoing pipe flows are balanced, i.e., $q_{l,a,t} = q_{r,a,t} = q_{a,t}$ for all pipes $(l, r) = a \in \mathcal{A}^{\mathrm{pi}}$. This is due to the Continuity Equation, in which $\partial_t p = 0$ holds as for all time dependent derivatives resulting in the mass flow balance. Hence, the Continuity Equation is no longer part of the model and the stationary Momentum Equation (20) for pipe $(l, r) = a \in \mathcal{A}^{\mathrm{pi}}$ and the stationary time step t we are considering can be stated as

$$p_{r,t} - p_{l,t} + \frac{\lambda_a L_a}{4 D_a A_a} \left(|v_{l,a}| + |v_{r,a}| \right) q_{a,t} + \frac{g s_a L_a}{2 R_{\mathrm{s}} T z_a} \left(p_{l,t} + p_{r,t} \right) = 0.$$

Note that we still calculate the fixed velocity based on the initial pressure and flow values from time step 0, since we are looking for a feasible solution for the original model $\mathscr{P}$ in the end.

For all other elements the constraints only consider exactly one time step and we therefore simply apply them for the one time step t of the stationary case. The only other part to adjust is the objective function, where we keep the penalties $\sigma_{v,t}^{p+}$, $\sigma_{v,t}^{p-}$, $\sigma_{v,t}^{d+}$, and $\sigma_{v,t}^{d-}$ for each boundary node $v \in \mathcal{V}^{\mathrm{b}}$ since they are defined

24

based on each individual time step t. In addition, we will keep tracking the change to a new network station operation mode in variable δ_t^{om} as well as the start of new compressor units using $\delta_{u,t}^{\text{us}}$ for all $u \in \mathcal{U}_a\, a \in \mathcal{A}^{\text{cs}}$. This is done by calling the model with the parameter `prevMode`, where `prevMode` represents the operation mode of the previous time step. The other variables tracking changes in regulator modes $\delta_{a,t}^{\text{rg}}$ as well as the current point of operation of regulators and compressor stations as $\delta_{a,t}^{\text{rg}-\text{pl}}$, $\delta_{a,t}^{\text{rg}-\text{pr}}$, $\delta_{a,t}^{\text{rg}-\text{q}}$, $\delta_{a,t}^{\text{cs}-\text{pl}}$, $\delta_{a,t}^{\text{cs}-\text{pr}}$, and $\delta_{a,t}^{\text{cs}-\text{q}}$ for all $a \in \mathcal{A}^{\text{rg}}$ respectively $a \in \mathcal{A}^{\text{cs}}$ will be removed from the model, as well as the constraints (48)-(53) respectively (55)-(66) defining their behavior. The final stationary objective function for the time step t under consideration reads as

$$
\begin{aligned}
\min \quad & \big(\tau(t) - \tau(t-1)\big) \cdot \sum_{v \in \mathcal{V}^{\text{b}}} w^{\sigma-\text{p}} \cdot (\sigma_{v,t}^{p+} + \sigma_{v,t}^{p-}) + w^{\sigma-\text{d}} \cdot (\sigma_{v,t}^{d+} + \sigma_{v,t}^{d-}) \\
& + w^{\text{om}} \cdot \delta_t^{\text{om}} \\
& + \sum_{u \in \mathcal{U}_a, a \in \mathcal{A}^{\text{cs}}} w^{\text{us}} \cdot \delta_{u,t}^{\text{us}}
\end{aligned}
$$

$\mathscr{P}^{\text{f}}$ – **Transient with fixed operation modes and flow directions** For this variant, we are given a fixed operation mode o_t and flow direction f_t for all future time steps $t \in \mathcal{T}$. This is used in our algorithm to finally determine all transient quantities after the decision for an operation mode and a flow direction for each time step has been made. By fixing the corresponding variables $m_{o_t,t}^{\text{om}}$ and $m_{f_t,t}^{\text{fd}}$ to 1 for all future time steps $t \in \mathcal{T}$, the majority of binary variables can be replaced by constants, since the operation modes already decide valve and compressor station modes, as well as the configuration of all active compressor stations. Only the binary variables $m_{a,t}^{\text{ac}}$, $m_{a,t}^{\text{by}}$ and $m_{a,t}^{\text{cl}}$ for the mode of a regulator $a \in \mathcal{A}^{\text{rg}}$ are still to be decided.

In addition, a lot of implicating big-M constraints can already be resolved, i.e., we can remove the current formulation and just add the implied constraints, if the corresponding condition is fulfilled. Examples for this are the constraints (22)-(25) describing the valve behavior or the flow direction conditions (43). Furthermore, we no longer need to use a disjunctive model, but can apply the corresponding constraints (2)-(12) of the active mode and/or configuration of time step t directly to the variables $p_{l,t}$, $p_{r,t}$ and $q_{a,t}$ for each compressor station $(l, r) = a \in \mathcal{A}^{\text{cs}}$.

$\mathscr{P}^{\text{sf}}$ – **Stationary with fixed operation mode** As a last variant we basically combine the two variants above and use the stationary version of the model with already fixed operation mode. Note, that in contrast to model $\mathscr{P}^{\text{f}}$ the flow direction of the network station is not already given, which results in more binary variables and still to decide big-M constraints. However, this variant still results in a very small and rather simple model and we can therefore solve it very often to test the appropriateness of a given operation mode for a certain time step.

4.2 Determining operation modes

Our algorithm to determine the operation modes of the network station is split into two steps: First, we create an initial solution by a greedy, forward oriented procedure presented in 4.2.1. We then in a second step improve this solution by

testing if certain operation modes can be replaced by similar ones to find a better sequence of operation modes over time. This second step is described in 4.2.2.

Regarding the flow directions we will return for each time step t that flow direction, which has been chosen in the optimal solution of the stationary model which determined the returned operation mode for t. For simplicity of notation, we will not further mention them in the rest of this section.

4.2.1 Initial solution creation

To find a first feasible sequence of operation modes over time, we follow a rather simple idea. In order to keep the number of needed operation mode changes small, we determine an operation mode for time step t by first testing the used operation mode of the previous time step $t-1$ using $\mathscr{P}^{\mathrm{sf}}$. Only when this previously used operation mode yields a costly solution in terms of the objective function value will we use the general stationary model $\mathscr{P}^{\mathrm{s}}$ to determine the best operation mode for t. By this mechanic, we also reduce the amount of calls to the $\mathscr{P}^{\mathrm{s}}$ model, which are in general much more expensive in terms of computing time then calls to the $\mathscr{P}^{\mathrm{sf}}$ model. A detailed description is given as Algorithm 2.

