

A CONSENSUS-BASED ALTERNATING DIRECTION METHOD FOR MIXED-INTEGER AND PDE-CONSTRAINED GAS TRANSPORT PROBLEMS

RICHARD KRUG¹, GÜNTER LEUGERING¹, ALEXANDER MARTIN¹,
MARTIN SCHMIDT², DIETER WENINGER¹

ABSTRACT. We consider dynamic gas transport optimization problems, which lead to large-scale and nonconvex mixed-integer nonlinear optimization problems (MINLPs) on graphs. Usually, the resulting instances are too challenging to be solved by state-of-the-art MINLP solvers. In this paper, we use graph decompositions to obtain multiple optimization problems on smaller blocks, which can be solved in parallel and which may result in simpler classes of optimization problems since not every block necessarily contains mixed-integer or nonlinear aspects. For achieving feasibility at the interfaces of the several blocks, we employ a tailored consensus-based penalty alternating direction method. Our numerical results show that such decomposition techniques can outperform the baseline approach of just solving the overall MINLP from scratch. However, a complete answer to the question of how to decompose MINLPs on graphs in dependence of the given model is still an open topic for future research.

1. INTRODUCTION

Many countries around the world strive towards a carbon-free energy supply but this energy turnaround is not yet accomplished. Hence, modern societies and economies still need to use energy carriers with a significant carbon footprint such as coal or natural gas, where the latter is often characterized as some kind of a bridging technology. To achieve their energy and climate goals, these societies and economies need to use resources like natural gas as efficiently as possible—an aspect that even got more important due to Russia’s 2022 attack on Ukraine and the corresponding energy crisis in Europe. Obviously, not only the energy-saving usage of natural gas is important but also a highly efficient operation of the transport infrastructure is necessary. In this paper, we consider the latter aspect and derive novel algorithms to dynamically control gas transport networks.

From a mathematical point of view, this leads to very challenging optimization problems due to at least three reasons. First, appropriate models of gas physics are highly nonlinear. Second, the control of certain elements in gas transport networks such as valves or compressor stations introduce mixed-integer aspects. Third, dynamic effects lead to large-scale problems after the discretization of the considered time horizon. This all results in large-scale and nonconvex mixed-integer nonlinear models (MINLPs) that need to be solved on large transport networks if real-world problems are considered. Usually, the resulting instances are beyond the scope of what can be tackled even with today’s most sophisticated MINLP solvers.

Our contribution is to exploit the special structure of the studied MINLPs that is induced by the transport networks, which are modeled by graphs. We decompose these graphs so that multiple smaller optimization problems can be solved on the resulting blocks. In addition

Date: October 24, 2022.

2020 Mathematics Subject Classification. 90Cxx, 90C11, 90C35, 90C90.

Key words and phrases. Gas transport networks, Mixed-integer nonlinear optimization, Alternating direction methods, Graph decomposition, Penalty methods.

to the reduced size of the graphs, and depending on the specific graph decomposition, the resulting optimization problems on the smaller blocks may also be in a simpler class of optimization problems because they, e.g., only contain mixed-integer linear or nonlinear but purely continuous models. Of course, the graph decompositions lead to couplings between the different blocks that need to be satisfied to obtain a feasible solution on the overall transport network. We ensure this by a problem-tailored and consensus-based penalty alternating direction method (PADM) that allows for solving the problems on the separate components in parallel in every iteration of the PADM.

The main idea just sketched is motivated by related works on the infinite-dimensional level, where non-overlapping domain decompositions have been studied a lot in the last years [13–15]. In this contribution, however, we focus on finite-dimensional MINLPs obtained by a full discretization as they are also considered in, e.g., [2, 10, 25]. In contrast to such non-overlapping decomposition methods, also overlapping methods have been applied with great success in the recent years; see [23] for an application to power networks, [24] for decentralized schemes for general graph-structured problems, [16] for applications in nonlinear optimal control, [22] for some theoretical foundations for these overlapping methods, and [12] for a recent Julia implementation of the general ideas. At the core of many of these contributions are the key ideas of alternating directions methods of multipliers (ADMMs); see [1] for an overview. In the latter paper, the authors discuss so-called sharing and consensus methods, which we also exploit in our contribution. However, due to the mixed-integer nature of the problems we want to solve, ADMMs are not directly applicable and we thus resort to alternating direction methods (ADMs) that do not rely on duality reasoning, which is not available in our setting. ADMs have been already used to solve gas transport problems; see, e.g., [8, 9]. However, the decomposition of the model used in [9] is not graph- but physics-based and in [8], very specific graph-decompositions are used that we generalize in the present paper. Moreover, both papers tackle the stationary case whereas we now consider a time-dependent model based on the partial differential equations (PDEs) given by the Euler equations.

The remainder of this paper is structured as follows. In Section 2, we present the PDE-constrained optimization model that we consider and also discuss the used discretization of the Euler equations to obtain a finite-dimensional MINLP. Afterward, we review the basics of penalty alternating direction methods in Section 3 and then show in Section 4 how we can reformulate our original model so that it fits into the format to which the PADM can be applied. The numerical results are presented and discussed in Section 5 and the paper closes with a brief discussion of open research questions for future work in Section 6.

2. GAS NETWORK MODELING

We consider a fixed and finite time horizon $[0, T]$ in which we control a gas network. The network itself is given as a directed and weakly connected graph (V, A) . The nodes V consist of entries V_+ , exits V_- , and inner nodes V_0 , i.e.,

$$V = V_+ \cup V_- \cup V_0,$$

while the arcs are made up of pipes, short pipes, valves, compressor stations, and control valves:

$$A = A_p \cup A_{sp} \cup A_v \cup A_{cs} \cup A_{cv}.$$

In what follows, we introduce models for the separate gas network components and state possible objective functions.

2.1. Pipes. The one-dimensional semi-linear Euler equations that we use to model the gas flow in a pipe $a \in A_p$ are given by

$$\partial_t p_a + \frac{c^2}{A_a} \partial_x q_a = 0, \quad (1a)$$

$$\partial_t q_a + A_a \partial_x p_a = -\frac{\lambda_a c^2}{2D_a A_a} \frac{q_a |q_a|}{p_a} - \frac{g A_a h'_a}{c^2} p_a \quad (1b)$$

on $(0, T) \times (0, L)$; see, e.g., [11]. They consist of the continuity equation (1a) and the momentum equation (1b). Here and in what follows, p and q model the pressure and the mass flow of gas, A , D , L , and h' describe the cross-sectional area, the diameter, the length, and the slope of the pipe. Furthermore, c and g are the speed of sound in natural gas and the gravitational acceleration. Note that pressure p and mass flow q are functions of time and space, i.e., $p_a = p_a(t, x)$ and $q_a = q_a(t, x)$. However, we omit these dependencies in (1) for the ease of better reading. In what follows, the notation $p_{a,u}(t)$ stands for the pressure in the pipe a at time t directly at the interface to node u .

There is a multitude of ways to model the friction coefficient λ ; see, e.g., [6, 21]. For the modeling in this paper, we use the flow-independent law of Nikuradse, i.e.,

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

where the parameter k_a describes the roughness of the pipe's inner wall.

Finally, we assume that the mass flow inside of every pipe and every other element of the network is uniformly (over time) bounded from below and above, i.e.,

$$q_a(t, x) \in [q_a^-, q_a^+] \quad (2)$$

holds on $(0, T) \times (0, L)$. The same holds true for pressures, i.e.,

$$p_a(t, x) \in [p_a^-, p_a^+] \quad (3)$$

holds on $(0, T) \times (0, L)$.

2.2. Short Pipes. A short pipe $a = (u, v) \in A_{sp}$ is an artificial modeling tool to directly connect two nodes so that there is no pressure loss, i.e.,

$$p_{a,u}(t) = p_{a,v}(t), \quad t \in [0, T], \quad (4)$$

and there is only one mass flow variable $q_a(t) = q_{a,u}(t) = q_{a,v}(t)$ for every point in time $t \in T$.

2.3. Valves. At each point in time $t \in [0, T]$, a valve $a = (u, v) \in A_v$ is either open or closed. If it is open, it is modeled as a short pipe. If the valve is closed, gas flow is blocked and the pressures at both incident nodes are decoupled. Thus, we have $p_{a,u}(t) = p_{a,v}(t)$, if the valve a is open and $q_a(t) = 0$, if the valve a is closed. This can be modeled by using a binary variable $o_a(t) \in \{0, 1\}$ that indicates if the valve is open ($o_a(t) = 1$) or closed ($o_a(t) = 0$). The mixed-integer model for valves then reads

$$p_{a,u}(t) \geq p_{a,v}(t) - (p_a^+ - p_a^-)(1 - o_a(t)), \quad (5a)$$

$$p_{a,u}(t) \leq p_{a,v}(t) + (p_a^+ - p_a^-)(1 - o_a(t)), \quad (5b)$$

$$q_a(t) \geq q_a^- o_a(t), \quad (5c)$$

$$q_a(t) \leq q_a^+ o_a(t) \quad (5d)$$

for all $a \in A_v$ and all $t \in [0, T]$.

2.4. Compressor Stations. Similar to valves, compressor stations $a \in A_{cs}$ can be open or closed. If they are open, they can be either active or in bypass mode. Active compressor stations can increase the pressure by a controlled amount $\Delta p_a \in [0, \Delta p_a^+]$, while compressor stations in bypass mode behave like short pipes. The flow direction of a compressor station $a \in A_{cs}$ is fixed, i.e., $q_a^- = 0$. It holds

$$\begin{aligned} p_{a,u}(t) + \Delta p_a(t) &= p_{a,v}(t), && \text{if the compressor station } a \text{ is active,} \\ p_{a,u}(t) &= p_{a,v}(t), && \text{if the compressor station } a \text{ is in bypass mode,} \\ q_a(t) &= 0, && \text{if the compressor station } a \text{ is closed.} \end{aligned}$$

If we assume that the compressor station is always open, we can use the linear compressor station model

$$p_{a,u}(t) + \Delta p_a(t) = p_{a,v}(t), \quad (6)$$

where $\Delta p_a(t) = 0$ corresponds to the bypass mode.

