Model and Discretization Error Adaptivity within Stationary Gas Transport Optimization

Volker Mehrmann[1], Martin Schmidt[2,3], Jeroen J. Stolwijk[1]

Abstract. The minimization of operation costs for natural gas transport networks is studied. Based on a recently developed model hierarchy ranging from detailed models of instationary partial differential equations with temperature dependence to highly simplified algebraic equations, modeling and discretization error estimates are presented to control the overall error in an optimization method for stationary and isothermal gas flows. The error control is realized by switching to more detailed models or finer discretizations if necessary to guarantee that a prescribed model and discretization error tolerance is satisfied in the end. We prove convergence of the adaptively controlled optimization method and illustrate the new approach with numerical examples.

Dedicated to Hans Georg Bock on the occasion of his 70th birthday.

1. Introduction

In this paper we discuss the minimization of operation costs for natural gas transport networks based on a model hierarchy, see [11, 21], which ranges from detailed models based on instationary partial differential equations with temperature dependence to highly simplified algebraic equations. The detailed models are necessary to achieve a good understanding of the system state, but in many practical optimization applications only the stationary algebraic equations—or even further simplifications like piecewise linearizations as in [15, 16, 32]—are used in order to reduce the high computational effort of evaluating the state of the system with the more sophisticated models. However, it is then unclear how good the true state is approximated by these simplified models and error bounds are typically not available in this context; see the chapter [22] in [23] for a more detailed discussion of this issue. Recently, in [37], a detailed error and perturbation analysis has been developed for several components in the model hierarchy and it has been shown how the more detailed model components can be used to estimate the error obtained in the simplified models.

Here, we use these error estimates from the model hierarchy together with classical error estimate grid adaptation techniques for the space discretization within an optimization method to control the error adaptively by switching to more detailed models or finer discretizations if necessary. Moreover, our adaptive method also allows to locally switch back to coarser models or to coarser discretizations if they are sufficiently accurate with respect to the local flow situation. Our new approach can, in general, be used for the entire model hierarchy by also using space-time grid adaptation. However, to keep things simple and to illustrate the functionality of the new adaptive approach, we will use three stationary isothermal models from the hierarchy in [11].

1

Using adaptive techniques to achieve a trade-off between computational efficiency and accuracy by using adaptive discretization methods in the context of optimization and optimal control problems is an important research topic, in particular in the context of real-time optimal control of constrained dynamical systems, see, e.g., [3, 9, 8, 29], or in the context of optimal control of problems constrained by partial differential equations; see, e.g., [1, 24, 25, 26]. We extend these ideas and combine adaptive grid refinement and model selection in a model hierarchy in the context of nonlinear optimization problems. We also theoretically analyze the new algorithm. First promising numerical results for such an approach were presented in [34, 35].

The paper is structured as follows. The models used in this paper are described in Sect. 2 together with a simple first-order Euler method for the space discretization. In Sect. 3 we introduce model and discretization error estimators, which are used in Sect. 4 to derive an adaptive model and discretization control algorithm for the nonlinear optimization of gas transport networks that, in the end, delivers solutions that satisfy prescribed error tolerances. Numerical results are presented in Sect. 5 and the paper concludes in Sect. 6.

2. Problem Description, Modeling Hierarchy, and Discretizations

In this section we introduce the problem of operation cost minimization for natural gas transport networks. We present our overall model of a gas transport network involving continuous nonlinear models describing a stationary flow for all the considered network elements. Since the majority of the elements are pipes, our focus lies on the precise and physically accurate modeling of these pipes. The typical models for the pipe flow are nonlinear instationary partial differential equations (PDEs) on a graph and their appropriate space-time discretizations. To address the fact that the behavior of the flow and the accuracy of the model may vary significantly in different regions of the network, we discuss a small part of the complete model hierarchy of instationary models, see [11], where the lower level models in the hierarchy are simplifications of the higher level models. Which model is most appropriate to obtain a computationally tractable, adequately accurate, and finite-dimensional approximation depends on the task that needs to be performed with the model.

Our modeling approach is based on the following physical assumptions. First, we only consider a stationary gas flow, i.e., we neglect all time effects of gas dynamics, so that we have ordinary differential equations (ODEs) in space instead of systems of PDEs on a graph. Second, we assume an isothermal regime, i.e., we neglect all effects arising from changes in the gas temperature.

These assumptions are taken carefully such that we still obtain physically meaningful solutions and such that we are still able to derive and analyze an adaptive model and discretization control algorithm—without unnecessarily overloading the models with all technical details of the application that may distract us from the main mathematical ideas.

2.1. **The Network.** We model the gas transport network by a directed and connected graph $G = (V, A)$. The node set is made up of entry nodes V_+, where gas is supplied, of exit nodes V_-, where gas is discharged, and of inner nodes V_0, i.e., we have $V = V_+ \cup V_- \cup V_0$. The set of arcs in our models comprises pipes A_{pi} and compressor machines A_{cm}, i.e., we have $A = A_{pi} \cup A_{cm}$.

Real-world gas transport networks contain many other element types like (control) valves, short cuts, or resistors. For detailed information on modeling these devices, see [14] in general or [34, 35] for a focus on nonlinear programming (NLP) type models. However, we restrict ourselves to models with pipes and compressors in order to streamline the presentation of our basic ideas and methods, and to show in

a prototypical way that our approach of space discretization and model adaptivity leads to major accuracy and efficiency improvements.

As basic quantities we introduce gas pressure variables p_u at all nodes $u \in V$ and mass flow variables q_a at all arcs $a \in A$ of the network. Both types of variables are bounded due to technical constraints on the pipes, i.e.,

$$p_u \in [\underline{p}_u, \overline{p}_u] \quad \text{for all } u \in V, \tag{1a}$$

$$q_a \in [\underline{q}_a, \overline{q}_a] \quad \text{for all } a \in A. \tag{1b}$$

All other required quantities are introduced where they are used first.

2.2. **Nodes.** In stationary gas network models, the nodes $u \in V$ are modeled by a mass balance equation, i.e., we have the constraint

$$\sum_{a \in \delta^{\mathrm{in}}(u)} q_a - \sum_{a \in \delta^{\mathrm{out}}(u)} q_a = q_u \quad \text{for all } u \in V, \tag{2}$$

where for ingoing arcs we use the notation

$$\delta^{\mathrm{in}}(u) := \{a \in A : \text{there exists } w \in V \text{ and } a = (w, u)\}$$

and for outgoing arcs

$$\delta^{\mathrm{out}}(u) := \{a \in A : \text{there exists } w \in V \text{ and } a = (u, w)\}.$$

Moreover, q_u models the supplied or discharged mass flow at the corresponding node, i.e., we have

$$q_u \begin{cases} \geq 0 & \text{for all } u \in V_-, \\ \leq 0 & \text{for all } u \in V_+, \\ = 0 & \text{for all } u \in V_0. \end{cases}$$

2.3. **Pipes.** Isothermal gas flow through cylindrical pipes is described by the Euler equations for compressible fluids,

$$\frac{\partial \rho}{\partial t} + \frac{1}{A} \frac{\partial q}{\partial x} = 0, \tag{3a}$$

$$\frac{1}{A} \frac{\partial q}{\partial t} + \frac{\partial p}{\partial x} + \frac{1}{A} \frac{\partial(qv)}{\partial x} = -\lambda(q) \frac{|v|v}{2D} \rho - g\rho h', \tag{3b}$$

see, e.g., [13, 27] for a detailed discussion. Here and in what follows, ρ is the gas density, v is its velocity, $\lambda = \lambda(q)$ is the friction term, A denotes the cross-sectional area of the pipe, h' is its slope, and D is the diameter of the pipe. Furthermore, g is the acceleration due to gravity, t is the temporal coordinate, and $x \in [0, L]$ is the spatial coordinate with L being the length of the pipe. Equation (3a) is called the continuity equation and (3b) the momentum equation. Since we only consider the stationary case, all partial derivatives with respect to time vanish and we obtain the simplified stationary model

$$\frac{1}{A} \frac{\partial q}{\partial x} = 0, \tag{4a}$$

$$\frac{\partial p}{\partial x} + \frac{1}{A} \frac{\partial(qv)}{\partial x} = -\lambda(q) \frac{|v|v}{2D} \rho - g\rho h'. \tag{4b}$$

Thus, the continuity equation in its stationary variant simply states that the mass flow along the pipe is constant, i.e., $q(x) \equiv q = \text{const}$ for all $x \in [0, L]$.

To simplify the stationary momentum equation (4b), we consider two more model equations. First, the equation of state

$$p = \rho c^2 \quad \text{with} \quad c = \sqrt{R_{\mathrm{s}} T z},$$

where c is the speed of sound, R_{s} is the specific gas constant, and z is the compressibility factor. The second model is the relation of gas mass flow, density, and velocity given by

$$q = A\rho v.$$

Substituting both these models into (4b), we obtain

$$\frac{\partial p}{\partial x}\left(1 - \frac{q^2}{A^2}\frac{c^2}{p^2}\right) = -\frac{\lambda c^2}{2A^2 D p}\,|q|\,q - \frac{gh'}{c^2}p, \tag{M_1}$$

i.e., the stationary momentum equation written in dependence of the gas pressure $p = p(x)$, $x \in [0, L]$, and the mass flow q.