There are a few things to note about Algorithm 2. First, the parameter `prevMode` given to the calls for solving the models $\mathscr{P}^{\mathrm{sf}}$ in line 5 and $\mathscr{P}^{\mathrm{s}}$ in line 13 is the operation mode of the previous time step, which we need to determine the operation mode change and compressor unit start variables δ_t^{om} respectively $\delta_{u,t}^{\mathrm{us}}$ for some unit $u \in \mathcal{U}_a$ $a \in \mathcal{A}^{\mathrm{cs}}$ and $t \in \mathcal{T}$ as explained in Section 4.1. In the call to $\mathscr{P}^{\mathrm{s}}$ we furthermore give the parameter `validModes`, which replaces the set of valid network station operation modes $\mathcal{O}$. We also call the functions `modeAvailable` and `transitionsWork` in Algorithm 2. These refer to the operation mode unavailability and the transition time restriction introduced in Section 2.7, where `modeAvailable` checks for a given operation mode o and time t if o is available at t and `transitionsWork` performs the checks described in Section 2.7 to test if a given operation mode sequence is valid regarding the corresponding transition times. In addition, Algorithm 2 uses a function called `notSoonInfeasible`. Here, we check if choosing a new operation mode for time t would result in an infeasibility at one of the subsequent time steps caused by a combination of operation mode unavailability and too long transition times. More specifically, we check if the new operation mode for time t will become unavailable in one of the future time steps. If this is the case, we then test if there is enough time left to transition into another operation mode until then, also taking into account the time needed by the transition from the old operation mode at time $t-1$ to this new operation mode at time t. This look into the near future turned out to be necessary according to our computational experiments in order to avoid cases where Algorithm 2 gets stuck in infeasible situations.

Two other lines in the algorithm may require further explanation. In line 6 we decide if the previous operation mode is good enough for the current time step by comparing its stationary objective function value against the cost of an operation mode change w^{om}. If the objective function value is indeed smaller we know that the previous operation mode is the best option considering this individual time step given the chosen operation modes for the past time steps, since each other operation mode would at least have to pay the penalty of w^{om} for changing the operation mode. As a last point to mention, Algorithm 2 may abort without a

Data: Operation mode o_0 of the network station in the initial state
Result: A list of operation modes for each time $t \in \mathcal{T}_0$

1 operationModes ← list()
2 operationModes.add(o_0)
3 for $t \in \mathcal{T}$ **do**
4 $\quad$ oldMode ← operationModes.last()
$\quad$ // call $\mathscr{P}^{\mathrm{sf}}$ with fixed operation mode oldMode for time t
5 $\quad$ oldModeFeasible, oldModeCost
$\qquad$ ← $\mathscr{P}^{\mathrm{sf}}(\text{oldMode}, \mathrm{t}, prevMode = \text{oldMode})$
6 $\quad$ **if** oldModeFeasible
$\qquad$ **and** *modeAvailable*(oldMode, t)
$\qquad$ **and** oldModeCost $< w^{\mathrm{om}}$ **then**
$\qquad$ $\quad$ // operation mode of t-1 is also good for t
7 $\qquad$ $\quad$ operationModes.add(oldMode)
8 $\quad$ **else**
$\qquad$ // operation mode of t-1 is NOT good for t
$\qquad$ // $\Rightarrow$ search best possible valid stationMode for t
9 $\qquad$ validModes ← list()
10 $\qquad$ **for** $o \in \mathcal{O}$ **do**
11 $\qquad$ $\quad$ **if** *modeAvailable*(o, t)
$\qquad$ $\qquad$ **and** *transitionsWork*(*concat*(operationModes, *list*(o)))
$\qquad$ $\qquad$ **and**
$\qquad$ $\qquad$ *notSoonInfeasible*(t, $newMode = o, oldMode = $ oldMode)
$\qquad$ $\qquad$ **then**
12 $\qquad$ $\quad$ $\quad$ validModes.add(o)
$\qquad$ // find best from validModes by calling $\mathscr{P}^{\mathrm{s}}$ for time t
13 $\qquad$ bestMode, bestModeFeasible, bestModeCost
$\qquad$ $\quad$ ← $\mathscr{P}^{\mathrm{s}}(\mathrm{t}, \mathcal{O} = \text{validModes}, prevMode = \text{oldMode})$
14 $\qquad$ **if** *not* bestModeFeasible **then** abort without solution
$\qquad$ // bestMode is best choice for t
15 $\qquad$ operationModes.add(bestMode)
16 return operationModes

Algorithm 2: Initial solution creation

feasible solution in line 14. This is no proof of infeasibility, since in theory cases are possible, where we abort although a feasible solution exists. However, in all of our test cases we have never aborted the algorithm at this point, also see our computational results in Section 5.3. Furthermore, the combination of transition times and unavailable operation modes can lead to very hard to find feasible solutions, which makes the design of an algorithm performing reasonably fast on average but guaranteeing to find all feasible solutions a challenge. Therefore, we leave this problem open for future research.

4.2.2 An improvement heuristic

After we have found a feasible solution using Algorithm 2, we look for further improvements of it. Due to its design, Algorithm 2 only considers individual time steps to decide which operation mode to choose for each time step. However, we

can easily imagine a situation in which the operation mode o_1 found by $\mathscr{P}^s$ is best for time step t, but another operation mode o_2 is slightly better for all subsequent time steps and would have been the overall better choice at time t. We might even be able to avoid operation mode changes, if o_1 would become unavailable in the future, while o_2 has only slightly worse objective function values, but stays available.

Data: A sequence S of valid operation modes over time
Result: A valid sequence S^* of operation modes with $\mathsf{obj}(S^*) \leq \mathsf{obj}(S)$

```
 1  backwards ← True
 2  while not having two iterations without improvements do
 3  │   changeTimes ← list of times t with S[t − 1] ≠ S[t]
 4  │   if backwards then reverse(changeTimes)
 5  │   for t ∈ changeTimes do
 6  │   │   if S[t − 1] = S[t] then continue
 7  │   │   if backwards then
 8  │   │   │   phaseToReplace ← phase ending at S[t − 1]
 9  │   │   else
10  │   │   │   phaseToReplace ← phase starting from S[t]
11  │   │   S^best, bestImprovement ← list(), 0.0
12  │   │   for newMode ∈ convexCombination(S[t − 1], S[t]) do
13  │   │   │   S^new ← S.replace(phaseToReplace, newMode)
14  │   │   │   improvement ← obj(S) − obj(S^new)
15  │   │   │   if allModesAvailable(S^new)
        │   │   │       and transitionsWork(S^new)
        │   │   │       and improvement > bestImprovement then
16  │   │   │   │   S^best, bestImprovement ← S^new, improvement
17  │   │   if bestImprovement > 0 then
        │   │   │   // Found improvement in this interation!
18  │   │   │   S ← S^best
19  │   backwards ← not backwards
20  return S
```

Algorithm 3: Improvement heuristic

To deal with these situations, we created for a given feasible solution, represented by a sequence of operation modes over time, the improvement heuristic stated as Algorithm 3. Here the idea is, to identify all sequences of identical operation modes over time in the solution. We call these sequences stable phases or just *phases* of a feasible solution. Obviously, the switch from one phase to the subsequent one happens if the operation mode changes to a new mode. For each of these phases we then check if we can replace the operation mode of the whole phase with a similar one being more beneficial in terms of the objective function value.

We obtain these similar network station operation modes from the call of the function `convexCombination`, which is the key feature of Algorithm 3. To define it, we use the the function $M(o, a)$ returning the mode or active configuration of a valve or compressor station a in operation mode o, see Section 2.7. Furthermore, we denote by $\mathcal{U}(x)$ the compressor units used in mode or configuration $x \in \{by, cl\} \cup \mathcal{C}_a$ for some compressor station $a \in \mathcal{A}^{cs}$, where $\mathcal{U}(by) = \mathcal{U}(cl) = \emptyset$.