If we want the pressure increase to be bounded $0 < \Delta p_a^- \leq \Delta p_a(t) \leq \Delta p_a^+$ if the compressor station is active, we have to introduce a binary variable $b_a(t) \in \{0, 1\}$. This variable indicates if the compressor station is active ($b_a(t) = 1$) or in bypass mode ($b_a(t) = 0$). The mixed-integer compressor station model then reads

$$\Delta p_a(t) \geq \Delta p_a^- b_a(t), \quad (7a)$$

$$\Delta p_a(t) \leq \Delta p_a^+ b_a(t), \quad (7b)$$

$$p_{a,u}(t) + \Delta p_a(t) = p_{a,v}(t). \quad (7c)$$

We will analyze the computational impact of the different presented compressor models later in our numerical study. Note, however, that there are much more realistic nonlinear compressor station models in the literature; see, e.g., [17, 18]. These highly accurate models are, however, out of the scope of this paper.

2.5. Control Valves. We model control valves $a = (u, v) \in A_{cv}$ similar to compressor stations, but instead of increasing the pressure they can decrease it. This can be modeled by using the compressor station model (6) or (7) with $\Delta p \in [\Delta p^-, 0]$.

2.6. Nodes. At nodes $u \in V$, we have to model pressure continuity, which is given by

$$p_u(t) = p_{a,u}(t) \quad \text{for all } a \in \delta(u) \quad (8)$$

for all $t \in [0, T]$. It states that the node pressure $p_u(t)$ is equal to the adjacent arc pressures $p_{a,u}(t)$ at the interface to the respective node u . Here, we use $\delta(u) = \delta^{\text{in}}(u) \cup \delta^{\text{out}}(u)$, where $\delta^{\text{in}}(u)$ and $\delta^{\text{out}}(u)$ are the sets of in- and outgoing arcs of node u . In addition, the pressure at all nodes are bounded from below and above via

$$p_u(t) \in [p_u^-, p_u^+] \quad (9)$$

for all $t \in [0, T]$.

Furthermore, at nodes, we have to model mass balance via

$$q_u(t) + \sum_{a \in \delta^{\text{in}}(u)} q_{a,u}(t) - \sum_{a \in \delta^{\text{out}}(u)} q_{a,u}(t) = 0 \quad (10)$$

for $u \in V$ and $t \in [0, T]$, where $q_u(t)$ is the mass flow that is supplied or withdrawn from the network at node u . It holds

$$q_u(t) \begin{cases} \geq 0 & \text{for entries } u \in V_+, \\ \leq 0 & \text{for exits } u \in V_-, \\ = 0 & \text{for inner nodes } u \in V_0 \end{cases}$$

and the mass flow at nodes is also bounded from below and above, i.e.,

$$q_u(t) \in [\hat{q}_u^-, \hat{q}_u^+] \quad (11)$$

for all $u \in V$ and $t \in [0, T]$.

2.7. Objective Function. A classic objective function in gas transport optimization is to minimize the operational costs of the compressor stations, i.e., one wants to minimize

$$f_c = \sum_{a \in A_{cs}} \int_0^T \Delta p_a(t) dt. \quad (12)$$

Another possibility is the tracking-type objective function on entries V_+ and exits V_- ,

$$f_t = \sum_{u \in V_+ \cup V_-} \eta \|p_u(T) - \hat{p}_u\|^2 + \theta \|q_u(T) - \hat{q}_u\|^2, \quad (13)$$

for given factors η , θ and target pressures \hat{p} and mass flows \hat{q} .

2.8. Problem Statement. Combining all the parts presented before, we arrive at the optimization problem

$$\begin{aligned} \min \quad & (12) \text{ or } (13) \\ \text{s.t.} \quad & \text{Euler equations: (1),} \\ & \text{short pipe model: (4),} \\ & \text{valve model: (5),} \\ & \text{compressor station and control valve model: (6) or (7),} \\ & \text{pressure continuity equations: (8),} \\ & \text{mass balance equation: (10),} \\ & \text{variable bounds: (2), (3), (9), (11).} \end{aligned} \quad (14)$$

2.9. Discretization. Model (14) is a time-dependent mixed-integer optimization problem that is further constrained by partial differential equations. In this paper, we follow the first-discretize-then-optimize approach. To this end, we have to discretize the Euler equations (1) in space and time. For both dimensions, we apply the implicit Euler method. For time, we use the grid given by the points t_κ , $\kappa = 0, \dots, N_t$, with $t_0 = 0$, $t_{N_t} = T$, and $t_\kappa < t_{\kappa+1}$. We assume that all variables at time point $t_0 = 0$ are fixed as they constitute the initial state of the system. For the spatial discretization, we use the grid given by the points $x_{a,\nu}$, $\nu = 0, \dots, N_x^a$, with $x_{a,0} = 0$, $x_{a,N_x^a} = L_a$, and $x_{a,\nu} < x_{a,\nu+1}$. Note that the spatial grid depends on the specific pipe $a \in A_p$. For the ease of presentation, we consider an equidistant discretization in time and space with a fixed time step

$$\Delta t = T/N_t = t_{\kappa+1} - t_\kappa \quad \text{for all } \kappa = 0, \dots, N_t - 1$$

and spatial step

$$\Delta x_a = L_a/N_x^a = x_{a,\nu+1} - x_{a,\nu} \quad \text{for all } \nu = 0, \dots, N_x^a - 1, a \in A_p.$$

Using this notation, we get the fully discretized set of Euler equations

$$\begin{aligned} \frac{p_a(t_{\kappa+1}, x_{a,\nu+1}) - p_a(t_\kappa, x_{a,\nu+1})}{\Delta t} + \frac{c^2}{A_a} \frac{q_a(t_{\kappa+1}, x_{a,\nu+1}) - q_a(t_{\kappa+1}, x_{a,\nu})}{\Delta x_a} = 0, \quad (15a) \\ \frac{q_a(t_{\kappa+1}, x_{a,\nu+1}) - q_a(t_\kappa, x_{a,\nu+1})}{\Delta t} + A_a \frac{p_a(t_{\kappa+1}, x_{a,\nu+1}) - p_a(t_{\kappa+1}, x_{a,\nu})}{\Delta x_a} \end{aligned}$$

$$+\frac{\lambda_a c^2}{2D_a A_a} \frac{q_a |q_a|}{p_a}(t_{\kappa+1}, x_{a,\nu+1}) + \frac{g A_a h'_a}{c^2} p_a(t_{\kappa+1}, x_{a,\nu+1}) = 0 \quad (15b)$$

for all $a \in A_p$, $\kappa = 0, \dots, N_t$, and $\nu = 0, \dots, N_x^a$. We now replace (1) in Model (14) with (15) and further discretize the remaining algebraic constraints by simply writing them for all time points t_κ with $\kappa = 1, \dots, N_t$. Moreover, we discretize the objective function (12) via the right Riemannian sum

$$f_c = \sum_{a \in A_{cs}} \sum_{\kappa=1}^{N_t} \Delta t \Delta p_a(t_\kappa).$$

Note that $\Delta p_a(t_0)$ is given and thus not part of the sum in the objective function. The tracking-type objective function (13) stays unchanged because it only depends on the time point $t_{N_t} = T$. In this way, we obtain a large-scale but finite-dimensional and structured mixed-integer nonlinear optimization problem. Here, the nonlinearities are all contained in the model of the pipes, namely in the discretized momentum equation (15b).

3. PENALTY ALTERNATING DIRECTION METHOD

In this section, we briefly review penalty alternating direction methods (PADM). We first sketch the original version of the method as introduced in [7] and then present a modified variant for so-called quasi-separable problems, which we will then apply to the discretized gas transport problem in the next section.

3.1. Classic Penalty Alternating Direction Methods. For the moment, let us consider the optimization problem

$$\min_{x,y} f(x,y) \quad (16a)$$

$$\text{s.t. } g(x,y) = 0, \quad (16b)$$

$$x \in X, y \in Y \quad (16c)$$

with $X \subseteq \mathbb{R}^{n_x}$ as well as $Y \subseteq \mathbb{R}^{n_y}$ being non-empty and compact sets and $f : \mathbb{R}^{n_x+n_y} \rightarrow \mathbb{R}$ as well as $g : \mathbb{R}^{n_x+n_y} \rightarrow \mathbb{R}^m$ are continuous functions. Classic PADM's try to compute so-called partial minima of this problem, which are points (x^*, y^*) that satisfy

$$f(x^*, y^*) \leq f(x, y^*) \quad \text{for all } x \in X \text{ with } g(x, y^*) = 0,$$

$$f(x^*, y^*) \leq f(x^*, y) \quad \text{for all } y \in Y \text{ with } g(x^*, y) = 0.$$

For what follows, we introduce the penalized objective function

$$\phi(x, y; \mu) = f(x, y) + \sum_{j=1}^m \mu_j |g_j(x, y)|^2 \quad (17)$$

with penalty parameters $\mu \in \mathbb{R}_{>0}^m$ and the penalty problem

$$\min_{x,y} \phi(x, y; \mu) \quad \text{s.t. } x \in X, y \in Y. \quad (18)$$

Note that we have a separate penalty parameter for every constraint. With this, we can state the PADM, which is formally given in Algorithm 1.

Let us also briefly sketch the convergence properties of Algorithm 1. To this end, we first make the following assumption that has already been mentioned above.

Assumption 1. *The objective function $f : \mathbb{R}^{n_x+n_y} \rightarrow \mathbb{R}$ and the constraint function $g : \mathbb{R}^{n_x+n_y} \rightarrow \mathbb{R}^m$ are continuous and the sets X and Y are non-empty and compact.*

Algorithm 1 Penalty Alternating Direction Method

Require: $x^{0,0} \in X$, $y^{0,0} \in Y$, and penalty parameters $\mu^0 \in \mathbb{R}_{>0}^m$.

- 1: **for** $k = 0, 1, 2, \dots$ **do**
- 2: Set $l = 0$.
- 3: **while** $(x^{k,l}, y^{k,l})$ is not a partial minimum of (18) **do**
- 4: Compute $x^{k,l+1} \in \arg \min_x \{\phi(x, y^{k,l}; \mu^k) : x \in X\}$.
- 5: Compute $y^{k,l+1} \in \arg \min_y \{\phi(x^{k,l+1}, y; \mu^k) : y \in Y\}$.
- 6: Set $l \leftarrow l + 1$.
- 7: **end while**
- 8: Choose new penalty parameters $\mu^{k+1} \geq \mu^k$.
- 9: Set $(x^{k+1,0}, y^{k+1,0}) \leftarrow (x^{k,l}, y^{k,l})$.
- 10: **end for**

With this assumption, we can state the main convergence theorem for Algorithm 1, which is taken from [7].