A simplified version of the latter equation can be obtained by ignoring the ram pressure term

$$\frac{1}{A}\frac{\partial(qv)}{\partial x},$$

in (4b), i.e., the total pressure exerted on the gas by the pipe wall, or, equivalently, the term

$$-\frac{q^2}{A^2}\frac{c^2}{p^2}\frac{\partial p}{\partial x} \tag{5}$$

in (M_1). For a discussion of this simplification step, see [38]. Neglecting the ram pressure term (5) yields

$$\frac{\partial p}{\partial x} = -\frac{\lambda c^2}{2A^2 D p}\,|q|\,q - \frac{gh'}{c^2}p. \tag{M_2}$$

Finally, one may also neglect gravitational forces, i.e., set the term $gh'p/c^2$ to 0 and obtain

$$\frac{\partial p}{\partial x} = -\frac{\lambda c^2}{2A^2 D p}\,|q|\,q. \tag{M_3}$$

Analytical solutions for the models (M_1)–(M_3) are only rarely known; see, e.g., [18, 19, 34]. Thus, in order to obtain finite-dimensional nonlinear optimization models, we discretize these differential equations in space. Applying, e.g., the implicit Euler method we obtain

$$\frac{p_k - p_{k-1}}{h}\left(1 - \frac{q^2}{A^2}\frac{c^2}{p_k^2}\right) = -\frac{\lambda c^2}{2A^2 D p_k}\,|q|\,q - \frac{gh'}{c^2}p_k, \qquad k = 1,\ldots,n, \tag{D_1}$$

$$\frac{p_k - p_{k-1}}{h} = -\frac{\lambda c^2}{2A^2 D p_k}\,|q|\,q - \frac{gh'}{c^2}p_k, \qquad k = 1,\ldots,n, \tag{D_2}$$

$$\frac{p_k - p_{k-1}}{h} = -\frac{\lambda c^2}{2A^2 D p_k}\,|q|\,q, \qquad k = 1,\ldots,n, \tag{D_3}$$

where $p_k = p(x_k)$ and $\Gamma = \{x_0, x_1, \ldots, x_n\}$ is an equidistant spatial discretization of the pipe with constant stepsize $h = x_k - x_{k-1}$ and $x_0 = 0, x_n = L$. Of course, one could also apply a higher-order Runge–Kutta method, which would allow a larger stepsize and would thus reduce the computational cost.

These discretizations extend the model hierarchy (M_1)–(M_3) for the Euler equations by infinitely many models that are parameterized by the discretization stepsize h applied in (D_1)–(D_3). In summary, we obtain the pipe model hierarchy of stationary Euler equations depicted in Fig. 1.

2.4. **Compressors.** Compressor machines $a = (u, w) \in A_{\mathrm{cm}}$ increase the inflow gas pressure to a higher outflow pressure, i.e., they can be described in a simplified way by

$$p_w = p_u + \Delta_a, \quad \Delta_a \in [0, \bar{\Delta}_a] \quad \text{for all } a \in A_{\mathrm{cm}}. \tag{6}$$

Moreover, for simplicity, we assume that we are given cost coefficients $\omega_a \geq 0$ for every compressor $a \in A_{\mathrm{cm}}$ that converts pressure increase to compression cost. Of
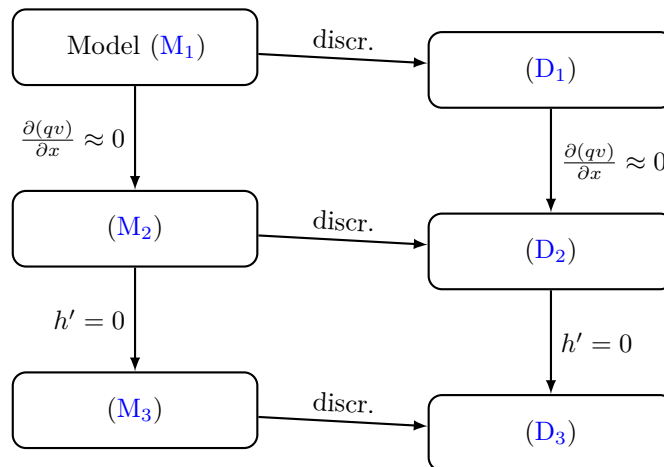
FIGURE 1. Pipe model hierarchy based on the Euler equations. The space continuous models are positioned in the left column and their space discretized counterparts are positioned in the right column.

course, this is an extremely coarse approximation of a compressor machine. An alternative would be to use a simple input-output surrogate model obtained from a realization or system identification of an input-output transfer function; see, e.g., [5]. However, our focus is on an accurate modeling of the gas flow in pipes and on deriving an adaptive model and discretization control algorithm. Model (6) allows for setting up a reasonable objective function for our NLPs and is thus appropriate in this work. For more details, see [31, 34, 35] or [14].

2.5. **The Optimization Problem.** We will use the adaptive model and discretization control algorithm in the context of the following nonlinear ODE-constrained optimization problem

$$\min \quad \sum_{a \in A_{\mathrm{cm}}} \omega_a \Delta_a \tag{7a}$$

$$\text{s.t.} \quad \text{variable bounds (1),} \tag{7b}$$

$$\text{mass balance (2),} \tag{7c}$$

$$\text{compressor model (6) for all } a \in A_{\mathrm{cm}}, \tag{7d}$$

$$\text{pipe model } (\mathrm{M}_1) \text{ for all } a \in A_{\mathrm{pi}}, \tag{7e}$$

where our objective function models the cost for the compressor activity that is constrained by an infinite-dimensional description of the gas flow in pipes. Problem (7) is a classical nonlinear optimal control problem. A typical approach to solve such problems in practice is the first-discretize-then-optimize paradigm; see, e.g., [2]. In this setting, one replaces the ODE constraints by finite sets of nonlinear constraints that arise, e.g., from implicit Euler discretizations like (D_1) for (M_1). Moreover, practical experience suggests that for the evaluation of the constraints, it is often not required to apply the most accurate model like (D_1) with a small stepsize for every pipe in the network. Instead, in many situations it is sufficient to use simplified models like (D_2) and (D_3) with a coarse grid, which then typically yields fast execution times for the evaluation of the constraint functions.

To this end, we define discretized problem variants of Problem (7) by specifying the model level $\ell_a \in \{1, 2, 3\}$ for every arc $a \in A_{\mathrm{pi}}$ (i.e., the discretized model (D_1),
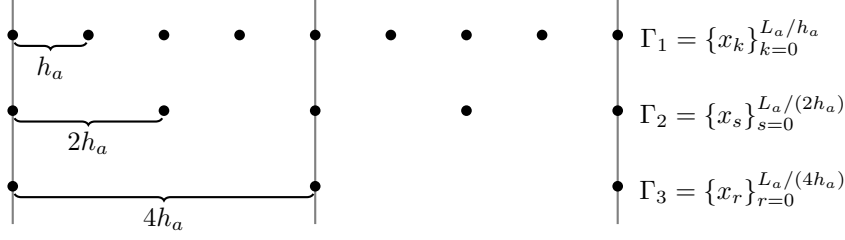
FIGURE 2. Overview of the three considered discretization grids Γ_1, Γ_2, and Γ_3 with gridpoints x_k, x_s, and x_r and stepsizes h_a, $2h_a$, and $4h_a$, respectively. The vertical lines represent the evaluation grid Γ_3 for the error estimators in (9) and (10).

(D_2), or (D_3), respectively) together with a stepsize h_a. This yields the family of finite-dimensional NLPs

$$\min \quad \sum_{a \in A_{\mathrm{cm}}} \omega_a \Delta_a \tag{8a}$$

$$\text{s.t.} \quad \text{variable bounds (1),} \tag{8b}$$

$$\text{mass balance (2),} \tag{8c}$$

$$\text{compressor model (6) for all } a \in A_{\mathrm{cm}}, \tag{8d}$$

$$\text{pipe model } (D_{\ell_a}) \text{ with stepsize } h_a \text{ for all } a \in A_{\mathrm{pi}}. \tag{8e}$$

Note that the constraints (7b)–(7d) in the infinite-dimensional problem are exactly the same as constraints (8b)–(8d) in the family of discretized problems.

3. ERROR ESTIMATORS

In this section we introduce a first-order estimate for the error between the most detailed infinite-dimensional and an arbitrary space-discretized model. This error estimator is obtained as the sum of a discretization and a model error estimator. Since we consider the stationary case, mass flows in pipes are constant in the spatial dimension. This is why we base our error estimators on the differences of the pressures $p(x)$ for different models and discretizations.

Suppose that for a given pipe $a \in A_{\mathrm{pi}}$, the model level $\ell_a \in \{1, 2, 3\}$ with discretization stepsize h_a is currently used for the computations. The overall solution of the optimization problem for the entire network, also including pressure increases in compressors etc., is denoted by y and contains the discretized pressure distributions of the separate pipes $a \in A_{\mathrm{pi}}$, which we denote by $p^{\ell_a}(x_k; h_a)$ with discretization grid $\Gamma_1 = \{x_k\}_{k=0}^{L_a/h_a}$ obtained by using the stepsize h_a. We now compute an estimate for the error between the solution of the currently used model (D_{ℓ_a}) and the solution of the reference model (M_1). Let the solution of model (M_1) for pipe $a \in A_{\mathrm{pi}}$ be denoted by $\hat{p}(x)$ with $x \in [0, L_a]$.