28

Then we first define the function `convexCombinationCS` on a tuple (x, y) with $x, y \in \{\text{by}, \text{cl}\} \cup \mathcal{C}_a$ as

$$\texttt{convexCombinationCS}(x, y) := \{x, y\} \cup$$
$$\Big\{ c \in \mathcal{C}_a \quad | \quad \forall u \in \mathcal{U}(x) \cap \mathcal{U}(y) : u \in \mathcal{U}(c)$$
$$\wedge \, \forall u \in \mathcal{U}(c) : u \in \mathcal{U}(x) \cup \mathcal{U}(y) \Big\}.$$

Note, that $\texttt{convexCombinationCS}(x, y) \subseteq \{\text{by}, \text{cl}\} \cup \mathcal{C}_a$ holds. Then we are ready to define `convexCombination` on a tuple (o_1, o_2) of operation modes as

$$\texttt{convexCombination}(o_1, o_2) := \Big\{ o \in \mathcal{O} \, |$$
$$\big(\forall a \in \mathcal{A}^{\text{va}} : M(o, a) = M(o_1, a) \quad \vee \quad \forall a \in \mathcal{A}^{\text{va}} : M(o, a) = M(o_2, a) \big)$$
$$\wedge \quad \forall a \in \mathcal{A}^{\text{cs}} : M(o, a) \in \texttt{convexCombinationCS}\big(M(o_1, a), M(o_2, a) \big) \Big\}.$$

Note here, that while we allow a compressor station a to have a configuration using a compressor unit set "in between" the used compressor unit set of the configurations used in o_1 and o_2 for a, we only allow the exact valve mode combination used in o_1 or the one used in o_2. The reason for this is, that a valve mode combination enables a very specify set of paths through each network station, and it is very unlikely that a valve mode set obtained from combining the modes used in the two given operation modes yields operation modes, which are able to handle the same demand situation. Since, each of the modes obtained by calling `convexCombination` is tested in Algorithm 3, we hereby restrict the result set to the most promising candidates.

Apart from calling `convexCombination`, Algorithm 3 uses the two functions `transitionsWork`, which works in the same way as described for Algorithm 2 above and `allModesAvailable`, which is similar to `modeAvailable` from Algorithm 2, but instead of checking the availability of a given operation mode o for time t checks the availability of a whole sequence of operation modes at the times corresponding to the position in the sequence. Furthermore, we evaluated the objective function value of a sequence of operation modes using the function `obj` by successively calling $\mathscr{P}^{\text{sf}}$ for each operation mode and time corresponding to its position in the sequence. If one of the models turns out to be infeasible, the returned objective function value will be infinity.

Finally, we note that we decided to start the algorithm in the backwards oriented mode. The reason for this decision is that the initial solution is obtained by Algorithm 2 which was operated in a forward direction. Furthermore, we highlight that Algorithm 3 has the potential to reduce the total number of needed operation mode changes, since the two original operation modes are always part of the result of `convexCombination`. In addition, it is possible that an operation mode change from the loop of line 5 has already been removed in the previous iteration by replacing one of the involved operation modes with the other one, which makes the check in line 6 necessary.

4.3 Transient solution smoothing

As a final step of our specialized network station algorithm, we solve the transient model variant $\mathscr{P}^{\text{f}}$ with fixed network station operation modes and flow directions.

We obtain both for each time step from the stationary model solutions created in the previous steps of the algorithm. When comparing pressure and flow values at single points of network over time, we expect the transient solution states to be more similar in general and in case of changing conditions to be more smooth compared to the series of stationary solution states. This is due to the missing penalty of operation point changes in the independent stationary models, which may result in considerably different solution states. This difference may for example occur in the pressure level of nodes inside the station, even if the demand situation as well as the determined operation mode and flow direction are the same.

> **Data:** A sequence S of tuples of operation modes and flow directions over time as well as a time horizon size h
> **Result:** A set of transient solution states for all $t \in \mathcal{T}_0$
> **1 if** h $\geq |\mathcal{T}_0|$ **then**
>   // overall time horizon covered by smoothing time horizon
> **2**   return $\mathscr{P}^{\mathrm{f}}(S, initialState = \text{initialState})$
> **3** solutionStates $\leftarrow$ list()
> **4** solutionStates.add(initialState)
> **5** currTime $\leftarrow 0$
> **6 while** currTime $+ h \leq |\mathcal{T}_0|$ **do**
> **7** $S^{\mathrm{h}} \leftarrow S.\text{slice}(\text{currTime}, \text{currTime} + \text{h})$
> **8**   thisTimeStates $\leftarrow \mathscr{P}^{\mathrm{f}}(S^{\mathrm{h}}, initialState = \text{solutionStates.last())}$
> **9**   **if** currTime $+ h = |\mathcal{T}_0|$ **then**
> **10**   solutionStates.addAll(thisTimeStates)
> **11**   **else**
> **12**   solutionStates.add(thisTimeStates.first())
> **13**   currTime $\leftarrow$ currTime $+ 1$
> **14** return solutionStates;

Algorithm 4: Transient smoothing

In our computational experiments we observed that even though most of the binary decision variables of $\mathscr{P}$ are fixed in $\mathscr{P}^{\mathrm{f}}$, only a limited number of time steps can be solved for large network stations. Therefore, we use a *rolling horizon* approach to solve $\mathscr{P}^{\mathrm{f}}$, which is described in Algorithm 4. Here, we specify a small fixed time horizon size h, which represents the number of time steps to solve in model $\mathscr{P}^{\mathrm{f}}$ including the time step for the given initial state. We then solve a series of models $\mathscr{P}^{\mathrm{f}}$, while always fixing the earliest time step and shifting the time horizon by 1 in each iteration. In the function call to solve $\mathscr{P}^{\mathrm{f}}$ in Algorithm 4, we give the subsequence of operation modes and flow directions, corresponding to and also encoding the current time horizon to solve. Furthermore, we specify the state to use as the fixed initial state.

The main benefit of this method is, that increasing the size $|\mathcal{T}_0 := \{0, \ldots, k\}|$ of the overall time horizon only increases the number of equally sized and therefore similarly complex MIP models to solve rather then increasing the complexity of the model, which may lead to an exponential increase in runtime.

5 Computational experiments

In order to verify the competitiveness of Algorithm 1 in terms of both solution quality as well as execution time, we evaluated it to a large number of test instances. These instances represent network stations in the network of our project partner Open Grid Europe(OGE), for which we generated scenario values based on historic real-world situations in the network.

Throughout this chapter, we will state all flow values as volumetric flow under normal conditions in the unit $1000\text{m}^3/\text{h}$. To create these values from mass flow values in kg/s we use the normal density ρ_0 given in kg/m^3, which is determined by the gas mixture and assumed to be constant all over the network. The linear transformation formula then reads as $[1000\text{m}^3/\text{h}] = \frac{3600}{1000\rho_0}[\text{kg/s}]$.

5.1 Instances

We consider 7 different network stations from the network of OGE with different sizes and properties. An overview can be found in Table 1. For each of these stations, we have a set of 159 instances. These have been created by solving a macroscopic gas flow model on an aggregated complete network containing simplified versions all the network stations. A full description of the model can be found in [17].