Theorem 1. *Suppose that Assumption 1 holds and that $\mu_j^k \nearrow \infty$ for all $j = 1, \dots, m$. Moreover, let (x^k, y^k) be a sequence of partial minima of (18) (for $\mu = \mu^k$) generated by Algorithm 1 with $(x^k, y^k) \rightarrow (x^*, y^*)$. Then, there exist $\bar{\mu} \geq 0$ such that (x^*, y^*) is a partial minimizer of the weighted ℓ_1 feasibility measure*

$$\chi_{\bar{\mu}}(x, y) := \sum_{j=1}^m \bar{\mu}_j |g_j(x, y)|.$$

If, in addition, (x^*, y^*) is feasible for the original problem (16), the following holds:

- (a) If f is continuous, then (x^*, y^*) is a partial minimum of (16).
- (b) If f is continuously differentiable, then (x^*, y^*) is a stationary point of (16).
- (c) If f is continuously differentiable and f and the feasible set of Problem (16) are convex, then (x^*, y^*) is a global optimum of (16).

3.2. A Penalty Alternating Direction Method for Quasi-Separable Problems. We now present a reformulation for problems of the form

$$\min_{x^1, \dots, x^\ell, z} \sum_{i=1}^{\ell} f_i(x^i) \tag{19a}$$

$$\text{s.t. } (x^i, z) \in X_i \quad \text{for } i = 1, \dots, \ell, \tag{19b}$$

$$z \in Z \tag{19c}$$

with $X_i \subseteq \mathbb{R}^{n_i} \times \mathbb{R}^{n_z}$, $f_i : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$ for $i = 1, \dots, \ell$, and a convex set $Z \subseteq \mathbb{R}^{n_z}$. The problem consists of ℓ constraint blocks X_i that are connected via the so-called linking variables $z \in Z$. In the next section, we will see that we can put the fully discretized gas transport problem (14) in this format in order to then apply a tailored PADM to solve it. Informally speaking, we call such problems “quasi-separable” if the dimension of the coupling constraints (19c) is not too large.

We reformulate problem (19) by introducing a copy z^i of z for each block, which yields the equivalent problem

$$\min_{x^1, \dots, x^\ell, z^1, \dots, z^\ell, z} \sum_{i=1}^{\ell} f_i(x^i) \quad (20a)$$

$$\text{s.t. } z^i = z \quad \text{for } i = 1, \dots, \ell, \quad (20b)$$

$$(x^i, z^i) \in X_i \quad \text{for } i = 1, \dots, \ell, \quad (20c)$$

$$z^i \in Z \quad \text{for } i = 1, \dots, \ell, \quad (20d)$$

$$z \in Z. \quad (20e)$$

Next, we apply Algorithm 1 to Problem (20) with $x = (x^1, z^1, \dots, x^\ell, z^\ell)$, $y = z$, $X = (X_1 \cap (\mathbb{R}^{n_1} \times Z)) \times \dots \times (X_\ell \cap (\mathbb{R}^{n_\ell} \times Z))$, and $Y = Z$. Moreover, the constraints g are identified with the equality constraints (20b). The optimization problem in Step 4 of Algorithm 1 is then given by

$$\min_{x^1, \dots, x^\ell, z^1, \dots, z^\ell} \sum_{i=1}^{\ell} \left(f_i(x^i) + \sum_{j=1}^{n_z} \mu_j^i |z_j^i - (z_j)^{k,l}|^2 \right)$$

$$\text{s.t. } (x^i, z^i) \in X_i \quad \text{for } i = 1, \dots, \ell,$$

$$z^i \in Z \quad \text{for } i = 1, \dots, \ell.$$

This problem is completely separable. Therefore, we can instead solve

$$\min_{x^i, z^i} f_i(x^i) + \sum_{j=1}^{n_z} \mu_j^i |z_j^i - (z_j)^{k,l}|^2 \quad (21a)$$

$$\text{s.t. } (x^i, z^i) \in X_i, \quad (21b)$$

$$z^i \in Z \quad (21c)$$

for all blocks $i = 1, \dots, \ell$, which can be done in parallel. The optimization problem in Step 5 of Algorithm 1 reads

$$\min_{z \in Z} \sum_{i=1}^{\ell} \sum_{j=1}^{n_z} \mu_j^i \left| (z_j^i)^{l+1} - z_j \right|^2.$$

The solution for this so-called consensus problem is the weighted arithmetic mean

$$z_j = \frac{1}{\sum_{i=1}^{\ell} \mu_j^i} \sum_{i=1}^{\ell} \mu_j^i (z_j^i)^{l+1}$$

for all $j = 1, \dots, n_z$, since Z is convex and $(z^i)_k^{l+1} \in Z$ for all $i = 1, \dots, \ell$ because of (20d); see, e.g., [1].

4. PADM TAILORED REFORMULATION OF THE GAS NETWORK PROBLEM

In this section, we describe how we can state the gas transport problem introduced in Section 2 in the format derived in the last section to then apply the PADM.

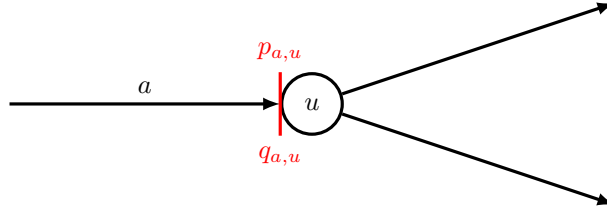


FIGURE 1. The vertical red line represents a possible decomposition at node u that minimizes the number of linking variables.

4.1. Decomposition of Gas Networks. To efficiently apply the PADM to the discretized version of Problem (14), we need to identify a quasi-separable structure as in (19). To this end, we can utilize the fact that we model the problem on a graph. As each variable can be associated with a certain location in this graph it follows that decomposing the graph yields corresponding blocks (indexed with i) of variables with respective constraint sets X_i . If the network is decomposed in a way that the resulting sub-networks are only coupled in a sparse way, then the number of linking variables z stays small and we have found the desired quasi-separable structure.

The question is still open on how to decompose the graph. One might assume that it is a good idea to decompose a network by taking out some selected nodes. This, however, significantly increases the number of linking variables if the degree of the node is larger than two as the pressures and mass flows of all adjacent arcs are connected across the node by the pressure continuity equation (8) and the mass balance equation (10). This undesired aspect can be avoided by splitting the network at points that lie on arcs instead of nodes because the aforementioned equations “stay intact”. To still assign each arc to exactly one sub-network, we decide to only split an arc at the interface to its start or end node. Figure 1 depicts a possible decomposition at the interface to a node of degree three. The only linking variables in this situation are $p_{a,u}$ and $q_{a,u}$ for all time points t_κ .

More precisely, we decompose the graph (V, A) into blocks (V_i, A_i) for $i = 1, \dots, \ell$ with

$$V = \bigcup_{i=1}^{\ell} V_i, \quad A = \bigcup_{i=1}^{\ell} A_i$$

and

$$\begin{aligned} V_i \cap V_j &= \emptyset \quad \text{for } i, j = 1, \dots, \ell, \quad i \neq j, \\ A_i \cap A_j &= \emptyset \quad \text{for } i, j = 1, \dots, \ell, \quad i \neq j. \end{aligned}$$

Thus, each node $u \in V$ and each arc $a \in A$ belongs to exactly one block (V_i, A_i) . Note that a block (V_i, A_i) itself is not a graph, as $a = (u, v) \in A_i$ does not imply $u \in V_i$ and $v \in V_i$. Vice versa, it can happen that $a \notin A_i$ for some arc $a \in \delta^{\text{in}}(u) \cup \delta^{\text{out}}(u)$ with $u \in V_i$. These cases appear exactly at the interfaces of the blocks. We define the interfaces of a single block $i \in \{1, \dots, \ell\}$ as

$$I_i := \{(u, a) \in V_i \times A : a \in \delta(u) \text{ and } a \notin A_i\} \cup \{(u, a) \in V \times A_i : a \in \delta(u) \text{ and } u \notin V_i\}.$$

The set of all interfaces is then given by

$$I := \bigcup_{i=1}^{\ell} I_i.$$

We now use this graph decomposition for gas networks. The constraint system of Problem (14) can then be written as

- Euler equations: (1) for $A_p \cap A_i$,
- short pipe model: (4) for $A_{sp} \cap A_i$,
- valve model: (5) for $A_v \cap A_i$,
- compressor station and control valve model: (6) or (7) for $(A_{cs} \cup A_{cv}) \cap A_i$,
- pressure continuity equations: (8) for V_i ,
- mass balance equation: (10) for V_i ,
- variable bounds: (2), (3), (9), (11) for (V_i, A_i) .

to fit to the constraints (20c) and (20d) for all $i = 1, \dots, \ell$. The objectives (12) and (13) can be rewritten as

$$f_c = \sum_{i=1}^{\ell} \sum_{a \in A_{cs} \cap A_i} \int_0^T \Delta p_a(t) dt$$

and

$$f_t = \sum_{i=1}^{\ell} \sum_{u \in (V_+ \cup V_-) \cap V_i} \eta \|p_u(T) - \hat{p}_u\|^2 + \theta \|q_u(T) - \hat{q}_u\|^2$$

to fit into the form of (20a). The linking constraints (20b) read

$$\begin{aligned} p_{a,u}^i &= p_{a,u} \quad \text{for } (u, a) \in I_i, \quad i = 1, \dots, \ell, \\ q_{a,u}^i &= q_{a,u} \quad \text{for } (u, a) \in I_i, \quad i = 1, \dots, \ell. \end{aligned}$$

Finally, the Constraints (20e) of the consensus problem are given by

$$\begin{aligned} p_{a,u} &\in [p_a^-, p_a^+] \quad \text{for } (u, a) \in I, \\ q_{a,u} &\in [q_a^-, q_a^+] \quad \text{for } (u, a) \in I. \end{aligned}$$

It is easy to see that the resulting optimization problem satisfies Assumption 1 if the original gas network problem (14) is feasible. The objective function and the constraint functions are all continuous. The blocks X and Y are non-empty because a feasible point of (14) can directly be transformed into a feasible point of the reformulated problem. Compactness follows from the variable bounds as well as the constraints in (14). Therefore, the convergence result of Theorem 1 can still be applied. Note, however, that the optimization problems in Step 4 and Step 5 of Algorithm 1 need to be solved to global optimality. For general nonconvex NLPs or MINLPs, this can be a challenging task in practice. Nevertheless, even if we do not solve all sub-problems to global optimality, we can use Algorithm 1 to compute feasible solutions of good quality if the algorithm converges. We will see in Section 5 that this is almost always the case for the instances that we test in our numerical experiments.