Furthermore, let the solutions of Model (D_1) with discretization grids $\Gamma_2 = \{x_s\}_{s=0}^{L_a/(2h_a)}$ and $\Gamma_3 = \{x_r\}_{r=0}^{L_a/(4h_a)}$ using stepsizes $2h_a$ and $4h_a$, be denoted by $p^1(x_s; 2h_a)$ and $p^1(x_r; 4h_a)$, respectively. Due to the larger stepsize, the computation of these two solutions is in general less expensive than computing a solution of Model (D_{ℓ_a}) on the grid Γ_1. Since the discretization grid Γ_3 is the coarsest grid and all computed pressure profiles can be evaluated on this grid, Γ_3 is called the evaluation grid. This grid is used in the definitions of the following error estimators. The considered discretization grids and the evaluation grid are depicted in Fig. 2.
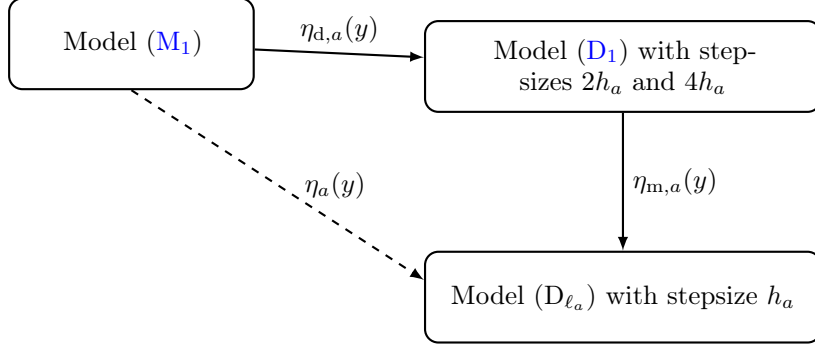
FIGURE 3. Overview of the models and stepsizes used for the computation of the overall error estimator $\eta_a(y)$ between models (M_1) and (D_{ℓ_a}) in (11). Here, for a pipe a, $\eta_{d,a}(y)$ is the discretization error estimator and $\eta_{m,a}(y)$ is the model error estimator.

For a pipe $a \in A_{pi}$, let the discretization error estimator be defined by

$$\eta_{d,a}(y) := \left\| p^1(x_r; 2h_a) - p^1(x_r; 4h_a) \right\|_\infty \tag{9}$$

and let the model error estimator be defined by

$$\eta_{m,a}(y) := \left\| p^1(x_r; 2h_a) - p^{\ell_a}(x_r; h_a) \right\|_\infty . \tag{10}$$

Here,

$$p^{\ell_a}(x_r; h_a) = [p^{\ell_a}(x_0; h_a), \dots, p^{\ell_a}(x_n; h_a)]^\top, \quad n = L_a/(4h_a),$$

denotes the solution of Model (D_{ℓ_a}) computed with stepsize h_a that is evaluated at the gridpoints x_r, i.e., on the grid Γ_3. If $\ell_a = 1$, i.e., if the considered solution already corresponds to the most accurate model, then we set the model error to zero, i.e., $\eta_{m,a}(y) = 0$. Furthermore, let the overall error estimator $\eta_a(y)$ for a pipe $a \in A_{pi}$ be defined to be a first-order upper bound for the maximum error between the solutions of models (M_1) and (D_{ℓ_a}) at gridpoints x_r with stepsize $4h_a$. Thus, we have

$$\begin{aligned}
&\left\| \hat{p}(x_r) - p^{\ell_a}(x_r; h_a) \right\|_\infty \\
\leq \ &\left\| \hat{p}(x_r) - p^1(x_r; 2h_a) \right\|_\infty + \left\| p^1(x_r; 2h_a) - p^{\ell_a}(x_r; h_a) \right\|_\infty \\
\doteq \ &\eta_{d,a}(y) + \eta_{m,a}(y) =: \eta_a(y),
\end{aligned} \tag{11}$$

where $\doteq$ denotes a first-order approximation in h_a, see [36, page 420], and we use that the implicit Euler method has convergence order 1. The error estimator $\eta_a(y)$ is the absolute counterpart of the componentwise relative error estimator given in [37]. An overview of the considered models in this section together with the considered stepsizes is depicted in Fig. 3.

We close this section with a remark on the computation of the discretization error estimator in (9). A straightforward way is to solve Model (D_1) once with stepsize $2h_a$ and once again with stepsize $4h_a$ for every $a \in A_{pi}$. Another possibility would be to use an embedded Runge–Kutta method, see, e.g., [20], which in general saves computational cost due to the reduced number of function evaluations.

4. THE GRID AND MODEL ADAPTATION ALGORITHM

In this section we present and analyze an algorithm that adaptively switches between the model levels in the hierarchy of Fig. 1 and adapts discretization stepsizes in order to find a convenient trade-off between physical accuracy and computational costs. To this end, the algorithm iteratively solves NLPs and initial value problems

(IVPs). Solutions of the latter are used to evaluate the error estimators discussed in the last section and to decide on the model levels and the discretization stepsizes for the next NLP.

Consider a single NLP of the sequence of NLPs that are solved during the algorithm and assume that pipe $a \in A_{\mathrm{pi}}$ is modeled using model (D_{ℓ_a}) and stepsize h_a. Let the solution of this NLP be denoted by y. According to the last section, the overall model and discretization error estimator for this pipe is given by $\eta_a(y)$ as defined in (11). Thus, it is given by the error estimator between the solutions of the most accurate model (M_1) and the current model (D_{ℓ_a}).

The overall goal of our method is to compute a solution of a member of the family of discretized problems (8) for which it is guaranteed that this solution has an estimated average error per pipe with respect to the reference model (M_1), that is less than an a-priorily given tolerance $\varepsilon > 0$. This leads us to the following definition:

Definition 1 (ε-feasibility)**.** Let $\varepsilon > 0$ be given. We say that a solution y of problem (8) with discretized models (D_{ℓ_a}), $\ell_a \in \{1, 2, 3\}$, and stepsizes h_a for the pipes $a \in A_{\mathrm{pi}}$ is ε-feasible with respect to the reference problem (7) if

$$\frac{1}{|A_{\mathrm{pi}}|} \sum_{a \in A_{\mathrm{pi}}} \eta_a(y) \leq \varepsilon.$$

The remainder of this section is organized as follows. Sect. 4.1 introduces rules about how the model levels and discretization stepsizes are modified. The strategies for marking pipes for model or grid adaptation are explained in Sect. 4.2. The adaptive model and discretization control algorithm is introduced in Sect. 4.3, together with a theorem for the finite termination of the algorithm. Finally, some remarks regarding the adaptive control algorithm are given in Sect. 4.4.

4.1. **Model and Discretization Adaptation Rules.** Before we present and discuss the overall adaptive model control algorithm we have to

(1) describe the mechanisms of switching up or down pipe model levels as well as that of refining and coarsening the discretization grids, and

(2) discuss our marking strategy that determines the arcs on which the model or grid should be adapted.

We start with the first issue and follow the standard PDE grid adaptation technique; see, e.g., [6, 7, 12] or [4]. The general strategy is as follows. We switch up one level in the model hierarchy if this yields an error reduction that is larger than ε; otherwise, we switch up to the most accurate discretized model (D_1). Hence, for pipe $a \in A_{\mathrm{pi}}$ we have the rule

$$\ell_a^{\mathrm{new}} = \begin{cases} \ell_a - 1, & \text{if } \eta_{\mathrm{m},a}(y; \ell_a) - \eta_{\mathrm{m},a}(y; \ell_a - 1) > \varepsilon, \\ 1, & \text{otherwise,} \end{cases} \tag{12}$$

for switching up levels in the model hierarchy. We apply this rule because it is possible that the effects of neglecting the ram pressure term (which is the difference between model levels $\ell = 1$ and $\ell = 2$) and neglecting gravitational forces for non-horizontal pipes (which is the difference between model levels $\ell = 2$ and $\ell = 3$) balance each other out in the computation of the pressure profile of model (D_3). In this case, switching from model (D_3) to (D_2) would increase the model error, which is why we switch from (D_3) to (D_1) directly.