Name	$\mid\mathcal{V}\mid$	$\mid\mathcal{A}\mid$	$\frac{\sum_{a\in\mathcal{A}^{\text{pi}}} L_a}{\mid\mathcal{A}^{\text{pi}}\mid}$	$\mid\mathcal{C}_a\mid\ \forall a\in\mathcal{A}^{\text{cs}}$	$\mid\mathcal{O}\mid$	$\mid\mathcal{F}\mid$	$\mid\mathcal{V}^{\text{b}-\text{ex}}\mid$	$\mid\mathcal{W}\mid$
A	14	11	0.001 km	16	34	3	1	0
B	11	12	0.001 km	18	23	2	1	0
C	27	34	0.012 km	2	13	4	2	0
D	25	31	0.404 km	2, 6	92	6	2	0
E	48	67	0.308 km	3, 5	82	12	1	4
F	51	66	0.024 km	2, 3, 6, 12	2836	3	3	0
$\text{G}_{\min}$	118	148	0.079 km	1, 1, 2, 7, 20	1267	15	2	0
$\text{G}_{\max}$	120	150	0.079 km	1, 1, 2, 7, 20	1285	20	2	0

Table 1: Overview of different properties of the 7 network stations A to G. For station G the topology changed during the considered time period of 91 days. Hence, we denoted by $\text{G}_{\min}$ the minimal values of all the quantities for station G and by $\text{G}_{\max}$ the corresponding maximum values.

As initial states we used state values that actually occurred for the network of our project partner. Analogously we are given measured pressure and flow values for the next 12 hours after the initial state time at the boundary nodes of the network containing all the 7 network stations. These instances are distributed over a period of 91 days. For each day we have two instances, whose initial state times have a difference of 12 hours. Inside this data period, we unfortunately are missing instances for 8 days as well as having only one instance for another 7 days due to technical problems during the data creation at our project partner. Therefore, our final instance set consists of $2 \cdot 91 - 2 \cdot 8 - 7 = 159$ instances.

The solutions of the macroscopic gas flow model for each of these instances yield pressure and inflow values at the boundaries of each network station. These values represent the scenario values for each instance of the test set of this paper.

For the macroscopic gas flow model, the time horizon of 12 hours has been split into 4 periods of 15 minutes followed by 11 periods of 60 minutes, resulting in 15 future time steps in total.

For our analysis, we created the following four different partitions of the 12 hour time horizon, where the horizon is built from left to right per row.

12 time steps:	$4 \times$ 15 min, 5×60 min, 3×120 min
24 time steps:	$4 \times$ 15 min, 18×30 min, $2 \times$ 60 min
48 time steps:	$48 \times$ 15 min
96 time steps:	96×7.5 min

Our partitioning of the time horizon can have additional or missing time points compared with the 15 time steps used in the macroscopic model. In order to create scenarios values that fit the time horizon we interpolate using the original values.

To conclude, we used the following set of weights for the objective function for all instances:

$$w^{\sigma-\mathrm{p}} = \frac{1000.0}{\mathrm{bar} \cdot \mathrm{h}} \qquad w^{\mathrm{om}} = 1000.0 \qquad w^{\mathrm{rg-pl}} = \frac{10.0}{\mathrm{bar}} \qquad w^{\mathrm{cs-pl}} = \frac{10.0}{\mathrm{bar}}$$

$$w^{\sigma-\mathrm{d}} = \frac{100.0}{1000 \, \mathrm{m}^3} \qquad w^{\mathrm{us}} = 1200.0 \qquad w^{\mathrm{rg-pr}} = \frac{10.0}{\mathrm{bar}} \qquad w^{\mathrm{cs-pr}} = \frac{10.0}{\mathrm{bar}}$$

$$w^{\mathrm{rg}} = 50.0 \qquad w^{\mathrm{rg-q}} = \frac{1.0}{1000 \, \frac{\mathrm{m}^3}{\mathrm{h}}} \qquad w^{\mathrm{cs-q}} = \frac{1.0}{1000 \, \frac{\mathrm{m}^3}{\mathrm{h}}}$$

Note that the unit of $w^{\sigma-\mathrm{d}}$ is deduced by $\frac{1}{1000 \, \frac{\mathrm{m}^3}{\mathrm{h}} \cdot \mathrm{h}} = \frac{1}{1000 \, \mathrm{m}^3}$.

5.2 Computational setup

We performed our computations on a cluster using 4 cores and 16 GB of RAM of a machine being composed of two *Intel Xeon CPU E5-2670 v2* running at 2.50 GHz. As a solver for the underlying MIP problems we used *Gurobi* in version 8.1.0 [12], which we accessed via the Pyomo modeling language [14][15]. Since the corresponding MIP models are numerically challenging we used the solver with maximal *NumericFocus* parameter. In addition, we specified in Table 2 the optimality conditions and maximum run times for each solved model variant.

Variant	Rel. Gap	Abs. Gap	Time limit
$\mathscr{P}^{\mathrm{s}}$, $\mathscr{P}^{\mathrm{sf}}$	1E-4	1E-2	10h
$\mathscr{P}^{\mathrm{f}}$	5E-3	1E-2	60s

Table 2: Optimality conditions and maximum run times we use for the single model variants. Note that the 10 hours serve as a representation for ∞, i.e., we have chosen the time limit high enough to always solve the models to optimality.

Finally, we specify the rolling horizon parameter h to be 4, so we are always solving the smoothing for 4 future time steps. We found in our experiments, that this number represents a good trade-off between efficient model solving speed and foresightedness of the obtained solution.

5.3 Results

As described in Section 5.1 above, we tested Algorithm 1 on 159 start times $\times$ 7 network stations $\times$ 4 time horizons $= 4452$ instances. For all of these instances, it found a feasible solution. In particular, we always found a feasible solution in the initial solution creation in Algorithm 2 described in Section 4.2.1 although this is not guaranteed by the algorithm itself.
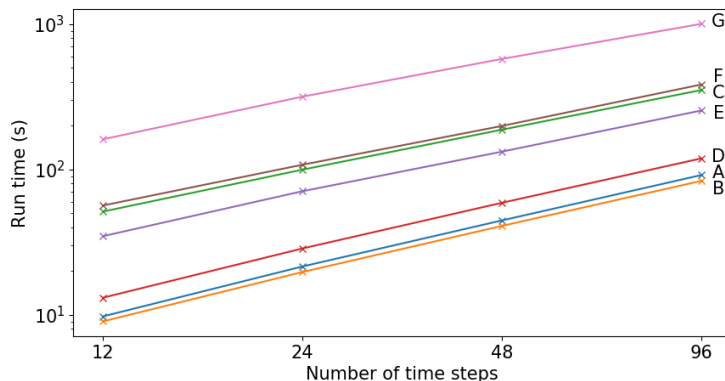

Figure 5: Average run times for Algorithm 1 sorted by number of time steps and network station.

In Figure 5 one can find an overview of the average run times of Algorithm 1 sorted by station and number of solved time steps. As one can see, the ranking of the single stations regarding run time as well as the ratio between the single stations run times is stable over the different time horizon partitions. Furthermore, the run time per station corresponds roughly to the complexity deduced from the station statistics in the sense that more nodes, arcs and operation modes make the overall problem more complex to solve and therefore increase the run time. The exception to this reasoning is station C, which has a rather high average run time although its node and arc counts are comparable to station D and the number of operation modes is even the smallest of all seven stations. We assume that this is due to the stations topology, in which the regulators seem to be arranged in a way which is hard to solve for the final smoothing step of the algorithm. Another noticeable fact is the almost perfectly proportional average run time increase with increasing number of time steps. As mentioned in Section 4.3 this property was an explicit goal when designing the algorithm.