Remark 1. (1) *Note that there are multiple strategies to decompose a gas network. If more sub-networks are created, the number of linking variables increases, which potentially makes it harder to find a point that is feasible at all the sub-networks' interfaces. This can lead to a higher number of iterations required by Algorithm 1 and, therefore, longer computation times. However, multiple smaller problems may be easier to solve than a few larger ones. This trade-off needs to be considered when choosing the number of sub-networks.*

- (2) *Another relevant question is which properties the sub-networks should have for Algorithm 1 to converge quickly. One of the approaches that we later test is to separate the active elements of the network, i.e., valves, compressor stations, and control valves, from the passive pipe network. The problems on the resulting sub-networks then contain either discrete variables and no nonlinearities or no discrete variables but nonlinear constraints. Therefore, these problems are MIPs or NLPs but not MINLPs anymore, which are usually much harder to solve. One drawback of this approach is that the sub-networks containing active elements are rather small compared to larger passive sub-networks, which may lead to many iterations that are required to obtain points that are feasible at the blocks' interfaces. A different approach is to create problems on sub-networks that are roughly equally hard to solve. Then, the fact that the sub-problems (21) can be solved in parallel can be utilized most effectively. We will explore these approaches further in multiple case studies in Section 5.*

4.2. Penalty Update Strategies. Theorem 1 requires the penalty parameters μ to go to infinity for Algorithm 1 to converge. This is ensured by updating μ in Step 8 of the algorithm. However, it is not obvious what a good update strategy for these parameters is. On the one hand, large μ can lead to numerical instability and worse objective values as the penalty term will dominate the original objective function in (17). On the other hand, slowly growing μ may lead to slow convergence rates because the feasibility violations at the interfaces of the blocks are not penalized strong enough. Therefore, it is reasonable to increase those μ_j^i stronger for which $|(z_j^i)^{l+1} - z_j|^2$ is large while increasing those μ_j^i with small $|(z_j^i)^{l+1} - z_j|^2$ only a little.

While it is theoretically possible to use a separate penalty parameter μ_j^i for each linking variable z_j^i of block i , it might be a good idea to use a combined parameter μ for multiple variables z_j^i of the same block that represent similar quantities. In the case of gas networks, the linking variables consist of pressure values $p_{a,u}^i$ and mass flow values $q_{a,u}^i$ at different time points and different locations in the network. Our preliminary numerical tests showed that having different penalty parameters for different pressure variables (or flow variables, respectively) can lead to strange and unintuitive controls in a block. To avoid this, it therefore makes sense to introduce a single penalty parameter μ_p^i for the pressures and a single penalty parameter μ_q^i for the mass flows of each block i . Note that for different blocks we still use different penalty parameters as these blocks should be clearly separated from each other while only exchanging information through the linking variables at their interfaces.

The combination of these ideas leads to the following update formulas for the penalty parameters in Step 8 of Algorithm 1:

$$\begin{aligned} (\mu_p^i)_{k+1} &= \left(1 + \omega \mu \frac{m_{p,i}}{\max_{j=1}^{\ell} m_{p,j}} \right) (\mu_p^i)_k, \quad i = 1, \dots, \ell, \\ (\mu_q^i)_{k+1} &= \left(1 + \omega \mu \frac{m_{q,i}}{\max_{j=1}^{\ell} m_{q,j}} \right) (\mu_q^i)_k, \quad i = 1, \dots, \ell, \end{aligned}$$

with

$$\begin{aligned} m_{p,i} &:= \max_{(u,a) \in I_i, \kappa=0, \dots, N_t} |p_{a,u}^i(t_\kappa) - p_{a,u}(t_\kappa)|^2, \quad i = 1, \dots, \ell, \\ m_{q,i} &:= \max_{(u,a) \in I_i, \kappa=0, \dots, N_t} |q_{a,u}^i(t_\kappa) - q_{a,u}(t_\kappa)|^2, \quad i = 1, \dots, \ell, \end{aligned}$$

where we omitted the iteration indices k and l on the pressure and mass flow variables for the ease of presentation. Finally, $\omega_\mu > 0$ serves as a scaling factor.

5. NUMERICAL RESULTS

In this section, we apply the PADM for quasi-separable problems to transient gas network optimization. To this end, we implemented the method in Python 3.8. We use GAMS 36.2.0 [3] to model the optimization problems, which we then solve using the solver CPLEX 20.1.0.1 [5] for LPs and MIPs and the solver KNITRO 12.4.0 [4] for NLPs and MINLPs. All computations were done on nodes of the HPC cluster Woody at FAU Erlangen-Nürnberg with Xeon E3-1240 v6 CPUs on 4 cores, 3.7 GHz, and 32 GB RAM. For the discretization, we choose a time step size of $\Delta t = 3600 \text{ s} = 1 \text{ h}$ and a spatial step size of $\Delta x = 5000 \text{ m}$. For each problem, we set a time limit of 1000 s.

To speed up the convergence of Algorithm 1, we limited the number of ADM loops (Steps 3–7) to 5. Additionally, we stop the ADM loop if the progress of two consecutive iterations is smaller than a given threshold $\varepsilon \geq 0$, i.e., if

$$\|z^{i,k,l+1} - z^{i,k,l}\|_\infty \leq \varepsilon \quad \text{for } i = 1, \dots, \ell.$$

In our case, we choose $\varepsilon := 0.01$ (which is either bar or kg s^{-1}). We consider a solution to be feasible if all copies z^i of the linking variables z are close enough to z , i.e., if

$$\|z^i - z\|_\infty \leq \delta \quad \text{for } i = 1, \dots, \ell$$

holds with $\delta := 0.1$ (again in bar or kg s^{-1}).

We initialize all penalty parameters μ with $\Delta t/T$. In this way, a different time discretization does not change the overall weight of the penalty term in the objective function. For a similar reason, we also divide the cost objective (12) by T . For the tracking-type objective (13), we use the factors $\eta = 1 \text{ bar}^{-2}$ and $\theta = 1 \text{ s}^2 \text{ kg}^{-2}$. The penalty scaling factor is set to $\omega_\mu = 2$. If a penalty parameter μ reaches a value of 10^9 , all penalty parameters get re-scaled by a factor of 10^{-6} ; see, e.g., [19], where a similar re-scaling was done with the help of sigmoidal functions.

5.1. Transient Gas Network Data. The gas network instances for our numerical experiments are taken from the GasLib [20]. It contains network data as well as stationary scenarios for these networks. We use this information to create transient scenarios on a time horizon of $T = 24 \text{ h}$. For each network, we fix the mass flow q_u of each exit node $u \in V_-$ and the pressure p_u for each entry node $u \in V_+$. For the mass flow at an exit node at $t = 0$ we use the value given in the stationary scenario. Then, we apply a vertically shifted sine curve with period T and an amplitude of 0.1 to this initial value to receive the mass-flow curve for the complete time horizon:

$$q_u(t) = \left(1 + 0.1 \sin\left(t \frac{2\pi}{T}\right)\right) q_u(0), \quad u \in V_-, t \in [0, T].$$

The pressure at the entry nodes stays constant over the time horizon. For the first three test networks we consider, namely GasLib-11, GasLib-24, and GasLib-40, which come with only one stationary scenario, we choose these values manually. For the last instance, GasLib-134, which represents the gas network of Greece, however, there are 1234 different stationary scenarios given in the GasLib. Here, we first solve all stationary scenarios by using the discretized version of Problem (14) with only one time point $t = 0 = T$. To this end, we use

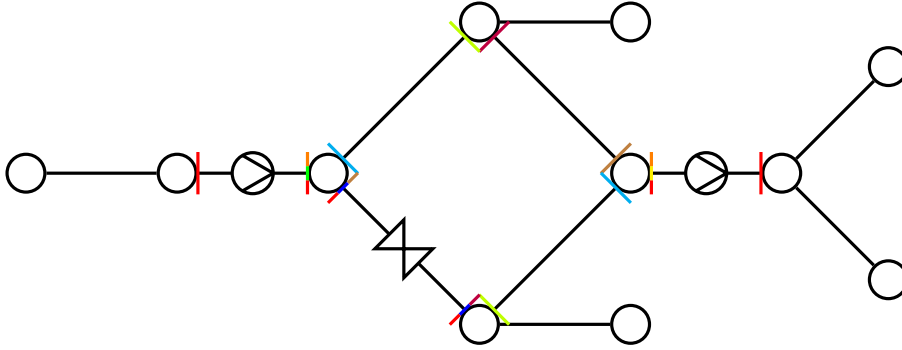


FIGURE 2. Decompositions of GasLib-11: Active (red), Valve (blue), Left Arm (green), Right Arm (yellow), Both Arms (orange), 1 o'clock (purple), 2 o'clock (brown), 4 o'clock (cyan), 5 o'clock (lime)

the stationary Euler equations

$$\begin{aligned}\partial_x q_a &= 0, \\ \partial_x p_a &= -\frac{\lambda_a c^2}{2D_a A_a^2} \frac{q_a |q_a|}{p_a} - \frac{gh'_a}{c^2} p_a,\end{aligned}$$

where all time derivatives vanished, the linear compressor station model (6), and the objective function (12), which now reads

$$f_c = \sum_{a \in A_{cs}} \Delta p_a.$$

Then, we fix the entry pressures to the corresponding values of the obtained stationary solution. Since for two of the stationary problems, the solver reported infeasibility, this results in 1232 transient scenarios for the GasLib-134 network.

For the initial conditions of each scenario we used the solution of the corresponding stationary model. The same holds for the terminal condition that are needed for the tracking-type objective function (13). This stationary solution is also used to initialize all variables including the linking variables.

5.2. GasLib-11. The first test network we consider is the GasLib-11. It contains 11 nodes including 3 entries and 3 exits, 8 pipes, 2 compressor stations, and 1 valve. The total pipe length is 440 km. We fixed the entry pressures to 53 bar, 51 bar, and 52 bar, respectively.