A discretization grid refinement or coarsening with a factor $\gamma > 1$ is defined by taking the new stepsize as

$$h_a^{\text{new}} := \begin{cases} h_a/\gamma, & \text{for a grid refinement,} \\ \gamma h_a, & \text{for a grid coarsening.} \end{cases} \tag{13}$$

For a discretization scheme of order β it is well-known that a first-order approximation for the discretization error in $x \in [0, L_a]$ is given by $e_{\text{d},a}(x) \doteq c(x) h_a^\beta$, where $c(x)$ is independent of h_a; see, e.g., [36]. From this, it follows that the new discretization error after a grid refinement or coarsening can be written as

$$e_{\text{d},a}^{\text{new}}(x) \doteq (h_a^{\text{new}}/h_a)^\beta e_{\text{d},a}(x).$$

Since the implicit Euler method has convergence order $\beta = 1$, with h_a^{new} in (13) and $\gamma = 2$, for the new discretization error estimator after a grid refinement or coarsening, it holds that

$$\eta_{\text{d},a}^{\text{new}}(y) \doteq \begin{cases} \eta_{\text{d},a}(y)/2, & \text{for a grid refinement,} \\ 2\eta_{\text{d},a}(y), & \text{for a grid coarsening.} \end{cases} \tag{14}$$

4.2. **Marking Strategies.** We now describe our marking strategies, i.e., how we choose which pipes should be switched up or down in their model level and which pipes should get a refined or coarsened grid. Given marking strategy parameters $\Theta_\text{d}, \Theta_\text{m} \in [0, 1]$, we compute subsets $\mathcal{R}, \mathcal{U} \subseteq A_\text{pi}$ such that they are the minimal subsets of arcs that satisfy

$$\Theta_\text{d} \sum_{a \in A_\text{pi}} \eta_{\text{d},a}(y) \leq \sum_{a \in \mathcal{R}} \eta_{\text{d},a}(y) \tag{15}$$

and

$$\Theta_\text{m} \sum_{a \in A_\text{pi}^{>\varepsilon}} (\eta_{\text{m},a}(y; \ell_a) - \eta_{\text{m},a}(y; \ell_a^{\text{new}})) \leq \sum_{a \in \mathcal{U}} (\eta_{\text{m},a}(y; \ell_a) - \eta_{\text{m},a}(y; \ell_a^{\text{new}})) \tag{16}$$

with

$$A_\text{pi}^{>\varepsilon} := \{a \in A_\text{pi} : \eta_{\text{m},a}(y; \ell_a) - \eta_{\text{m},a}(y; \ell_a^{\text{new}}) > \varepsilon\},$$

where ℓ_a^{new} is given in (12). Analogously, given marking strategy parameters $\Phi_\text{d}, \Phi_\text{m} \in [0, 1]$ and $\tau \geq 1$, we compute $\mathcal{C}, \mathcal{D} \subseteq A_\text{pi}$ such that they are the maximal subsets of arcs that satisfy

$$\Phi_\text{d} \sum_{a \in A_\text{pi}} \eta_{\text{d},a}(y) \geq \sum_{a \in \mathcal{C}} \eta_{\text{d},a}(y) \tag{17}$$

and

$$\Phi_\text{m} \sum_{a \in A_\text{pi}^{<\varepsilon}(\tau)} (\eta_{\text{m},a}(y; \ell_a^{\text{new}}) - \eta_{\text{m},a}(y; \ell_a)) \geq \sum_{a \in \mathcal{D}} (\eta_{\text{m},a}(y; \ell_a^{\text{new}}) - \eta_{\text{m},a}(y; \ell_a)) \tag{18}$$

with

$$A_\text{pi}^{<\varepsilon}(\tau) := \{a \in A_\text{pi} : \eta_{\text{m},a}(\ell_a^{\text{new}}) - \eta_{\text{m},a}(\ell_a) \leq \tau\varepsilon\}.$$

In (18), ℓ_a^{new} is always set to $\min\{\ell_a + 1, 3\}$. For every arc $a \in \mathcal{R}$ ($a \in \mathcal{C}$) we refine (coarsen) the discretization grid by halving (doubling) the stepsize, i.e., we set $\gamma = 2$ in (13). We note that these marking strategies are very similar to the greedy strategies on a network described in [10], where those pipes are marked for a spatial, temporal, or model refinement which yield the largest error reduction.

Algorithm 1: Adaptive Model and Discretization Control

Input: A full specification of the gas network $G = (V, A)$, a tolerance $\varepsilon > 0$,
   initial marking strategy parameters $\Theta_d^0, \Theta_m^0, \Phi_d^0, \Phi_m^0 \in [0, 1]$, $\tau^0 \geq 1$,
   and an initial safeguard parameter $\mu^0 \in \mathbb{N}$.

Output: An ε-feasible solution of the reference problem (7).

1 Choose an initial model level ℓ_a^0 and a stepsize h_a^0 for every $a \in A_{\mathrm{pi}}$.

2 Solve Problem (8) and let y^0 denote the optimal solution.

3 Compute $\eta_a(y^0)$ for every $a \in A_{\mathrm{pi}}$.

4 **if** y^0 *is ε-feasible* **then**

5 $\quad\lfloor$ **return** y^0.

6 Set $k = 1$ and $\Theta_d^k = \Theta_d^0, \Theta_m^k = \Theta_m^0, \Phi_d^k = \Phi_d^0, \Phi_m^k = \Phi_m^0, \mu^k = \mu^0, \tau^k = \tau^0$.

7 **for** $k = 1, 2, \ldots$ **do**

8 $\quad$ **for** $j = 1, \ldots, \mu^k$ **do**

9 $\quad\quad$ Compute the sets $\mathcal{U}_{k,j}, \mathcal{R}_{k,j} \subseteq A_{\mathrm{pi}}$ according to (15) and (16).

10 $\quad\quad$ Switch up the model level for every pipe $a \in \mathcal{U}_{k,j}$.

11 $\quad\quad$ Refine the discretization grid for every pipe $a \in \mathcal{R}_{k,j}$.

12 $\quad\quad$ Solve Problem (8) and let $y^{k,j}$ denote the solution.

13 $\quad\quad$ Compute $\eta_a(y^{k,j})$ for every $a \in A_{\mathrm{pi}}$.

14 $\quad\quad$ **if** $y^{k,j}$ *is ε-feasible* **then**

15 $\quad\quad\quad\lfloor$ **return** $y^{k,j}$.

16 $\quad$ Compute the sets $\mathcal{D}_k, \mathcal{C}_k \subseteq A_{\mathrm{pi}}$ according to (17) and (18).

17 $\quad$ Switch down the model level for every pipe $a \in \mathcal{D}_k$.

18 $\quad$ Coarsen the discretization grid for every pipe $a \in \mathcal{C}_k$.

19 $\quad$ Increase $k \leftarrow k + 1$ and update parameters $\Theta_d^k, \Theta_m^k, \Phi_d^k, \Phi_m^k, \mu^k, \tau^k$.

4.3. **The Algorithm.** With these preliminaries we can now state the overall adaptive model and discretization control algorithm for finding an ε-feasible solution of the reference problem (7). The formal listing is given in Alg. 1.

The algorithm makes use of the safeguard parameter $\mu \in \mathbb{N}$. This parameter ensures that the algorithm performs grid coarsenings and switches down the model level only after applying μ rounds of grid refinements and switching up model levels. It prevents an alternating switching up and down model levels or an alternating refining and coarsening of the discretization grid. We note that this technique is similar to the use of hysteresis parameters; see, e.g., [28]. By employing this safeguard, we can prove that Alg. 1 terminates after a finite number of iterations with an ε-feasible point of the reference model (M_1).

To improve readability, we split the proof of our main theorem into two parts. The first lemma states finite termination at an ε-feasible point if only discretization grid refinements and coarsenings are applied, whereas the second lemma considers the case of switching levels in the model hierarchy only, i.e., with a fixed stepsize for every pipe.

Lemma 1. *Suppose that the model level $\ell_a \in \{1, 2, 3\}$ is fixed for every pipe $a \in A_{\mathrm{pi}}$. Let the resulting set of model levels be denoted by $\mathcal{M}$. Suppose further that $\eta_a(y) = \eta_{d,a}(y)$ holds in (11) and that every NLP is solved to local optimality. Consider Alg. 1 without applying the model switching steps in Lines 10 and 17. Then, the algorithm terminates after a finite number of refinements in Line 11 and coarsenings in Line 18 with an ε-feasible solution with respect to model level set $\mathcal{M}$ if there exists*

a constant $C > 0$ such that

$$\frac{1}{2}\Theta_d^k \mu^k > \Phi_d^k + C \tag{19}$$

holds for all k.

Proof. We consider the total discretization error

$$\eta_{\mathrm{d}}(y) = \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}(y)$$

and show that for every iteration k the difference between the decrease obtained in the inner for-loop and the increase obtained due to the coarsenings applied in Line 18 is positive and uniformly bounded away from zero. In what follows, we only consider a single iteration and drop its index k for better readability.