In addition to the average run times, we also depict the distribution of run times sorted by station in Figure 6 for instances solved for 12 time steps. The distribution of the other time steps are quite similar in appearance and can be found in the Appendix as Figures A.1, A.2, and A.3 respectively.

For the fast solving stations A, B, and D, the run times of the instances are all very similar in that the middle box is barely visible. Station C is again outstanding by having a rather large spread of run times only matched by the biggest and most complex station G. However, in general the difference between the median and maximum run time are lower than an order of magnitude for all of the stations. That means that for our instance set even the extreme cases are still in reach and have a somewhat similar run time to the rest of the instances,
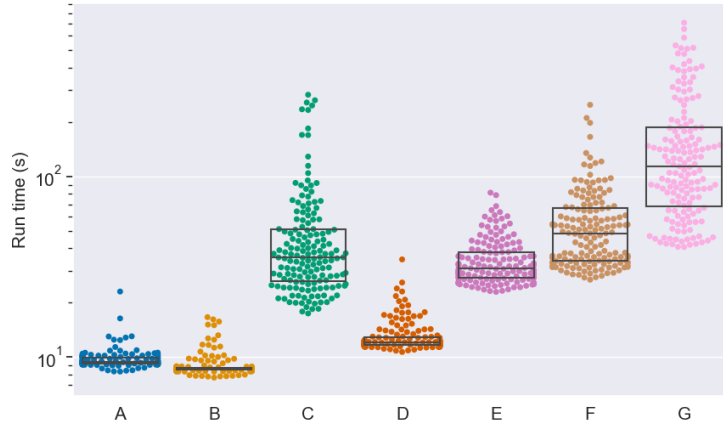
33

Figure 6: Run times of instances with 12 time steps, sorted by station. One dot represents one instance, however dots may overlap when too many have similar run time. In addition we draw a typical whiskerless box plot. Therefore the lines represent the 25th percentile, the median and the 75th percentile.

making Algorithm 1 well suited for strict time limit situations, which we would face in a production environment.

As the last run time related graphic, Figure 7 shows the average portion of run time spent in the single subroutines of Algorithm 1 for 12 time steps as well as for 96 time steps. It states that the transient smoothing dominates the run time already for 12 time steps. With an increasing number of time steps, the influence even increases, while the improvement heuristics impact is decreasing. This can also be verified through the 24 and 48 time step instances displayed in Figure A.4 in the Appendix. The observed shares fit to the structure of the single subroutines from Section 4. While the initial solve as well as the smoothing have to solve more instances of the challenging model variants $\mathscr{P}^{\mathrm{s}}$ respectively $\mathscr{P}^{\mathrm{f}}$ with each additional time step, the improvement heuristic iterates over pairs of subsequent time steps with different operation modes. Since the differently sized time horizon partitions all cover the same 12 hours, the number of total operation mode changes in the result of the initial solve is very unlikely to increase much with increasing time granularity. This is also due to the initial solve algorithm itself, which tries to use the same operation mode as long as possible before changing to a different one.

Apart from the run time we are also interested in the quality of the solution of Algorithm 1. Therefore, we solved the complete model $\mathscr{P}$ directly on the smaller instances A to E to obtain a valid lower bound for them. Note that $\mathscr{P}$ itself is already only a relaxation of the problem solved by Algorithm 1, since the transition time restrictions for operation modes mentioned in Section 2.7 are not included in this model. The obtained lower bound can be used to give an upper bound on the relative difference of the solution found by Algorithm 1 to the optimal solution. We call this difference *gap* and define it as $\frac{\mathrm{obj(solution)} - \mathrm{lowerBound}}{\mathrm{obj(solution)}}$. Since the lower bound cannot be negative, the gap is always smaller than or equal to 100%.

We solved model $\mathscr{P}$ on stations A to E for 12 time steps using the same com-
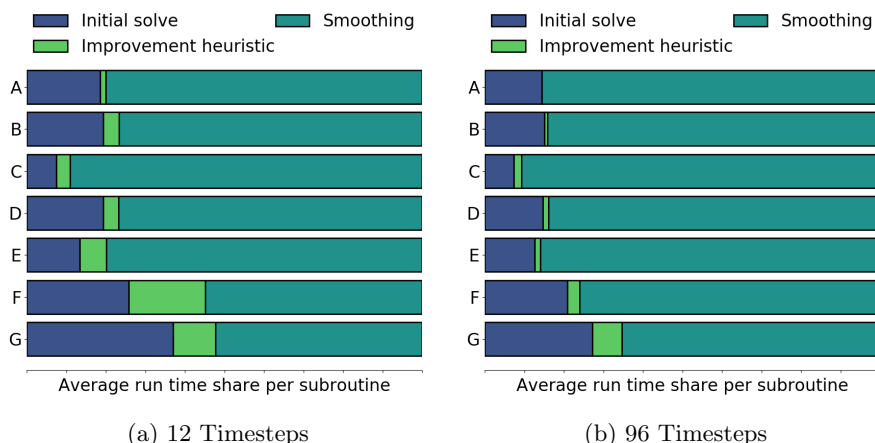
Figure 7: Portion of run time spend in the single subroutines of Algorithm 1 displayed for each of the single stations. Figure 7a represents the results for 12 time steps, Figure 7b the results for 96 time steps.

putational setup as described above with a time limit of 10 hours. Furthermore, we gave the solution obtained from Algorithm 1 as a starting solution. The solver finished all 159 instances of the stations A and B within the time limit. For the stations C, D and E we hit the time limit for 47, 8 and 18 instances respectively, resulting in potentially sub-optimal lower bounds in these cases. For the stations F and G more than half of the instances did not finish in time, which is why we did not include these in our evaluation.

The results can be found in Figure 8, where we plotted the gap for all 5 network stations on a logarithmic scale and then plotted the same data for stations C and E on a linear scale to better depict their distributions. Note that the instances which have a potentially suboptimal lower bound are marked with triangles instead of dots. As one can observe, we have good results for all of the network stations with more than half of the instances for each station having a solution with at most 10% difference to the lower bound. For the best three stations A, B, and D we can even state that at least 75% the of values have less than 1% difference to the optimal solution. The picture is more diverse for the other two stations, in which the 75% percentile is between 20% and 30% gap. However, for those stations we have the highest value of instances for which $\mathscr{P}$ did not solve the problem in time. Therefore, there is the potential for further improvement of the gap by increasing the lower bound here. For all stations C, D and E with instances having potentially suboptimal lower bounds, the corresponding instances are among those having high gap values. The picture is extreme for Station C, where all but 3 instances above the 75% percentile had suboptimal lower bounds, making it very likely that one can further decrease the gap value by increasing the run time limit of $\mathscr{P}$ directly.