GasLib-11 has a cycle containing the valve in the middle and one arm containing a compressor station on the left and right side, respectively. We apply several decompositions to this network, which we depict through colored cuts in the network graph in Figure 2. If there are multiple cuts at the same location, the corresponding line has multiple colors. Each decomposition gets its own name to be easily identifiable.

The first decomposition, which we apply to every network, is called “Active”, which creates a block for each active element (compressor stations, valves, and control valves) and each connected sub-network of the other elements (pipes, short pipes, and nodes). The resulting blocks of active elements are all LPs (or QPs) and MIPs (or MIQPs), depending on the chosen objective function, and the remaining pipe networks are NLPs, which, in the case of GasLib-11, do not contain cycles. The next decomposition approach, called “Valve”, separates the valve from the rest of the network. The two resulting blocks are an MIP and an NLP or MINLP, depending on the compressor station model and the chosen objective function. Both

TABLE 1. Numerical results for GasLib-11

Decomposition	CS model	Objective model	Time (s)	Objective	#piter	#aiter
No decomp.	LP	CS cost	59.52	0.67423	–	–
Active	LP	CS cost	50.50	1.24971	8	37
Valve	LP	CS cost	7.49	0.66856	1	4
Left Arm	LP	CS cost	695.35	0.80452	3	15
Right Arm	LP	CS cost	808.04	0.92277	4	17
Both Arms	LP	CS cost	246.02	1.11062	5	21
1 o'clock	LP	CS cost	63.28	0.65873	3	13
2 o'clock	LP	CS cost	78.76	0.63744	3	14
4 o'clock	LP	CS cost	491.89	0.76768	4	20
5 o'clock	LP	CS cost	284.16	0.73092	4	16
No decomp.	LP	tracking	14.15	0.00014	–	–
Active	LP	tracking	55.52	1.62423	8	37
Valve	LP	tracking	23.71	0.01442	3	11
Left Arm	LP	tracking	297.16	0.03976	2	7
Right Arm	LP	tracking	536.96	0.00371	2	6
Both Arms	LP	tracking	229.81	0.31451	5	21
1 o'clock	LP	tracking	147.36	0.04843	6	27
2 o'clock	LP	tracking	131.53	0.06025	5	24
4 o'clock	LP	tracking	661.71	1.37777	7	31
5 o'clock	LP	tracking	817.67	1.41519	8	37
No decomp.	MIP	CS cost	258.69	1.04396	–	–
Active	MIP	CS cost	322.47	4.31027	42	205
Valve	MIP	CS cost	444.19	1.03152	1	5
No decomp.	MIP	tracking	1001.46	265.60613	–	–
Active	MIP	tracking	42.77	0.58586	7	32
Valve	MIP	tracking	11.40	0.00294	1	2
Both Arms	MIP	tracking	163.71	0.00802	3	12
1 o'clock	MIP	tracking	63.23	0.00189	2	6
2 o'clock	MIP	tracking	51.34	0.01047	3	11
4 o'clock	MIP	tracking	18.87	0.00013	1	2
5 o'clock	MIP	tracking	19.58	0.00004	1	2

blocks do not contain a cycle. Then, there are the decompositions “Left Arm”, “Right Arm”, and “Both Arms” which separate either one or two arms containing a compressor station from the cycle in the middle of the network. The block containing the cycle is always an MINLP and the separated arm-blocks are NLPs or MINLPs, depending on the compressor station model. Next, we introduce four decompositions that create cycle-free blocks by cutting the cycle at two opposite locations. The position of the first cut in the cycle inspire the names “1 o'clock”, “2 o'clock”, “4 o'clock”, and “5 o'clock”. The block containing the valve is again an MINLP and the other block is again an NLP or MINLP, depending on the compressor station model. Finally, we tried applying no decomposition (“No decomp.”) and, thus, solving the MINLP as a whole to have a baseline comparison for our method applied to the different graph decompositions.

Now, we apply these decompositions to all combinations of compressor station (CS) models and objective functions. The numerical results are given in Table 1. It contains the computation time (Time (s)), the objective value without the penalty term (Objective), the number of penalty iterations (#piter), and the number of ADM-loops (#aiter). If for a decomposition no feasible solution was found within the time limit of 1000s, the corresponding row does not appear in the table. To better understand the given objective values we recall their meaning. The “CS cost” objective value is the mean value over time of the pressure increases Δp in all compressor stations in bar. For the tracking-type objective we use the squared distance of the pressure and mass flow at entry and exit nodes to given target values. Here, we use the factors $\eta = 1 \text{ bar}^{-2}$ and $\theta = 1 \text{ s}^2 \text{ kg}^{-2}$, respectively, for scaling the different terms.

For the LP compressor station model, a solution was found for every decomposition. For the compressor station cost objective, the “Valve” decomposition was by far the fastest with 7.49s, followed by the “Active” decomposition with 50.50s, and “No decomp.” with 59.52s. It even has a better objective value compared to “No decomp.”. Note, however, that these objective values should be read with care as they can be affected by small pressure and mass flow jumps at the interfaces of adjacent blocks due to the tolerance δ . For the tracking type objective, “No decomp.” is the winner with regard to both computation time and objective value. It is, however, rather closely followed by “Valve” with 23.71s compared to 14.15s.

Now, we take a closer look at the MIP compressor station model. Here, the combination with the compressor station cost objective seems to be the hardest problem as there is only a feasible solution for three different cases. The fastest solution comes from “No decomp.” with 258.69s but “Active” with 322.47s and “Valve” with 444.19s are at least comparable. The latter even has a slightly better objective value.

For the tracking-type objective, “No decomp.” reached the time limit and only returned a feasible point with a bad objective value. In contrast, for “Valve” a solution was found in 11.40s. This time, even the decompositions “4 o’clock” and “5 o’clock” found very good solutions in only 18.87s and 19.58s, respectively.

For all cases we can see that the “Active” decomposition has somewhat fast computation times but comparably bad objective values. This could be due to the fact that the active elements that are used to control the network now only have a very small scope, i.e., their respective blocks are very small. The decompositions “Right Arm”, “Left Arm”, and “Both Arms” are not competitive due to their slow solution times.

Next, we take a closer look at the development of linking variables over the iterations of Algorithm 1. To this end, we plot all copies and the consensus of the mass flow variables of the left compressor station for four different penalty iterations in Figure 3. Here, we used the “Active” decomposition with the LP compressor station model and the compressor station cost objective, which took 8 penalty iterations and overall 37 ADM-loops to produce a feasible point. In the figure, one can clearly see the consensus as the weighted arithmetic mean of the three variables and how these variables change until they converge to the same curve.

5.3. GasLib-24. Next, we consider the GasLib-24 network. It contains 24 nodes including 3 entries and 5 exits, 19 pipes, 1 short pipe, 3 compressor stations, 1 control valve, and 1 resistor, which we replace with a short pipe. There are two cycles in the network. The total pipe length is 820.01 km. For this network, we fixed the pressure of the second entry to 49 bar.

We introduce the decompositions as depicted in Figure 4. We, again, use “No decomp.” as well as the decomposition “Active”. Besides this, the decomposition “Halfs” splits the network

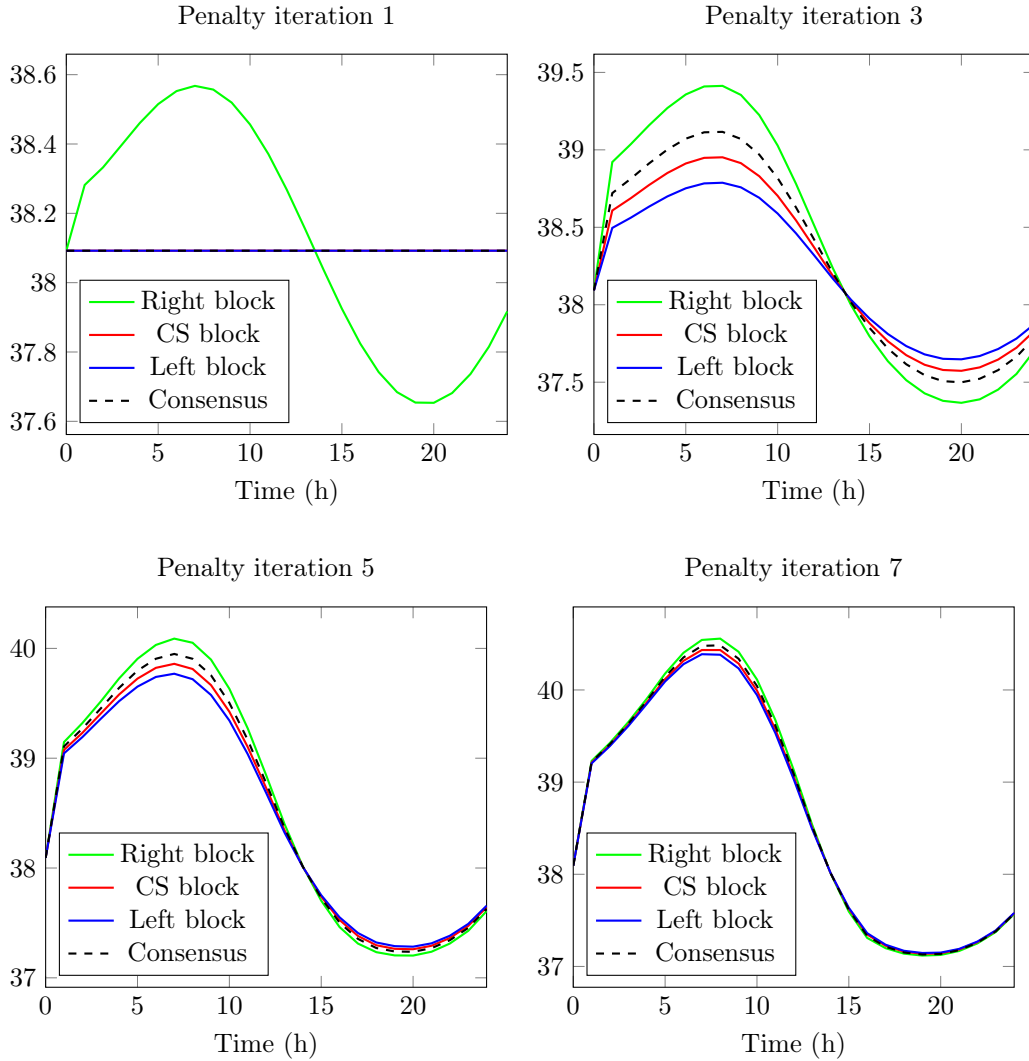


FIGURE 3. The linking variables representing the mass flow q_a (in kg s^{-1}) in the left compressor station of GasLib-11 for the decomposition “Active” at the beginning of different penalty iterations

in two parts of roughly the same size. In the same manner, the decomposition “Thirds” splits the network in three parts of roughly the same size. Finally, the decomposition “Tree” cuts both cycles in a way that the resulting three blocks are trees.