First, we consider one refinement step in Line 11. Let $\eta_{\mathrm{d},a}^{j-1}$ denote the discretization error before the jth inner iteration and let $\eta_{\mathrm{d},a}^{j}$ denote the discretization error after the jth inner iteration. With this, we have

$$\sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{j}$$

$$= \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} + \sum_{a \in A_{\mathrm{pi}} \setminus \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j} - \sum_{a \in A_{\mathrm{pi}} \setminus \mathcal{R}_j} \eta_{\mathrm{d},a}^{j}$$

$$= \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j}$$

$$= \sum_{a \in \mathcal{R}_j} \frac{1}{2}\eta_{\mathrm{d},a}^{j-1}$$

for every $j = 1, \ldots, \mu$. For the last equality we have used that the implicit Euler method has convergence order 1, which (for small stepsizes h_a) implies $\eta_{\mathrm{d},a}^{j} = \frac{1}{2}\eta_{\mathrm{d},a}^{j-1}$ when we take the new stepsize as half the current stepsize; see (14). Summing up over all μ inner iterations we obtain a telescopic sum and finally get an error decrease of

$$\sum_{j=1}^{\mu} \left(\sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{j} \right) = \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{0} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu} = \frac{1}{2}\sum_{j=1}^{\mu} \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1}.$$

We now consider the coarsening step. For this, let $\eta_{\mathrm{d},a}^{\mu}$ denote the discretization error before and $\eta_{\mathrm{d},a}^{\mu+1}$ the discretization error after the coarsening step in Line 18. Using similar ideas like above we obtain

$$\sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu+1} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu}$$

$$= \sum_{a \in A_{\mathrm{pi}} \setminus \mathcal{C}} \eta_{\mathrm{d},a}^{\mu+1} + \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu+1} - \sum_{a \in A_{\mathrm{pi}} \setminus \mathcal{C}} \eta_{\mathrm{d},a}^{\mu} - \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu}$$

$$= \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu+1} - \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu}$$

$$= 2 \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu} - \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu}$$

$$= \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu}.$$

Thus, we are finished if we prove that

$$\frac{1}{2}\sum_{j=1}^{\mu}\sum_{a\in\mathcal{R}_j}\eta_{\mathrm{d},a}^{j-1}-\sum_{a\in\mathcal{C}}\eta_{\mathrm{d},a}^{\mu}$$

is positive and uniformly bounded away from zero. Using

$$\eta_{\mathrm{d},a}^{j-1}\geq\eta_{\mathrm{d},a}^{\mu},\quad\text{for all }j=1,\ldots,\mu,$$

(15), (17), and (19), we obtain

$$\frac{1}{2}\sum_{j=1}^{\mu}\sum_{a\in\mathcal{R}_j}\eta_{\mathrm{d},a}^{j-1}\geq\frac{1}{2}\Theta_{\mathrm{d}}\sum_{j=1}^{\mu}\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{d},a}^{j-1}\geq\frac{1}{2}\Theta_{\mathrm{d}}\sum_{j=1}^{\mu}\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{d},a}^{\mu}$$

$$=\frac{1}{2}\Theta_{\mathrm{d}}\mu\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{d},a}^{\mu}>(\Phi_{\mathrm{d}}+C)\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{d},a}^{\mu}>\sum_{a\in\mathcal{C}}\eta_{\mathrm{d},a}^{\mu}+C\,|A_{\mathrm{pi}}|\,\varepsilon,$$

which completes the proof. $\square$

Next, we prove an analogous lemma for the case that we fix the stepsize of every arc $a\in A_{\mathrm{pi}}$ and only allow for model switching.

Lemma 2. *Suppose that the discretization stepsize h_a is fixed for every pipe $a\in A_{\mathrm{pi}}$. Suppose further that $\eta_a(y)=\eta_{m,a}(y)$ holds in (11) and that every NLP is solved to local optimality. Consider Alg. 1 without applying the discretization refinements in Line 11 and the coarsenings in Line 18. Then, Algorithm 1 terminates after a finite number of model switches in Lines 10 and 17 with an ε-feasible solution with respect to the stepsizes h_a, $a\in A_{\mathrm{pi}}$, if there exists a constant $C>0$ such that*

$$\Theta_m^k\mu^k>\tau^k\Phi_m^k\,|A_{\mathrm{pi}}|+C \tag{20}$$

holds for all k.

Proof. We consider the total model error

$$\eta_{\mathrm{m}}(y)=\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}(y)$$

and show that the difference between the decrease obtained in the inner loop and the increase obtained due to switching model levels down in Line 17 is positive and uniformly bounded away from zero for every iteration k. We again consider only a single iteration and drop the corresponding index.

First, we consider a single step of switching up the model level in Line 10. Let $\eta_{\mathrm{m},a}^{j-1}$ denote the model error before the jth inner iteration and $\eta_{\mathrm{m},a}^j$ the model error after the jth inner iteration. We then have

$$\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}^{j-1}-\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}^{j}$$

$$=\sum_{a\in\mathcal{U}_j}\eta_{\mathrm{m},a}^{j-1}+\sum_{a\in A_{\mathrm{pi}}\backslash\mathcal{U}_j}\eta_{\mathrm{m},a}^{j-1}-\sum_{a\in\mathcal{U}_j}\eta_{\mathrm{m},a}^{j}-\sum_{a\in A_{\mathrm{pi}}\backslash\mathcal{U}_j}\eta_{\mathrm{m},a}^{j}$$

$$=\sum_{a\in\mathcal{U}_j}\eta_{\mathrm{m},a}^{j-1}-\sum_{a\in\mathcal{U}_j}\eta_{\mathrm{m},a}^{j}$$

for every $j=1,\ldots,\mu$. Summing up over all j yields the overall model error decrease after μ for-loop iterations of

$$\sum_{j=1}^{\mu}\left(\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}^{j-1}-\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}^{j}\right)=\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}^{0}-\sum_{a\in A_{\mathrm{pi}}}\eta_{\mathrm{m},a}^{\mu}=\sum_{j=1}^{\mu}\sum_{a\in\mathcal{U}_j}(\eta_{\mathrm{m},a}^{j-1}-\eta_{\mathrm{m},a}^{j}).$$

We now consider the step of switching down the model level in Line 17. Let $\eta_{\mathrm{m},a}^{\mu}$ denote the model error before and $\eta_{\mathrm{m},a}^{\mu+1}$ the model error after this step. It holds that

$$\sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{m},a}^{\mu+1} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{m},a}^{\mu}$$

$$= \sum_{a \in \mathcal{D}} \eta_{\mathrm{m},a}^{\mu+1} + \sum_{a \in A_{\mathrm{pi}} \setminus \mathcal{D}} \eta_{\mathrm{m},a}^{\mu+1} - \sum_{a \in \mathcal{D}} \eta_{\mathrm{m},a}^{\mu} - \sum_{a \in A_{\mathrm{pi}} \setminus \mathcal{D}} \eta_{\mathrm{m},a}^{\mu}$$

$$= \sum_{a \in \mathcal{D}} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu}).$$

Thus, the proof is finished if we show that

$$\sum_{j=1}^{\mu} \sum_{a \in \mathcal{U}_j} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) - \sum_{a \in \mathcal{D}} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu})$$

is positive and uniformly bounded away from zero. With similar ideas as in the proof of Lemma 1 and using (16), (18), and (20), we obtain

$$\sum_{j=1}^{\mu} \sum_{a \in \mathcal{U}_j} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) \geq \Theta_{\mathrm{m}} \sum_{j=1}^{\mu} \sum_{a \in A_{\mathrm{pi}}^{>\varepsilon}} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) > \Theta_{\mathrm{m}} \sum_{j=1}^{\mu} \sum_{a \in A_{\mathrm{pi}}^{>\varepsilon}} \varepsilon$$

$$= \Theta_{\mathrm{m}} \mu \left| A_{\mathrm{pi}}^{>\varepsilon} \right| \varepsilon \geq \Theta_{\mathrm{m}} \mu \varepsilon > \tau \Phi_{\mathrm{m}} \left| A_{\mathrm{pi}} \right| \varepsilon + C\varepsilon \geq \Phi_{\mathrm{m}} \sum_{a \in A_{\mathrm{pi}}^{<\varepsilon}(\tau)} \varepsilon \tau + C\varepsilon$$

$$\geq \Phi_{\mathrm{m}} \sum_{a \in A_{\mathrm{pi}}^{<\varepsilon}(\tau)} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu}) + C\varepsilon \geq \sum_{a \in \mathcal{D}} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu}) + C\varepsilon,$$

where we used that $|A_{\mathrm{pi}}^{>\varepsilon}| \geq 1$. This completes the proof. $\square$

Let $\eta_{\mathrm{m},a}^{\mathrm{new}}(y)$ denote the new model error estimator after a grid refinement or coarsening. In order to prove our main theorem we need to assume that, for every pipe $a \in A_{\mathrm{pi}}$, the change in the model error estimator after a grid refinement or coarsening can be neglected as compared to $\eta_{\mathrm{m},a}(y)$, i.e., $|\eta_{\mathrm{m},a}(y) - \eta_{\mathrm{m},a}^{\mathrm{new}}(y)| \ll \eta_{\mathrm{m},a}(y)$, such that we may write $\eta_{\mathrm{m},a}^{\mathrm{new}}(y) = \eta_{\mathrm{m},a}(y)$. A sufficient condition for this assumption to hold is given by $\eta_{\mathrm{d},a}(y) \ll \eta_{\mathrm{m},a}(y)$ for every $a \in A_{\mathrm{pi}}$. This condition also implies that $\eta_{\mathrm{m},a}(y)$ is a first-order approximation of the exact model error $e_{\mathrm{m},a}(y)$ and is thus reliable for small stepsizes h_a.