Altogether, the results show that Algorithm 1 is able to find good solutions in a short time and is therefore able to solve even large instances of the presented gas transportation problem, which would be far out of reach when trying to solve the original MIP model formulation $\mathscr{P}$ directly.
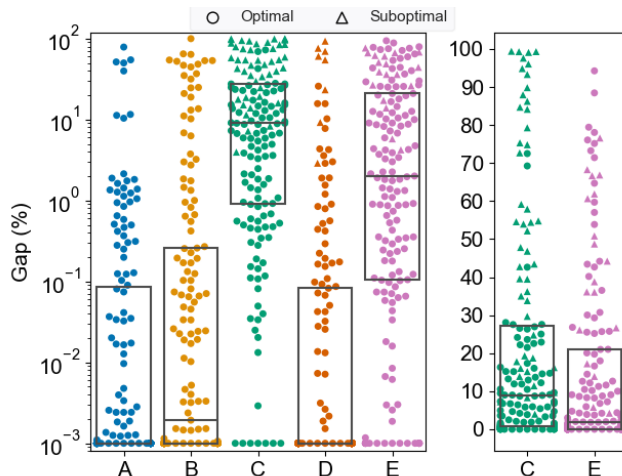
35

Figure 8: Gap between the solution of Algorithm 1 and a lower bound obtained from solving $\mathscr{P}$ directly. We considered the stations A to E and solved them using 12 time steps. One dot represents one instance, however dots may overlap when too many have similar run time. The instances with potentially suboptimal lower bounds due to hitting the time limit are marked by triangles instead of dots. In addition we draw a typical whiskerless box plot. Therefore the lines represent the 25th percentile, the median and the 75th percentile. The left graphic uses logarithmic scale for the gap and the right one displays the same data for stations C and E on linear scale. For all instances in which the objective function value and the lower bound are smaller than 0.1 we defined the gap to be zero. Furthermore we plotted all values below 10^{-3} as 10^{-3} for the logarithmic scale.

6 Conclusion

We presented the transient gas network transportation problem on so-called network stations, which represent the intersection points of major transportation pipelines and contain the majority of active elements to control the network. For this problem we introduced a mixed-integer programming model including a complex model for compressor stations as well as additional variables and constraints for the network station itself. For the pipes, we found that due to their shortness they have less overall impact in network stations, which enabled us to use a linear description for them. Using state-of-the art solvers the MIP Model is not tractable for large network stations. Therefore we developed a specialized algorithm to solve the problem. Here we again use the fact that network stations contain only short pipes and therefore negligible linepack, causing the decision making to mostly depend on the current demand situation at the boundary of the station. Therefore, we determined the operation modes and gas flow directions of the network station based on multiple solves of a stationary version of the MIP. By using this approach we are also able to satisfy transition time constraints for the operation modes, which have been excluded from the original MIP formulation. In order to obtain a feasible solution for the problem, we finally solved another variant of the MIP with fixed network station operation modes and flow directions in a rolling horizon fashion.

To verify the competitiveness of our algorithm regarding both run time and solution quality, we did tests on 159 different scenarios based on past flow situations in 7 real world network stations provided by our project partner. Our algorithm is able to compute feasible solutions for all of the presented instances very fast. Even for the biggest of the presented network stations consisting of more than 100 nodes and arcs as well as over 1000 different network station operation modes, our algorithm terminates on average in under 20 minutes for each of the stations. By running experiments using 4 different types of granularity, we found that the run time increases proportional to the number of time steps used, indicating that our algorithm scales very well. In terms of quality of the solution, we tested the results against a lower bound obtained from solving the original MIP formulation for 12 time steps on all except the biggest 2 network stations. For the worst of the 5 remaining stations, more than half of the instances have a solution with at most 10% difference to the lower bound and for the best 3 stations, we find near optimal solution with less than 1% difference for more than 75% of the instances. Altogether we were able to show, that our algorithm can reliably find good solutions to the problem in a short amount of time.

There are a lot of different possibilities to continue this research. To increase the model accuracy, the approximative linearization of the friction in pipes and resistors as well as the maximum power bound of compressor units could be replaced by their original non-linear versions. This would turn the model into a MINLP and thereby increase its complexity. From a theoretical point of view, extending Algorithm 2 such that it has the guarantee to always find existing feasible solutions would greatly improve the overall robustness. Finally, real-world gas network operation is a complicated business with a never-ending list of special elements and extra constraints, which can still be added to our model. As examples we name ramp-up and cool down times for compressor units as well as target value based control of regulators and compressor stations.

Acknowledgements

References

[1] E. Balas. Disjunctive Programming and a Hierarchy of Relaxations for Discrete Optimization Problems. *SIAM Journal on Algebraic Discrete Methods*, 6(3):466–486, 1985.

[2] E. Balas. The Convex Hull of a Disjunctive Set. In *Disjunctive Programming*, pages 17–39. Springer International Publishing, Cham, 2018.

[3] J. Brouwer, I. Gasser, and M. Herty. Gas pipeline models revisited: Model hierarchies, nonisothermal models, and simulations of networks. *Multiscale Modeling & Simulation*, 9(2):601–623, 2011.

[4] R. Burlacu, H. Egger, M. Groß, A. Martin, M. Pfetsch, L. Schewe, M. Sirvent, and M. Skutella. Maximizing the storage capacity of gas networks: A global MINLP approach. *Optimization and Engineering*, 20(2):543–573, June 2019.

[5] C. F. Colebrook and C. M. White. Experiments with fluid friction in roughened pipes. *Proceedings of the Royal Society of London. Series A - Mathematical and Physical Sciences*, 161(906):367–381, 1937.

[6] P. Domschke, B. Geißler, O. Kolb, J. Lang, A. Martin, and A. Morsi. Combination of Nonlinear and Linear Optimization of Transient Gas Networks. *INFORMS Journal on Computing*, 23(4):605–617, 2011.

[7] P. Domschke, B. Hiller, J. Lang, and C. Tischendorf. Modellierung von Gasnetzwerken: Eine Übersicht. 2017.

[8] K. Ehrhardt and M. C. Steinbach. Nonlinear Optimization in Gas Networks. Technical Report 03-46, ZIB, Takustr. 7, 14195 Berlin, 2003.

[9] Federal Ministry for Economic Affairs and Energy. Still indispensable for a reliable energy supply. `https://www.bmwi.de/Redaktion/EN/Dossier/conventional-energy-sources.html`, 2019. Accessed: 2019-10-31.

[10] A. Fügenschuh, B. Geißler, R. Gollmer, A. Morsi, M. E. Pfetsch, J. Rövekamp, M. Schmidt, K. Spreckelsen, and M. C. Steinbach. Physical and technical fundamentals of gas networks. In Koch et al. [19].

[11] M. Gugat, G. Leugering, A. Martin, M. Schmidt, M. Sirvent, and D. Wintergerst. MIP-based instantaneous control of mixed-integer PDE-constrained gas transport problems. *Computational Optimization and Applications*, 70(1):267–294, May 2018.

[12] L. Gurobi Optimization. Gurobi optimizer reference manual, version 8.1.0. http://www.gurobi.com, 2018.

[13] M. Hahn, S. Leyffer, and V. M. Zavala. Mixed-Integer PDE-Constrained Optimal Control of Gas Networks. *Preprint.*, 2017.

[14] W. E. Hart, C. D. Laird, J.-P. Watson, D. L. Woodruff, G. A. Hackebeil, B. L. Nicholson, and J. D. Siirola. *Pyomo–Optimization Modeling in Python*, volume 67. Springer Science & Business Media, second edition, 2017.

[15] W. E. Hart, J.-P. Watson, and D. L. Woodruff. Pyomo: Modeling and solving mathematical programs in Python. *Mathematical Programming Computation*, 3(3):219–260, 2011.