Table 2 contains the numerical results for the aforementioned decompositions. For the LP compressor station model, using “No decomp.” produces solutions very fast. In the case of the compressor station cost objective, the second-fastest option is “Tree” with 72.98 s. For the tracking-type objective, however, the decomposition “Halves” is able to beat the “No decomp.” time of 4.92 s with 2.38 s and an even better objective value. It seems that the initial values of the variables at the interface were already so close to the optimal solution that it took only one iteration.

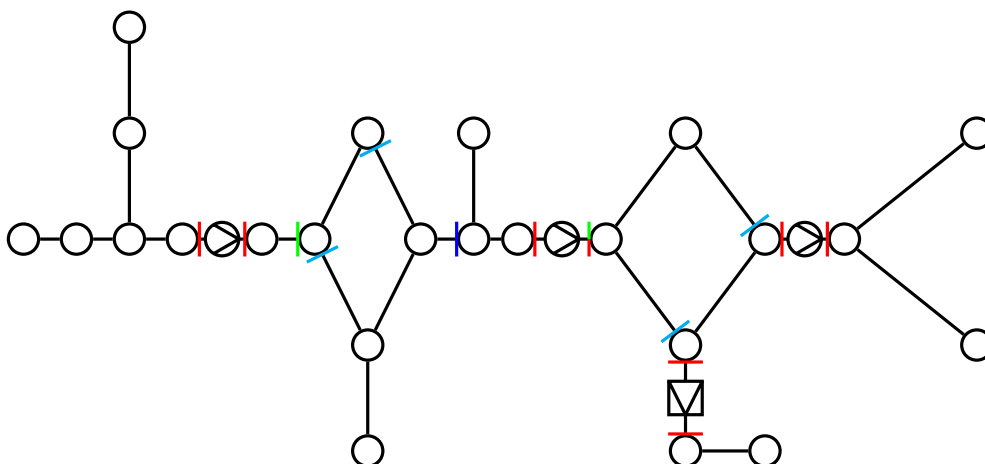


FIGURE 4. Decompositions of GasLib-24: Active (red), Halfs (blue), Thirds (green), Tree (cyan)

TABLE 2. Numerical results for GasLib-24

Decomposition	CS model	Objective model	Time (s)	Objective	#piter	#aiter
No decomp.	LP	CS cost	5.17	0.00077	–	–
Active	LP	CS cost	99.57	0.16778	10	48
Halfs	LP	CS cost	92.44	0.27290	7	34
Thirds	LP	CS cost	112.79	0.22215	10	48
Tree	LP	CS cost	72.98	0.34839	6	30
No decomp.	LP	tracking	4.92	0.00001	–	–
Active	LP	tracking	48.01	0.00979	5	24
Halfs	LP	tracking	2.38	0.00000	1	1
Thirds	LP	tracking	58.77	0.03470	5	21
Tree	LP	tracking	14.61	0.00031	1	5
No decomp.	MIP	CS cost	170.07	0.72917	–	–
Active	MIP	CS cost	121.23	0.00000	11	52
Tree	MIP	CS cost	494.80	0.00000	7	35
No decomp.	MIP	tracking	134.94	0.00001	–	–
Active	MIP	tracking	61.28	0.03131	6	30
Halfs	MIP	tracking	358.06	0.00013	1	1
Thirds	MIP	tracking	698.14	0.01383	5	25
Tree	MIP	tracking	262.78	0.00890	4	16

For the MIP compressor station model with compressor station cost objective, no feasible solutions were found within the time limit for the decompositions “Halfs” and “Thirds”. Here, the “Active” decomposition produces the optimal objective value in the shortest amount of time 121.23s. Even though the “Tree” decomposition takes a much longer time than “No decomp.”, its objective value is also much better. In the case of the MIP compressor station model combined with the tracking-type objective, the “Active” decomposition scores the

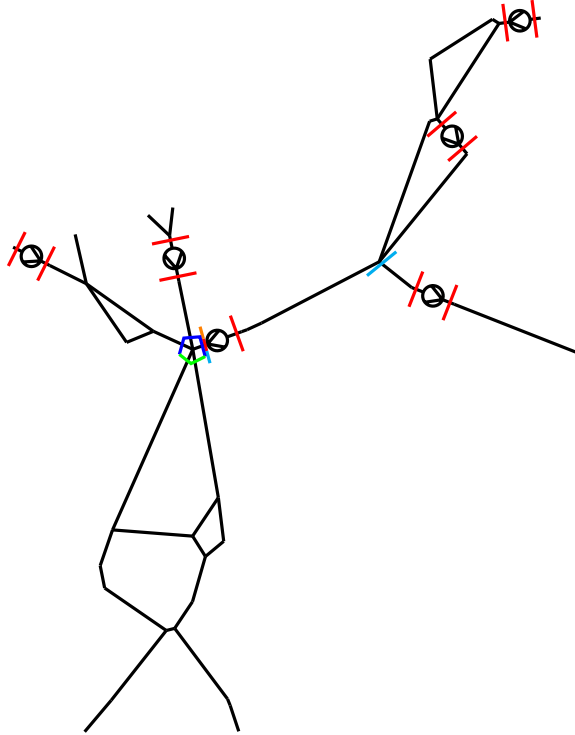


FIGURE 5. Decompositions of GasLib-40: Active (red), All Arms (blue), Lower (green), Right Arm (orange), Right Arm 2 (cyan)

fastest time (61.28s), again. However, the much slower “No decomp.” approach (134.94s) produces a nearly optimal objective value.

5.4. GasLib-40. The last of the three test networks of the GasLib library is GasLib-40. It contains 40 nodes including 3 entries and 29 exits, 39 pipes, and 6 compressor stations. Its total pipe length is 1112.47 km. The network contains multiple cycles. One key point is a node of degree 5, which connects the passive lower part of the network with the three arms that contain the entries and all compressor stations. While the left and upper arm have one compressor station each, the right arm has four of them. All three entry pressures are fixed to 67 bar.

Figure 5 shows our decompositions of GasLib-40. Next to “No decomp.” and “Active”, we introduce four additional decompositions. The first one, “All Arms”, separates the three arms containing compressor stations from the passive lower part of the network. The second decomposition, “Lower”, does something similar but cuts towards the passive lower part instead of the three arms. The third one, “Right Arm”, only separates the right arm from the rest of the network because it is the most complicated one containing the most compressor stations. The last decomposition, “Right Arm 2”, follows the same idea but also adds an additional cut in the right arm to further simplify the resulting blocks.

Since GasLib-40 is already rather large, not all details can be displayed in Figure 5. Hence, we also show the block information of the different decompositions in Table 3. Here, the numbers of nodes, entries, exits, pipes, and compressor stations are given together with the model type of each block, which might still depend on the compressor station model in use.

TABLE 3. Information on the decompositions of GasLib-40

Block Nr.	#nodes	#entries	#exits	#pipes	#cs	Type
No decomp.						
1	40	3	29	39	6	MINLP
Active						
1	3	1	2	2	0	NLP
2-3	1	1	0	0	0	LP
4	22	0	20	25	0	NLP
5	2	0	1	1	0	NLP
6	11	0	6	11	0	NLP
7-12	0	0	0	0	1	LP/MIP
All Arms						
1	4	1	2	3	1	NLP/MINLP
2	6	1	4	6	1	NLP/MINLP
3	14	1	7	12	4	NLP/MINLP
4	16	0	16	18	0	NLP
Lower						
1	15	0	15	18	0	NLP
2	25	3	14	21	6	NLP/MINLP
Right Arm						
1	14	1	7	12	4	NLP/MINLP
2	26	2	22	27	2	NLP/MINLP
Right Arm 2						
1	3	0	2	2	1	NLP/MINLP
2	26	2	22	27	2	NLP/MINLP
3	11	1	5	10	3	NLP/MINLP

TABLE 4. Numerical results for GasLib-40

Decomposition	CS model	Objective model	Time (s)	Objective	#piter	#aiter
No decomp.	LP	CS cost	50.82	5.86715	–	–
Right Arm	LP	CS cost	126.23	6.29847	4	17
Right Arm 2	LP	CS cost	117.32	6.49352	5	24
No decomp.	LP	tracking	111.99	0.09216	–	–
All Arms	LP	tracking	137.50	1.70478	8	36
Lower	LP	tracking	477.95	1.07486	8	39
Right Arm	LP	tracking	90.12	1.17842	3	11
Right Arm 2	LP	tracking	89.63	1.18113	3	11
No decomp.	MIP	CS cost	1004.86	7.49440	–	–
No decomp.	MIP	tracking	1004.77	56.70914	–	–

We show the numerical results for **GasLib-40** in Table 4. For the LP compressor station model together with the compressor station cost function, a feasible solution was found for only three settings. The fastest time of 50.82 s as well as best objective value was achieved by applying “No decomp.”. The other two results for “Right Arm” and “Right Arm 2” took more than twice this time and have slightly worse objective values. For the tracking-type objective, the fastest results were achieved for “Right Arm 2” with 89.63 s and “Right Arm” with 90.12 s. Using “No decomp.” was slower (111.99 s) but resulted in a much better objective value.

In the case of the MIP compressor station model, only “No decomp.” produces solutions. However, the time limit of 1000 s was reached in both cases and only a feasible point with no optimality guarantee is returned. For all four combinations for **GasLib-40**, the decomposition “Active” never resulted in a feasible solution within the time limit. This could be due to the small scope of the active elements and the large size of the remaining passive network. The decompositions “All Arms” and “Lower” also did not perform very well.