Lemma 3. *Let the discretization and model error estimator $\eta_{d,a}(y)$ and $\eta_{m,a}(y)$ as defined in (9) and (10) be given for every $a \in A_{\mathrm{pi}}$. Let further $e_{m,a}(y)$ be the exact error between models (M_1) and (M_{ℓ_a}) and let $\eta_{m,a}^{new}(y)$ be the new model error estimator after a grid refinement or coarsening. Then, the implications*

(1) $\eta_{d,a}(y) \ll \eta_{m,a}(y) \implies \eta_{m,a}(y) \doteq e_{m,a}(y)$,
(2) $\eta_{d,a}(y) \ll \eta_{m,a}(y) \implies \eta_{m,a}^{new}(y) = \eta_{m,a}(y)$

hold for every $a \in A_{\mathrm{pi}}$.

Proof. Let pipe $a \in A_{\mathrm{pi}}$ be arbitrary. To improve readability, in the following we drop the dependencies of the exact errors and the error estimators on a and y. Without loss of generality, we consider only one arbitrary spatial gridpoint x_k.

Let us first introduce some notation. The exact model error is given by $e_{\mathrm{m}}(x_k) = \hat{p}(x_k) - p^{\mathrm{M}_{\ell_a}}(x_k)$ for the current model level ℓ_a, the exact discretization error for model (D_1) is given by $e_{\mathrm{d}}^1(x_k) = \hat{p}(x_k) - p^1(x_k; 2h_a)$ and the exact discretization error for model (D_{ℓ_a}) is denoted by $e_{\mathrm{d}}^{\ell_a}(x_k) = p^{\mathrm{M}_{\ell_a}}(x_k) - p^{\ell_a}(x_k; 2h_a)$. Furthermore, the model error estimator is given by $\eta_{\mathrm{m}}(x_k) = p^1(x_k; 2h_a) - p^{\ell_a}(x_k; 2h_a)$, see (10), and we define the discretization error estimators $\eta_{\mathrm{d}}^1(x_k) := p^1(x_k; 2h_a) - p^1(x_k; 4h_a)$

and $\eta_{\mathrm{d}}^{\ell_a}(x_k) := p^{\ell_a}(x_k; 2h_a) - p^{\ell_a}(x_k; 4h_a)$ as in (9). Then, we have $\eta_{\mathrm{d}}^1(x_k) \doteq e_{\mathrm{d}}^1(x_k)$ and $\eta_{\mathrm{d}}^{\ell_a}(x_k) \doteq e_{\mathrm{d}}^{\ell_a}(x_k)$; see [36, page 420]. Further, it holds that

$$\left|\eta_{\mathrm{d}}^1(x_k)\right| \ll |\eta_{\mathrm{m}}(x_k)| \iff |\eta_{\mathrm{d}}^{\ell_a}(x_k)| \ll |\eta_{\mathrm{m}}(x_k)|, \tag{21}$$

because $\eta_{\mathrm{d}}^1(x_k)$ and $\eta_{\mathrm{d}}^{\ell_a}(x_k)$ use the same stepsizes $2h_a$ and $4h_a$ to compute the discrete pressure distributions.

We now prove implication (1). Using the previously defined notation it holds that

$$\begin{aligned}
e_{\mathrm{m}}(x_k) &= \hat{p}(x_k) - p^{\mathrm{M}_{\ell_a}}(x_k) \\
&= e_{\mathrm{d}}^1(x_k) + p^1(x_k; 2h_a) - e_{\mathrm{d}}^{\ell_a}(x_k) - p^{\ell_a}(x_k; 2h_a) \\
&\doteq \eta_{\mathrm{d}}^1(x_k) + p^1(x_k; 2h_a) - \eta_{\mathrm{d}}^{\ell_a}(x_k) - p^{\ell_a}(x_k; 2h_a) \\
&= \eta_{\mathrm{d}}^1(x_k) - \eta_{\mathrm{d}}^{\ell_a}(x_k) + \eta_{\mathrm{m}}(x_k).
\end{aligned}$$

Thus, if $|\eta_{\mathrm{d}}^1(x_k)|$ and $|\eta_{\mathrm{d}}^{\ell_a}(x_k)|$ may be neglected as compared to $|\eta_{\mathrm{m}}(x_k)|$, then we have $e_{\mathrm{m}}(x_k) \doteq \eta_{\mathrm{m}}(x_k)$, i.e.,

$$\left|\eta_{\mathrm{d}}^1(x_k)\right| \ll |\eta_{\mathrm{m}}(x_k)| \wedge |\eta_{\mathrm{d}}^{\ell_a}(x_k)| \ll |\eta_{\mathrm{m}}(x_k)| \implies e_{\mathrm{m}}(x_k) \doteq \eta_{\mathrm{m}}(x_k).$$

Considering also the equivalence relation (21) it follows that

$$\left|\eta_{\mathrm{d}}^1(x_k)\right| \ll |\eta_{\mathrm{m}}(x_k)| \implies e_{\mathrm{m}}(x_k) \doteq \eta_{\mathrm{m}}(x_k),$$

from which implication (1) follows directly.

Finally, we prove implication (2). We show that this implication holds for the case that $\eta_{\mathrm{m}}^{\mathrm{new}}(x_k)$ is the new model error estimator after a grid coarsening. The case for a grid refinement can be shown analogously. It holds that

$$\begin{aligned}
\eta_{\mathrm{m}}^{\mathrm{new}}(x_k) &= p^1(x_k; 4h_a) - p^{\ell_a}(x_k; 4h_a) \\
&= -\eta_{\mathrm{d}}^1(x_k) + p^1(x_k; 2h_a) + \eta_{\mathrm{d}}^{\ell_a}(x_k) - p^{\ell_a}(x_k; 2h_a) \\
&= -\eta_{\mathrm{d}}^1(x_k) + \eta_{\mathrm{d}}^{\ell_a}(x_k) + \eta_{\mathrm{m}}(x_k).
\end{aligned}$$

This yields

$$\left|\eta_{\mathrm{d}}^1(x_k)\right| \ll |\eta_{\mathrm{m}}(x_k)| \wedge |\eta_{\mathrm{d}}^{\ell_a}(x_k)| \ll |\eta_{\mathrm{m}}(x_k)| \implies \eta_{\mathrm{m}}^{\mathrm{new}}(x_k) = \eta_{\mathrm{m}}(x_k). \tag{22}$$

Again, considering (21) and (22) results in

$$\left|\eta_{\mathrm{d}}^1(x_k)\right| \ll |\eta_{\mathrm{m}}(x_k)| \implies \eta_{\mathrm{m}}^{\mathrm{new}}(x_k) = \eta_{\mathrm{m}}(x_k),$$

from which implication (2) follows immediately. $\qquad\square$

With the three preceding lemmas at hand, we are now ready to state and prove our main theorem about finite termination of Alg. 1.

Theorem 1 (Finite termination). *Suppose that $\eta_{d,a} \ll \eta_{m,a}$ for every $a \in A_{\mathrm{pi}}$ and that every NLP is solved to local optimality. Then, Algorithm 1 terminates after a finite number of refinements, coarsenings and model switches in Lines 10, 11, 17, and 18 with an ε-feasible solution with respect to the reference problem (7) if there exist constants $C_1, C_2 > 0$ such that*

$$\frac{1}{2}\Theta_d^k \mu^k > \Phi_d^k + C_1, \quad \Theta_m^k \mu^k > \tau^k \Phi_m^k |A_{\mathrm{pi}}| + C_2$$

hold for all k.

Proof. We consider the total error $\sum_{a \in A_{\mathrm{pi}}} \eta_a$ and show that the difference between the decrease obtained in the inner loop and the increase obtained due to switching down the model level and coarsening the grid is positive and uniformly bounded away from zero for every iteration k. Again, we consider only a single iteration and

drop the corresponding index. We first consider Lines 10 and 11 for fixed j. It holds that

$$
\sum_{a \in A_{\mathrm{pi}}} \eta_a^{j-1} - \sum_{a \in A_{\mathrm{pi}}} \eta_a^j
$$

$$
= \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{m},a}^{j-1} + \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{m},a}^{j} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{j}
$$

$$
= \sum_{a \in A_{\mathrm{pi}} \setminus (\mathcal{U}_j \cup \mathcal{R}_j)} \eta_{\mathrm{m},a}^{j-1} - \sum_{a \in A_{\mathrm{pi}} \setminus (\mathcal{U}_j \cup \mathcal{R}_j)} \eta_{\mathrm{m},a}^{j} + \sum_{a \in \mathcal{U}_j \setminus \mathcal{R}_j} \eta_{\mathrm{m},a}^{j-1} - \sum_{a \in \mathcal{U}_j \setminus \mathcal{R}_j} \eta_{\mathrm{m},a}^{j}
$$

$$
+ \sum_{a \in \mathcal{R}_j \setminus \mathcal{U}_j} \eta_{\mathrm{m},a}^{j-1} - \sum_{a \in \mathcal{R}_j \setminus \mathcal{U}_j} \eta_{\mathrm{m},a}^{j} + \sum_{a \in \mathcal{R}_j \cap \mathcal{U}_j} \eta_{\mathrm{m},a}^{j-1} - \sum_{a \in \mathcal{R}_j \cap \mathcal{U}_j} \eta_{\mathrm{m},a}^{j}
$$