[16] F. Hennings. Benefits and Limitations of Simplified Transient Gas Flow Formulations. In *Operations Research Proceedings 2017*, pages 231–237. Springer, 2018.

[17] K. Hoppmann, F. Hennings, R. Lenz, and T. Koch. Optimal Operation of Macroscopic Gas Transport Networks Over Time. Technical Report 19-23, ZIB, Takustr. 7, 14195 Berlin, 2019.

[18] J. Humpola, A. Fügenschuh, B. Hiller, T. Koch, T. Lehmann, R. Lenz, R. Schwarz, and J. Schweiger. The specialized MINLP approach. In Koch et al. [19].

[19] T. Koch, B. Hiller, M. E. Pfetsch, and L. Schewe, editors. *Evaluating Gas Network Capacities*, volume 21 of *MOS-SIAM Series on Optimization*. SIAM, 2015.

[20] O. Kolb, J. Lang, and P. Bales. An implicit box scheme for subsonic compressible flow with dissipative source term. *Numerical Algorithms*, 53(2-3):293–307, 2010.

[21] D. Mahlke, A. Martin, and S. Moritz. A simulated annealing algorithm for transient optimization in gas networks. *Mathematical Methods of Operations Research*, 66(1):99–115, Aug. 2007.

[22] T. W. Mak, P. Van Hentenryck, A. Zlotnik, H. Hijazi, and R. Bent. Efficient dynamic compressor optimization in natural gas transmission systems. In *American Control Conference (ACC), 2016*, pages 7484–7491. IEEE, 2016.

[23] S. Moritz. *A Mixed Integer Approach for the Transient Case of Gas Network Optimization*. PhD thesis, Technische Universität Darmstadt, Darmstadt, 2007.

[24] J. Nikuradse. *Laws of Flow in Rough Pipes*. National Advisory Committee for Aeronautics Washington, 1950.

[25] F. M. Odom and G. L. Muster. Tutorial On Modeling of Gas Turbine Driven Centrifugal Compressors. In *PSIG-09A4*, PSIG, Jan. 2009. Pipeline Simulation Interest Group.

[26] A. J. Osiadacz. Different Transient Flow Models - Limitations, Advantages, And Disadvantages. *Pipeline Simulation Interest Group*, 1996.

[27] Papay. A termeléstechnológiai paraméterek változása a gáztelepek muvelése során. *OGIL Musz. Tud. Kozl.*, 1968.

[28] M. E. Pfetsch, A. Fügenschuh, B. Geißler, N. Geißler, R. Gollmer, B. Hiller, J. Humpola, T. Koch, T. Lehmann, A. Martin, A. Morsi, J. Rövekamp, L. Schewe, M. Schmidt, R. Schultz, R. Schwarz, J. Schweiger, C. Stangl, M. C. Steinbach, S. Vigerske, and B. M. Willert. Validation of Nominations in Gas Network Optimization: Models, Methods, and Solutions. *Optimization Methods and Software*, 2014.

[29] R. Z. Ríos-Mercado and C. Borraz-Sánchez. Optimization problems in natural gas transportation systems: A state-of-the-art review. *Applied Energy*, 147:536–555, 2015.

[30] C. Rocchini and P. Cignoni. Generating random points in a tetrahedron. *Journal of Graphics Tools*, 5, 02 2001.

[31] J. M. Saleh. *Fluid Flow Handbook*. McGraw-Hill Professional, 2002.

[32] T. Walther and B. Hiller. Modelling compressor stations in gas networks. Technical Report 17-67, ZIB, Takustr. 7, 14195 Berlin, 2017.

[33] A. Zlotnik, M. Chertkov, and S. Backhaus. Optimal control of transient flow in natural gas networks. In *54th IEEE Conference on Decision and Control (CDC)*, pages 4563–4570. IEEE, 2015.

A Appendix

Variable		Meaning	Unit
$p_{v,t}$	$\in \mathbb{R}_{\geq 0}$	pressure at node $v \in \mathcal{V}$	bar
$q_{a,t}$	$\in \mathbb{R}$	flow over arc $a \in \mathcal{A} \setminus \mathcal{A}^{\mathrm{pi}}$	kg/s
$q_{v,a,t}$	$\in \mathbb{R}$	flow into or out of pipe $a \in \mathcal{A}^{\mathrm{pi}}$	kg/s
$d_{v,t}$	$\in \mathbb{R}$	inflow into boundary node $v \in \mathcal{V}^{\mathrm{b}}$	kg/s
$m^{\mathrm{cf}}_{c,a,t}$	$\in \{0,1\}$	selection of configuration $c \in \mathcal{C}_a$ for $a \in \mathcal{A}^{\mathrm{cs}}$	1
$m^{\mathrm{by}}_{a,t}$	$\in \{0,1\}$	selection of bypass mode for $a \in \mathcal{A}^{\mathrm{rg}} \cup \mathcal{A}^{\mathrm{cs}}$	1
$m^{\mathrm{cl}}_{a,t}$	$\in \{0,1\}$	selection of closed mode for $a \in \mathcal{A}^{\mathrm{rg}} \cup \mathcal{A}^{\mathrm{cs}}$	1
$m^{\mathrm{ac}}_{a,t}$	$\in \{0,1\}$	selection of active mode for $a \in \mathcal{A}^{\mathrm{rg}}$	1
$m^{\mathrm{op}}_{a,t}$	$\in \{0,1\}$	selection of open/closed mode for $a \in \mathcal{A}^{\mathrm{va}}$	1
$p^{\mathrm{by}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	pressure in bypass mode for $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$q^{\mathrm{by}}_{a,t}$	$\in \mathbb{R}$	flow in bypass mode for $a \in \mathcal{A}^{\mathrm{cs}}$	kg/s
$p^{\mathrm{l-cl}}_{c,a,t}$	$\in \mathbb{R}_{\geq 0}$	incoming pressure in closed mode for $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$p^{\mathrm{r-cl}}_{c,a,t}$	$\in \mathbb{R}_{\geq 0}$	outgoing pressure in closed mode for $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$p^{\mathrm{l-cf}}_{c,a,t}$	$\in \mathbb{R}_{\geq 0}$	incoming pressure in configuration $c \in \mathcal{C}_a$ of $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$p^{\mathrm{r-cf}}_{c,a,t}$	$\in \mathbb{R}_{\geq 0}$	outgoing pressure in configuration $c \in \mathcal{C}_a$ of $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$q^{\mathrm{cf}}_{c,a,t}$	$\in \mathbb{R}$	flow in configuration $c \in \mathcal{C}_a$ of $a \in \mathcal{A}^{\mathrm{cs}}$	kg/s
$m^{\mathrm{om}}_{o,t}$	$\in \{0,1\}$	selection of mode $o \in \mathcal{O}$	1
$m^{\mathrm{fd}}_{f,t}$	$\in \{0,1\}$	selection of flow direction $f \in \mathcal{F}$	1
$\sigma^{p+}_{v,t}$	$\in \mathbb{R}_{\geq 0}$	positive pressure slack for boundary node $v \in \mathcal{V}^{\mathrm{b}}$	bar
$\sigma^{p-}_{v,t}$	$\in \mathbb{R}_{\geq 0}$	negative pressure slack for boundary node $v \in \mathcal{V}^{\mathrm{b}}$	bar
$\sigma^{d+}_{v,t}$	$\in \mathbb{R}_{\geq 0}$	positive flow slack for boundary node $v \in \mathcal{V}^{\mathrm{b}}$	kg/s
$\sigma^{d-}_{v,t}$	$\in \mathbb{R}_{\geq 0}$	negative flow slack for boundary node $v \in \mathcal{V}^{\mathrm{b}}$	kg/s
δ^{om}_{t}	$\in \{0,1\}$	operation mode change	1
$\delta^{\mathrm{rg}}_{a,t}$	$\in \{0,1\}$	mode change for regulator $a \in \mathcal{A}^{\mathrm{rg}}$	1
$\delta^{\mathrm{us}}_{u,t}$	$\in \{0,1\}$	start of compressor unit $u \in \mathcal{U}_a$ for $a \in \mathcal{A}^{\mathrm{cs}}$	1
$\delta^{\mathrm{rg-pl}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	change of incoming pressure of active $a \in \mathcal{A}^{\mathrm{rg}}$	bar
$\delta^{\mathrm{rg-pr}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	change of outgoing pressure of active $a \in \mathcal{A}^{\mathrm{rg}}$	bar
$\delta^{\mathrm{rg-q}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	change of flow over of active $a \in \mathcal{A}^{\mathrm{rg}}$	kg/s
$\delta^{\mathrm{cs-pl}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	change of incoming pressure of active $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$\delta^{\mathrm{cs-pr}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	change of outgoing pressure of active $a \in \mathcal{A}^{\mathrm{cs}}$	bar
$\delta^{\mathrm{cs-q}}_{a,t}$	$\in \mathbb{R}_{\geq 0}$	change of flow over of active $a \in \mathcal{A}^{\mathrm{cs}}$	kg/s