5.5. GasLib-134. In our case studies for the test networks **GasLib-11**, **GasLib-24**, and **GasLib-40**, we have seen that applying Algorithm 1 to transient gas network problems can yield a significant benefit w.r.t. solution times with only a slight loss (if at all) regarding the objective value. However, the results can vary strongly and are dependent on a number of factors including the compressor station model, the objective function, the network’s structure, and the chosen decomposition. Therefore, the generalizability of these results is limited, especially since there was only a single transient scenario for each of the networks. Thus, we now apply our decomposition strategies to the real-world gas network of Greece **GasLib-134**. For this network, we have constructed 1232 different transient scenarios based on the stationary scenarios given in the **GasLib**. Each of these scenarios represents a day between the 1st of November 2011 and the 17th of February 2016.

GasLib-134 consists of 134 nodes including 3 entries and 45 exits, 86 pipes, 45 short pipes, 1 compressor station, and 1 control valve. The total pipe length is 1447.02 km. The network is a tree with the compressor station in the north and the control valve in the south east.

Figure 6 shows **GasLib-134** and the decompositions that we use. We again use the already known “No decomp.” and the “Active” decomposition. Additionally, we introduce the “Halves” decomposition which splits the network in two blocks of roughly the same size. Then, there are “CS Arm” and “CV Arm”, which split the network directly after the compressor station in the north or directly before the control valve in the south. The decomposition “Both Arms” combines the two aforementioned ideas. Since **GasLib-134** is a large network, all block details of the proposed decompositions are shown in Table 5. This includes the number of nodes, entries, exits, pipes, short pipes, compressor stations, and control valves as well as the model type of each block.

Figure 7 shows the percentage of instances that can be solved in which amount of time for each decomposition, compressor station model, and objective. For the cases with the LP compressor station model, applying no decomposition is dominating. Consequently, there is not much gain by using a decomposition. However, the picture is completely different for the MIP compressor station model. If the compressor station cost objective is used, the decompositions “Active”, “CV Arm”, and “Both Arms” outperform “No decomp.” for most of the scenarios. For the tracking-type objective, the decompositions “Active” and “Both Arms” produce the best results. Here, “CV Arm” performs the worst.

The two remaining decompositions “Halves” and “CS Arm” only produce mediocre results even though they performed quite good in the LP compressor station model case.

Noteworthy is also that the “Active” decomposition solves the least amount of scenarios within the time limit for the LP compressor station model but the most scenarios for the

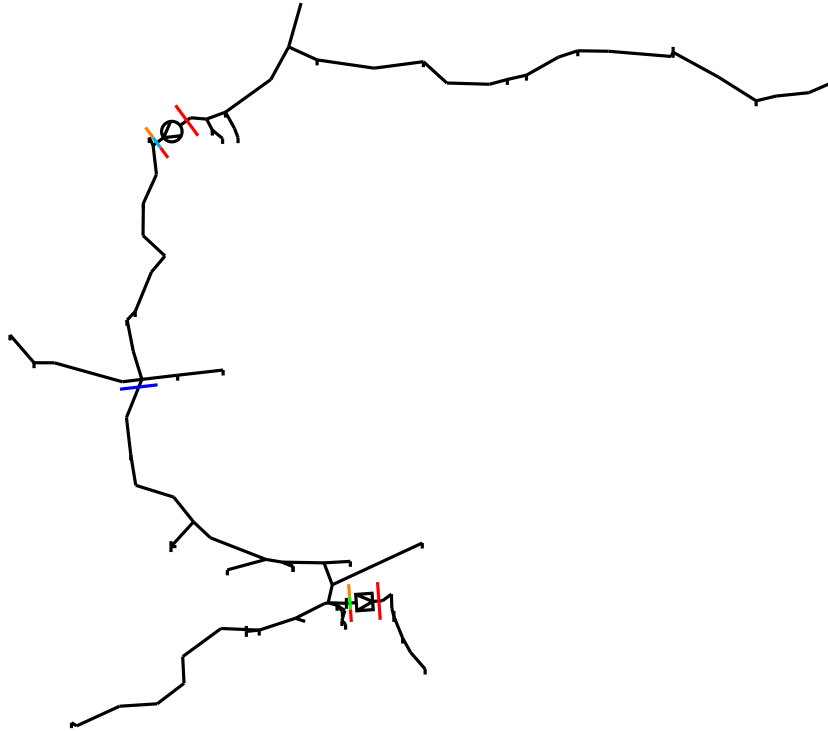


FIGURE 6. Decompositions of GasLib-134: Active (red), Halves (blue), CS Arm (cyan), CV Arm (green), Both Arms (orange)

MIP compressor station model. The corresponding curves in Figure 7 stay almost identical when the compressor station model is changed. Therefore, the compressor station model has almost no effect on the “Active” decomposition but on all other decompositions.

Overall, the two decompositions “Active” and “Both Arms” are the clear winners for the MIP compressor station model. Separating the mixed-integer parts of the active elements from the large middle part of GasLib-134 seems to generate blocks that are not too hard to solve while simultaneously do not require too much time for getting feasible couplings at the blocks’ interfaces.

6. CONCLUSION

We considered the dynamic optimization of gas transport networks. After a full discretization in space and time, the resulting model is a nonconvex MINLP that needs to be solved on large-scale transport networks. This puts a challenging burden even on today’s most sophisticated MINLP solvers. Hence, we proposed a graph decomposition and a tailored consensus-based PADM to compute feasible controls of good quality as fast as possible.

The interpretation of the numerical results is somewhat tricky. On the one hand, we clearly show that using graph decompositions together with a tailored solution method can outperform the baseline method, which just solves the large MINLP from scratch using a state-of-the-art MINLP solver. However, whether this is the case seems to depend on very many aspects such as the chosen objective function, the chosen decomposition, or the chosen model for compressor stations (continuous vs. mixed-integer). This directly leads to the main open research question (for which we think that answering it is out of the scope of this

TABLE 5. Decompositions of GasLib-134

Block Nr.	#nodes	#entries	#exits	#pipes	#sp	#cs	#cv	Type
No decomp.								
1	134	3	45	86	45	1	1	NLP/MINLP
Active								
1	42	2	13	28	13	0	0	NLP
2	81	1	28	52	28	0	0	NLP
3	11	0	4	6	4	0	0	NLP
4	0	0	0	0	0	1	0	LP/MIP
5	0	0	0	0	0	0	1	LP/MIP
Halfs								
1	65	1	23	41	23	0	1	NLP/MINLP
2	69	2	22	45	22	1	0	NLP/MINLP
CV Arm								
1	11	0	4	6	4	0	1	NLP/MINLP
2	123	3	41	80	41	1	0	NLP/MINLP
CS Arm								
1	42	2	13	28	13	1	0	NLP/MINLP
2	92	1	32	58	32	0	1	NLP/MINLP
Both Arms								
1	42	2	13	28	13	1	0	NLP/MINLP
2	11	0	4	6	4	0	1	NLP/MINLP
3	81	1	28	52	28	0	0	NLP

paper): Given a large-scale MINLP defined on a graph, what is the best way to decompose the network so that a tailored decomposition method outperforms the baseline method? Although this paper provides a first step to understanding the relevant effects, we leave a complete answer to this question for future work.

ACKNOWLEDGMENTS

The authors thank the Deutsche Forschungsgemeinschaft for their support within projects A05 and B08 in the Sonderforschungsbereich/Transregio 154 “Mathematical Modelling, Simulation and Optimization using the Example of Gas Networks”.

REFERENCES

- [1] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers.” In: *Foundations and Trends® in Machine Learning* 3.1 (2011), pp. 1–122. DOI: [10.1561/22000000016](https://doi.org/10.1561/22000000016).

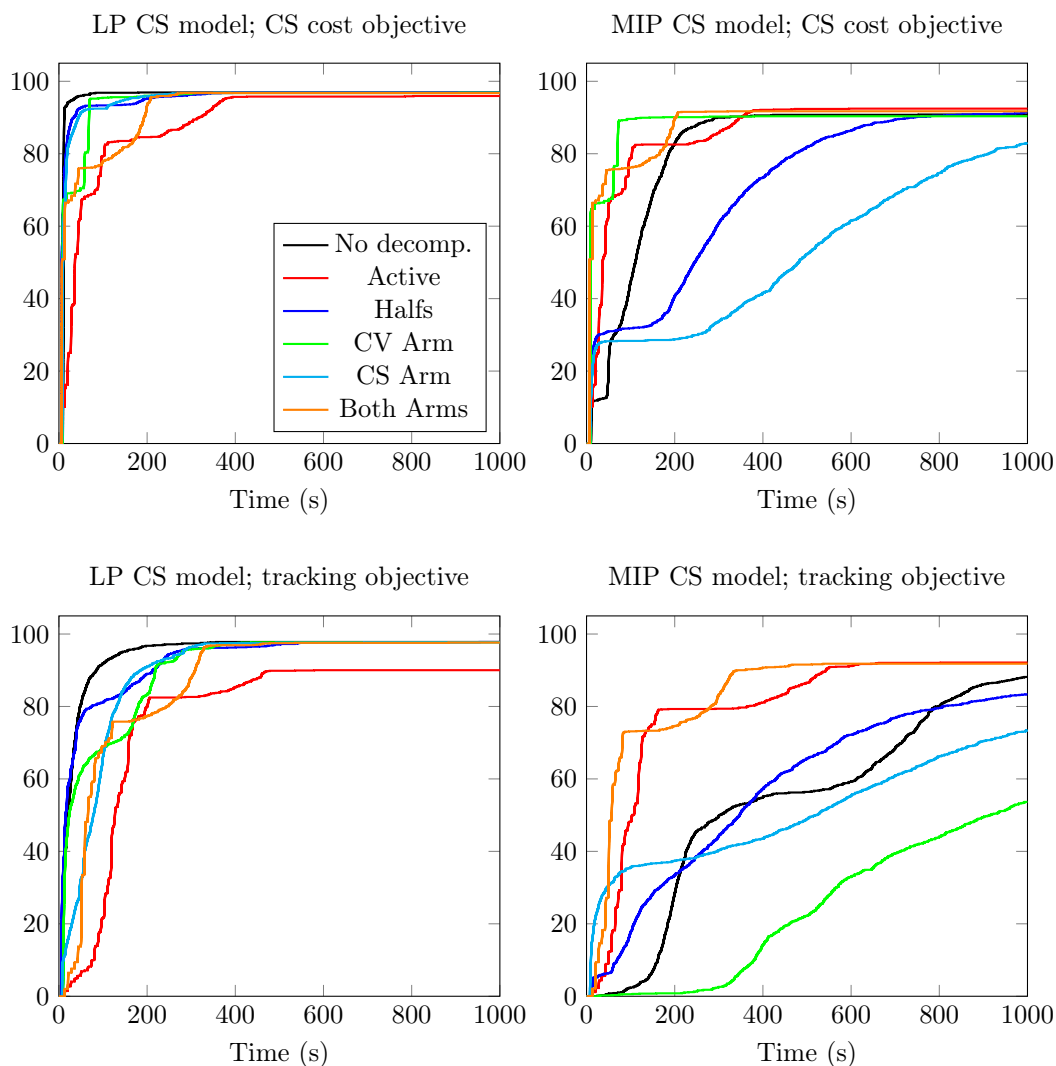


FIGURE 7. Percentage of solved instances of GasLib-134 in the time limit of 1000s for different decompositions, the LP (6) and MIP (7) compressor station (CS) models, and compressor station cost (12) and tracking-type (13) objective functions.