$$
+ \sum_{a \in A_{\mathrm{pi}} \setminus (\mathcal{U}_j \cup \mathcal{R}_j)} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in A_{\mathrm{pi}} \setminus (\mathcal{U}_j \cup \mathcal{R}_j)} \eta_{\mathrm{d},a}^{j} + \sum_{a \in \mathcal{U}_j \setminus \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in \mathcal{U}_j \setminus \mathcal{R}_j} \eta_{\mathrm{d},a}^{j}
$$

$$
+ \sum_{a \in \mathcal{R}_j \setminus \mathcal{U}_j} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in \mathcal{R}_j \setminus \mathcal{U}_j} \eta_{\mathrm{d},a}^{j} + \sum_{a \in \mathcal{R}_j \cap \mathcal{U}_j} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in \mathcal{R}_j \cap \mathcal{U}_j} \eta_{\mathrm{d},a}^{j}
$$

$$
= \sum_{a \in \mathcal{U}_j} \eta_{\mathrm{m},a}^{j-1} - \sum_{a \in \mathcal{U}_j} \eta_{\mathrm{m},a}^{j} + \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} - \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j}
$$

$$
= \sum_{a \in \mathcal{U}_j} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) + \frac{1}{2} \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1},
$$

where we use that $\eta_{\mathrm{m},a}^{j} = \eta_{\mathrm{m},a}^{j-1}$ for every $a \in \mathcal{R}_j \setminus \mathcal{U}_j$ since $\eta_{\mathrm{d},a}^{j-1} \ll \eta_{\mathrm{m},a}^{j-1}$ for every $a \in A_{\mathrm{pi}}$; see Lemma 3. Moreover, the discretization error estimator $\eta_{\mathrm{d},a}$ does not change after a switching up the model level.

Again, summing up over all $j = 1, \ldots, \mu$ yields the overall error decrease after μ for-loop iterations of

$$
\sum_{j=1}^{\mu} \left(\sum_{a \in A_{\mathrm{pi}}} \eta_a^{j-1} - \sum_{a \in A_{\mathrm{pi}}} \eta_a^j \right) = \sum_{a \in A_{\mathrm{pi}}} \eta_a^0 - \sum_{a \in A_{\mathrm{pi}}} \eta_a^{\mu}
$$

$$
= \sum_{j=1}^{\mu} \left(\sum_{a \in \mathcal{U}_j} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) + \frac{1}{2} \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} \right).
$$

With similar arguments as before for Lines 10 and 11 we consider Lines 17 and 18 and obtain

$$
\sum_{a \in A_{\mathrm{pi}}} \eta_a^{\mu+1} - \sum_{a \in A_{\mathrm{pi}}} \eta_a^{\mu}
$$

$$
= \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu+1} + \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{m},a}^{\mu+1} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu} - \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{m},a}^{\mu}
$$

$$
= \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu+1} - \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu} + \sum_{a \in \mathcal{D}} \eta_{\mathrm{m},a}^{\mu+1} - \sum_{a \in \mathcal{D}} \eta_{\mathrm{m},a}^{\mu}
$$

$$
= \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu} + \sum_{a \in \mathcal{D}} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu}).
$$

Finally, it remains to prove that

$$
\sum_{j=1}^{\mu} \left(\sum_{a \in \mathcal{U}_j} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) + \frac{1}{2} \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1} \right) - \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu} - \sum_{a \in \mathcal{D}} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu})
$$

is positive and uniformly bounded away from zero. Using the proofs of Lemmas 1 and 2 we have

$$\sum_{j=1}^{\mu} \sum_{a \in \mathcal{U}_j} (\eta_{\mathrm{m},a}^{j-1} - \eta_{\mathrm{m},a}^{j}) + \frac{1}{2} \sum_{j=1}^{\mu} \sum_{a \in \mathcal{R}_j} \eta_{\mathrm{d},a}^{j-1}$$

$$> \Theta_{\mathrm{m}} \mu \varepsilon + \frac{1}{2} \mu \Theta_{\mathrm{d}} \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu}$$

$$> \tau \Phi_{\mathrm{m}} |A_{\mathrm{pi}}| \varepsilon + C_2 \varepsilon + (\Phi_{\mathrm{d}} + C_1) \sum_{a \in A_{\mathrm{pi}}} \eta_{\mathrm{d},a}^{\mu}$$

$$> \sum_{a \in \mathcal{D}} (\eta_{\mathrm{m},a}^{\mu+1} - \eta_{\mathrm{m},a}^{\mu}) + C_2 \varepsilon + \sum_{a \in \mathcal{C}} \eta_{\mathrm{d},a}^{\mu} + C_1 |A_{\mathrm{pi}}| \varepsilon,$$

which completes the proof. $\qquad \square$

4.4. **Remarks.** Before we close this section we discuss some details and extensions regarding Alg. 1. First, we give an overview of the main computations that are performed in the algorithm. In Lines 2 and 12, the NLP (8) is solved using the current model level ℓ_a and the current stepsize h_a for every pipe $a \in A_{\mathrm{pi}}$. Most types of NLP algorithms are iterative methods. That is, the computational costs of the algorithms depend on the number of iterations required to converge to a (local) optimal solution and the costs per iteration. The latter mainly consist of the solution of a linear system (e.g., suitable forms of the KKT system for interior-point or active-set methods) for computing the search direction. The size of this linear system typically is $\mathcal{O}(n + m)$, where n is the number of variables and m is the number of constraints of the NLP. Both n and m are directly controlled by the stepsizes h_a that we use in our NLP models. The model level ℓ_a mainly determines the sparsity/density of the system matrices of the linear systems and the overall nonlinearity of the NLP, which typically influences the number of required iterations.

In Lines 3 and 13, the overall error estimator $\eta_a(y)$ is computed for every pipe $a \in A_{\mathrm{pi}}$. Thus, for all pipes, the solution of model (D_1) is computed with stepsize both $2h_a$ and $4h_a$ and the solution of model (D_{ℓ_a}) is computed with stepsize h_a. These solutions are obtained by solving the initial value problems consisting of the ordinary differential equations (M_1) and (M_{ℓ_a}) together with the initial value $p(x_0)$, which is contained in the optimal solution y of Problem (8). Continuing with the example of the implicit Euler method that we use as numerical integration scheme throughout this paper, the initial value problems can be solved (i) by considering the implicit equations in (D_1) and (D_{ℓ_a}) and using, e.g., the Newton method to solve for p_k in every space integration step or (ii) by using an existing software code and setting the order of the numerical integration scheme to one.

The subset $\mathcal{R}$ in Line 9 can be determined efficiently, since $\eta_{\mathrm{d},a}(y)$ has already been computed in Line 3 or 13 for every $a \in A_{\mathrm{pi}}$. For subset $\mathcal{U}$ in Line 9 and in (16) the error estimator $\eta_{\mathrm{m},a}(y)$ has also already been computed in Line 3 or 13 for every $a \in A_{\mathrm{pi}}$. Moreover, ℓ_a^{new} in (12) has to be computed in order to determine $\mathcal{U}$. For this, we compute $\eta_{\mathrm{m},a}(y; \ell_a - 1)$ if and only if $\ell_a = 3$. In the case $\ell_a = 2$ we have $\eta_{\mathrm{m},a}(y; \ell_a - 1) = 0$ and for $\ell_a = 1$ we have $\eta_{\mathrm{m},a}(y; \ell_a^{\mathrm{new}}) = \eta_{\mathrm{m},a}(y; \ell_a) = 0$. Subset $\mathcal{C}$ in Line 16 can also be computed efficiently, since $\eta_{\mathrm{d},a}(y)$ has already been computed in Line 3 or 13 for every $a \in A_{\mathrm{pi}}$. For subset $\mathcal{D}$ in Line 16 and in (18) the error estimator $\eta_{\mathrm{m},a}(y)$ has been computed already in Line 3 or 13 for every $a \in A_{\mathrm{pi}}$. If $\ell_a \in \{1, 2\}$, then $\eta_{\mathrm{m},a}(y; \ell_a + 1)$ has to be computed for every $a \in A_{\mathrm{pi}}$ in order to determine $\mathcal{D}$.

We note that the optimal solution y of Problem (8) contains, among others, the model level ℓ_a, stepsize h_a, and pressure $p^{\ell_a}(x_0)$ at the beginning of the pipe,

for every $a \in A_{\mathrm{pi}}$. Using ℓ_a, h_a, and $p^{\ell_a}(x_0)$, the discretization and model error estimator for pipe $a \in A_{\mathrm{pi}}$ can be computed without information from other pipes. Hence, the error estimators, e.g., in Line 13, can be computed in parallel.

Up to now, we have discussed two types of errors: modeling and discretization errors. Both are handled by Alg. 1 and we have shown that the algorithm terminates with a combined model and discretization error that satisfies a user-specified error tolerance $\varepsilon > 0$. What we have ignored so far is that the NLPs are also solved by a numerical method that introduces numerical errors as well. However, it is easy to integrate the control of this additional error source into Alg. 1. Let $\varepsilon^{\mathrm{opt}} > 0$ be the optimality tolerance that we hand over to the optimization solver and suppose that the solver always satisfies this tolerance. Furthermore, let the tolerance ε considered so far now be denoted by $\varepsilon^{\mathrm{dm}}$. Using the triangle inequality we easily see that the upper bound of the total error (that is aggregated modeling, discretization, and optimization error) is $\varepsilon^{\mathrm{opt}} + \varepsilon^{\mathrm{dm}}$. Hence, in order to satisfy an overall error tolerance $\varepsilon > 0$, we have to ensure that $\varepsilon^{\mathrm{opt}} + \varepsilon^{\mathrm{dm}} \leq \varepsilon$ holds, which can be formally introduced in Alg. 1 by replacing ε with $\varepsilon^{\mathrm{opt}} + \varepsilon^{\mathrm{dm}}$.