Table A.1: List of all used variables, specifying their domain, meaning and unit. All variables are defined for $t \in \mathcal{T}$, not taking into account the given values for the initial state, see Section 2.8. Note that $1 \, \mathrm{bar} = 10^5 \, \mathrm{Pa}$.
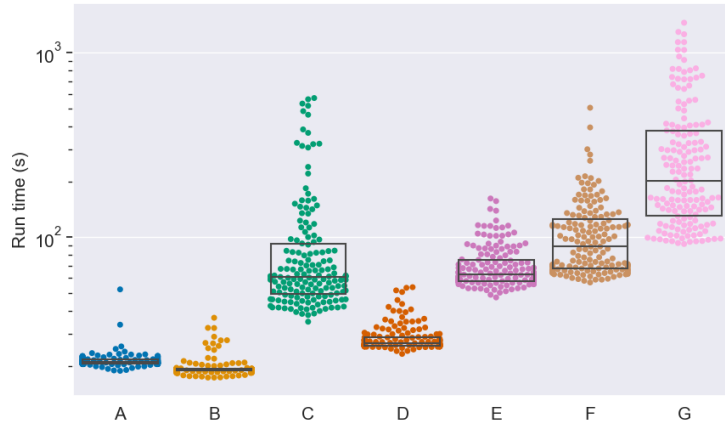
Figure A.1: Run times of instances with 24 time steps, sorted by station. One dot represents one instance, however dots may overlap when too many have similar run time. In addition we draw a typical whiskerless box plot. Therefore the lines represent the 25th percentile, the median and the 75th percentile.
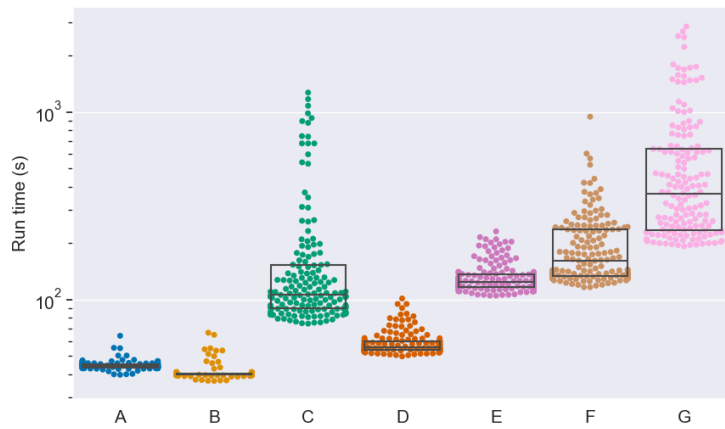

Figure A.2: Run times of instances with 48 time steps, sorted by station. One dot represents one instance, however dots may overlap when too many have similar run time. In addition we draw a typical whiskerless box plot. Therefore the lines represent the 25th percentile, the median and the 75th percentile.
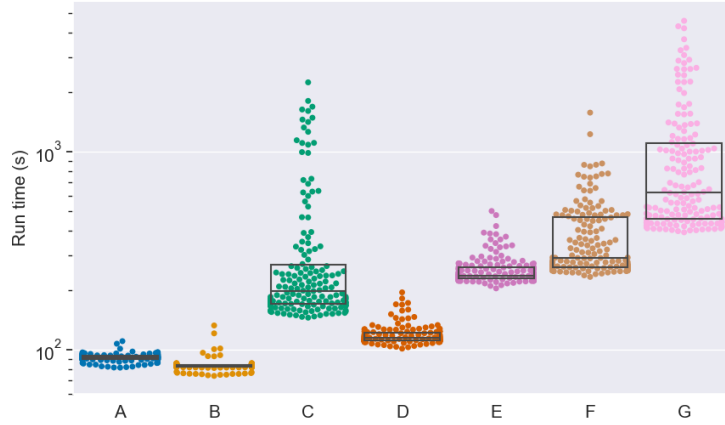
Figure A.3: Run times of instances with 96 time steps, sorted by station. One dot represents one instance, however dots may overlap when too many have similar run time. In addition we draw a typical whiskerless box plot. Therefore the lines represent the 25th percentile, the median and the 75th percentile.
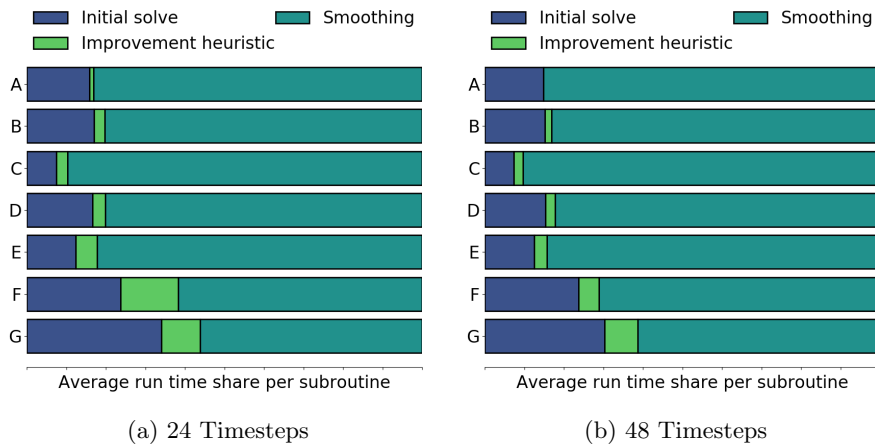

(a) 24 Timesteps

(b) 48 Timesteps

Figure A.4: Portion of run time spend in the single subroutines of Algorithm 1 displayed for each of the single stations. Figure A.4a represents the results for 24 time steps, Figure A.4b the results for 48 time steps.