- [2] R. Burlacu, H. Egger, M. Groß, A. Martin, M. E. Pfetsch, L. Schewe, M. Sirvent, and M. Skutella. “Maximizing the storage capacity of gas networks: a global MINLP approach.” In: *Optimization and Engineering 20.2* (2019), pp. 543–573. DOI: [10.1007/s11081-018-9414-5](https://doi.org/10.1007/s11081-018-9414-5).
- [3] M. R. Bussieck and A. Meeraus. “General algebraic modeling system (GAMS).” In: *Modeling languages in mathematical optimization*. Springer, 2004, pp. 137–157. DOI: [10.1007/978-1-4613-0215-5_8](https://doi.org/10.1007/978-1-4613-0215-5_8).

- [4] R. H. Byrd, J. Nocedal, and R. A. Waltz. “KNITRO: An integrated package for nonlinear optimization.” In: *Large Scale Nonlinear Optimization, 35–59, 2006*. Springer Verlag, 2006, pp. 35–59. DOI: [10.1007/0-387-30065-1_4](https://doi.org/10.1007/0-387-30065-1_4).
- [5] CPLEX, IBM ILOG. *V12. 1: User’s Manual for CPLEX*. 53. 2009, p. 157.
- [6] A. Fügenschuh, B. Geißler, R. Gollmer, A. Morsi, M. E. Pfetsch, J. Rövekamp, M. Schmidt, K. Spreckelsen, and M. C. Steinbach. “Physical and technical fundamentals of gas networks.” In: *Evaluating Gas Network Capacities*. Ed. by T. Koch, B. Hiller, M. E. Pfetsch, and L. Schewe. SIAM-MOS series on Optimization. SIAM, 2015. Chap. 2, pp. 17–43. DOI: [10.1137/1.9781611973693.ch2](https://doi.org/10.1137/1.9781611973693.ch2).
- [7] B. Geißler, A. Morsi, L. Schewe, and M. Schmidt. “Penalty Alternating Direction Methods for Mixed-Integer Optimization: A New View on Feasibility Pumps.” In: *SIAM Journal on Optimization* 27.3 (2017), pp. 1611–1636. DOI: [10.1137/16M1069687](https://doi.org/10.1137/16M1069687).
- [8] B. Geißler, A. Morsi, L. Schewe, and M. Schmidt. “Solving Highly Detailed Gas Transport MINLPs: Block Separability and Penalty Alternating Direction Methods.” In: *INFORMS Journal on Computing* 30.2 (2018), pp. 309–323. DOI: [10.1287/ijoc.2017.0780](https://doi.org/10.1287/ijoc.2017.0780).
- [9] B. Geißler, A. Morsi, L. Schewe, and M. Schmidt. “Solving power-constrained gas transportation problems using an MIP-based alternating direction method.” In: *Computers & Chemical Engineering* 82 (2015), pp. 303–317. DOI: [10.1016/j.compchemeng.2015.07.005](https://doi.org/10.1016/j.compchemeng.2015.07.005).
- [10] M. Hahn, S. Leyffer, and V. M. Zavala. *Mixed-integer PDE-constrained optimal control of gas networks*. Tech. rep. 2017. URL: <https://www.mcs.anl.gov/papers/P7095-0817.pdf>.
- [11] F. M. Hante, G. Leugering, A. Martin, L. Schewe, and M. Schmidt. “Challenges in Optimal Control Problems for Gas and Fluid Flow in Networks of Pipes and Canals: From Modeling to Industrial Applications.” In: *Industrial Mathematics and Complex Systems: Emerging Mathematical Models, Methods and Algorithms*. Ed. by P. Manchanda, R. Lozi, and A. H. Siddiqi. Industrial and Applied Mathematics. Singapore: Springer Singapore, 2017, pp. 77–122. DOI: [10.1007/978-981-10-3758-0_5](https://doi.org/10.1007/978-981-10-3758-0_5).
- [12] J. Jalving, S. Shin, and V. M. Zavala. “A graph-based modeling abstraction for optimization: concepts and implementation in Plasmojl.” In: *Mathematical Programming Computation* (2022). DOI: [10.1007/s12532-022-00223-3](https://doi.org/10.1007/s12532-022-00223-3).
- [13] R. Krug, G. Leugering, A. Martin, M. Schmidt, and D. Weninger. “Time-Domain Decomposition for Optimal Control Problems Governed by Semilinear Hyperbolic Systems.” In: *SIAM Journal on Control and Optimization* 59.6 (2021), pp. 4339–4372. DOI: [10.1137/20M138329X](https://doi.org/10.1137/20M138329X).
- [14] R. Krug, G. Leugering, A. Martin, M. Schmidt, and D. Weninger. “Time-Domain Decomposition for Optimal Control Problems Governed by Semilinear Hyperbolic Systems with Mixed Two-Point Boundary Conditions.” In: *Control and Cybernetics* 50.4 (2021), pp. 427–455. DOI: [10.2478/candc-2021-0026](https://doi.org/10.2478/candc-2021-0026).
- [15] G. Leugering, A. Martin, M. Schmidt, and M. Sirvent. “Nonoverlapping Domain Decomposition for Optimal Control Problems governed by Semilinear Models for Gas Flow in Networks.” In: *Control and Cybernetics* 46.3 (2017), pp. 191–225.
- [16] S. Na, S. Shin, M. Anitescu, and V. M. Zavala. “On the Convergence of Overlapping Schwarz Decomposition for Nonlinear Optimal Control.” In: *IEEE Transactions on Automatic Control* (2022), pp. 1–16. DOI: [10.1109/TAC.2022.3194087](https://doi.org/10.1109/TAC.2022.3194087).
- [17] 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.” In: *Optimization Methods and Software* 30.1 (2015), pp. 15–53. DOI: [10.1080/10556788.2014.888426](https://doi.org/10.1080/10556788.2014.888426).
- [18] D. Rose, M. Schmidt, M. C. Steinbach, and B. M. Willert. “Computational optimization of gas compressor stations: MINLP models versus continuous reformulations.” In: *Mathematical Methods of Operations Research* 83.3 (2016), pp. 409–444. DOI: [10.1007/s00186-016-0533-5](https://doi.org/10.1007/s00186-016-0533-5).
- [19] L. Schewe, M. Schmidt, and D. Weninger. “A decomposition heuristic for mixed-integer supply chain problems.” In: *Operations Research Letters* 48.3 (2020), pp. 225–232. DOI: [10.1016/j.orl.2020.02.006](https://doi.org/10.1016/j.orl.2020.02.006).
- [20] M. Schmidt, D. Aßmann, R. Burlacu, J. Humpola, I. Joormann, N. Kanelakis, T. Koch, D. Oucherif, M. E. Pfetsch, L. Schewe, R. Schwarz, and M. Sirvent. “GasLib—A Library of Gas Network Instances.” In: *Data* 2.4 (2017). DOI: [10.3390/data2040040](https://doi.org/10.3390/data2040040).
- [21] M. Schmidt, M. C. Steinbach, and B. M. Willert. “High detail stationary optimization models for gas networks.” In: *Optimization and Engineering* 16.1 (2015), pp. 131–164. DOI: [10.1007/s11081-014-9246-x](https://doi.org/10.1007/s11081-014-9246-x).
- [22] S. Shin, M. Anitescu, and V. M. Zavala. “Exponential Decay of Sensitivity in Graph-Structured Nonlinear Programs.” In: *SIAM Journal on Optimization* 32.2 (2022), pp. 1156–1183. DOI: [10.1137/21M1391079](https://doi.org/10.1137/21M1391079).
- [23] S. Shin, P. Hart, T. Jahns, and V. M. Zavala. “A Hierarchical Optimization Architecture for Large-Scale Power Networks.” In: *IEEE Transactions on Control of Network Systems* 6.3 (2019), pp. 1004–1014. DOI: [10.1109/TCNS.2019.2906917](https://doi.org/10.1109/TCNS.2019.2906917).
- [24] S. Shin, V. M. Zavala, and M. Anitescu. “Decentralized Schemes With Overlap for Solving Graph-Structured Optimization Problems.” In: *IEEE Transactions on Control of Network Systems* 7.3 (2020), pp. 1225–1236. DOI: [10.1109/TCNS.2020.2967805](https://doi.org/10.1109/TCNS.2020.2967805).
- [25] V. M. Zavala. “Stochastic optimal control model for natural gas networks.” In: *Computers & Chemical Engineering* 64 (2014), pp. 103–113. DOI: [10.1016/j.compchemeng.2014.02.002](https://doi.org/10.1016/j.compchemeng.2014.02.002).

(R. Krug, G. Leugering, A. Martin, D. Weninger) ¹FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DEPARTMENT OF DATA SCIENCE, CAUERSTR. 11, 91058 ERLANGEN, GERMANY

Email address: richard.krug@fau.de

Email address: guenter.leugering@fau.de

Email address: alexander.martin@fau.de

Email address: dieter.weninger@fau.de

(M. Schmidt) ²TRIER UNIVERSITY, DEPARTMENT OF MATHEMATICS, UNIVERSITÄTSRING 15, 54296 TRIER, GERMANY

Email address: martin.schmidt@uni-trier.de