Finally, note that this additional error source directly suggests itself for adaptive treatment as well. In the early iterations of Alg. 1 it is not important that $\varepsilon^{\mathrm{opt}}$ is small. That is, the optimization is allowed to produce coarser approximate local solutions. However, in the course of the algorithm, one can observe the achieved modeling and discretization error and can adaptively tighten the optimization tolerance. Since this strategy allows the optimization method to produce coarse approximate solutions in the beginning, it can be expected that this leads to a speed-up in the overall running times of Alg. 1.

The choice of the error tolerance ε that has to be provided in Alg. 1 will depend on the user requirements, however, one should be aware that due to the round-off errors committed during every single step of the procedure, and due to possible ill-conditioning of the linear systems solved by the NLP solver, none of the three errors, the discretization error, the modeling error, and the NLP error can be chosen extremely small. Since the backward error and the associated condition number of the linear systems can be estimated during the procedure, see [17], and since the error estimates for the discretization method are at hand, it is just the modeling error which is not known a priori. To estimate this latter error (of the finest model) usually requires a comparison with experimental data. If these are available during a real-world process, then it is possible to adjust the required tolerances ε in a feedback loop using a standard PI controller, see, e.g., [30], i.e., if measured data are available that show that the finest model has a given accuracy, then ε should not be chosen smaller than this.

Finally, we want to stress that the described adaptive error control algorithm can be used with any number of model levels in the hierarchy, with any higher order discretization scheme, and with any number of grid refinement levels.

5. Computational Results

In this section we present numerical results obtained by the adaptive error control algorithm. To this end, we compare the efficiency of the method with an approach that directly solves an NLP that satisfies the same error tolerance and that is obtained without using adaptivity. Before we discuss the results in detail we briefly mention the computational setup and the gas transport network instances that we solve.

TABLE 1. Statistics for the instances

Network	# nodes	# pipes	# compressor stations	total pipe length (km)
GasLib-40	40	39	6	1112
GasLib-135	135	141	29	6935

We implemented the adaptive error control algorithm 1 in Python 2.7.13 and used the scipy 0.14.0 module for solving the initial value problems. All nonlinear optimization models have been implemented using the C++ framework LaMaTTO++[1] for modeling and solving mixed-integer nonlinear optimization problems on networks. The computations have been done on a six-core AMD Opteron$^{\text{TM}}$ Processor 2435 with 2.2 GHz and 64 GB of main memory. The NLPs have been solved using Ipopt 3.12; see [39, 40].

For our computational study, we choose publicly available GasLib instances; see [33]. This has the advantage that, if desired, all numerical results can be reproduced on the same data. In what follows, we consider the networks GasLib-40 and GasLib-135, since these are the largest networks in the GasLib that only contain pipes and compressor stations as arc types. Detailed statistics are given in Table 1.

Next, we describe the parameterization of Alg. 1. We initialize every pipe $a \in A_{\text{pi}}$ with the coarsest model level $\ell_a = 3$ and with the coarsest possible discretization grid. In order to yield a well-defined algorithm, the number of discretization grid intervals has to be a multiple of 4; see Fig. 2. Thus, we initially set $h_a = L_a/4$ and ensure in Step 18 of Alg. 1 that we never obtain a coarser grid size than the initial one. The overall tolerance is set to $\varepsilon = 10^{-4}$ bar. Moreover, we set $\Theta_{\text{d}} = \Theta_{\text{m}} = 0.7$, $\Phi_{\text{d}} = \Phi_{\text{m}} = 0.3$, $\tau = 1.1$, and $\mu = 4$. Here, we refrain from updating these parameters from iteration to iteration, which is possible in general. Note that our parameter choice violates the second inequality of Theorem 1. This could be fixed by simply increasing the hysteresis parameter μ. However, we refrain from using a larger μ in order to give the adaptive algorithm more chances to also switch down in the model hierarchy or to coarsen discretization grids. Our numerical experiments show that the violation of the second inequality of Theorem 1 does not harm convergence in practice but leads to slightly faster computations.

The same rationale holds for the relation between model and discretization error as assumed in Theorem 1; see also Lemma 3. To be fully compliant with the theory, the initial discretization grids need to be much finer. Again, coarser initial discretization grids do not harm convergence in our numerical experiments but yield much faster computations.

We now turn to the discussion of the numerical results. Both instances are solved using 8 iterations. Thus, together with the initially solved NLP, we have to solve 9 NLPs for solving both instances.

Using the adaptive control algorithm, it takes 3.82 s to solve the GasLib-40 instance and 7.50 s to solve the GasLib-135 instance. For the GasLib-40 network, the final NLP contains 2026 variables and 1988 constraints, whereas for the GasLib-135 the final NLP contains 3405 variables and 3271 constraints.

Most interesting is the speed-up that we obtain by using the adaptive control algorithm. Thus, we compare the above given solution times with the solution times for an NLP that satisfies the same error tolerances but that is obtained without using model level and discretization grid adaptivity. This NLP contains 40 034 variables and 39 996 constraints for the GasLib-40 instance and 144 757 variables as well as 144 623 constraints for the GasLib-135 instance. Compared to the final NLPs that
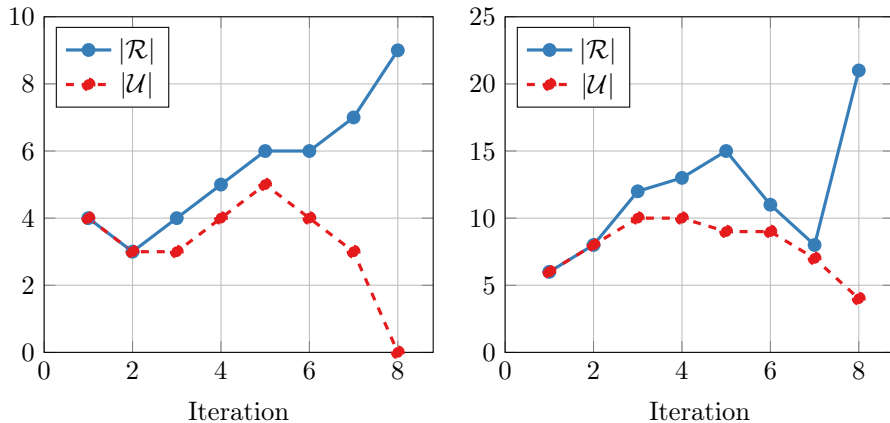
---

[1]http://www.mso.math.fau.de/edom/projects/lamatto.html

FIGURE 4. Number of pipes with refined grid (y-axis; $|\mathcal{R}|$) and number of pipes where the model is switched up in the model hierarchy (y-axis; $|\mathcal{U}|$) over the course of the iterations (x-axis). Left: GasLib-40, right: GasLib-135.

have to be solved within the adaptive algorithm, the NLPs obtained without using adaptivity are quite large scale. This directly translates to solution times. The GasLib-40 instance requires 53.11 s and the GasLib-135 instance requires 122.42 s. Thus, we get a speed-up factor of 13.89 and 16.33, respectively.

Figure 4 illustrates the adaptivity of the algorithm by plotting how many pipe grids are refined ($|\mathcal{R}|$) and how many pipe models are switched up in the hierarchy ($|\mathcal{U}|$). It can be clearly seen that increasing the accuracy is only needed for a small fraction of the pipes. For the GasLib-40 network, we never refine grids for more than 9 pipes, whereas we never refine grids for more than 21 pipes for the GasLib-135 network. Thus, for the larger network, we never refine grids for more than 15 % of all pipes.

For both networks, the Lines 17 and 18 are only reached once. For the smaller network, only 1 pipe grid is coarsened, whereas 3 pipe grids are coarsened for the larger network. Moreover, the algorithm never switches down in the model hierarchy. Consequently, the NLPs get larger from iteration to iteration. This then yields increased running times for the NLP solver as depicted in Fig. 5. It can be seen that the subsequent NLPs can be solved quite fast. There are two main reasons for this phenomenon. First, the NLP's size only increases moderately due to the adaptive control strategy. Second, the overall algorithm allows for warm-starting: When solving a single NLP we always use the last NLP's solution to set up the initial iterate.

Lastly, we consider the decrease in the respective errors. In Fig. 6, the discretization, model, and total errors are plotted over the course of the iterations. Both profiles show the expected decrease in the errors.

6. CONCLUSION

We have considered the problem of operation cost minimization for gas transport networks. In this context, we have focused on stationary and isothermal models and developed an adaptive model and discretization error control algorithm for nonlinear optimization that uses a hierarchy of continuous and finite-dimensional models. Out of this hierarchy, the new method adaptively chooses different models in order to finally achieve an optimal solution that satisfies a prescribed combined
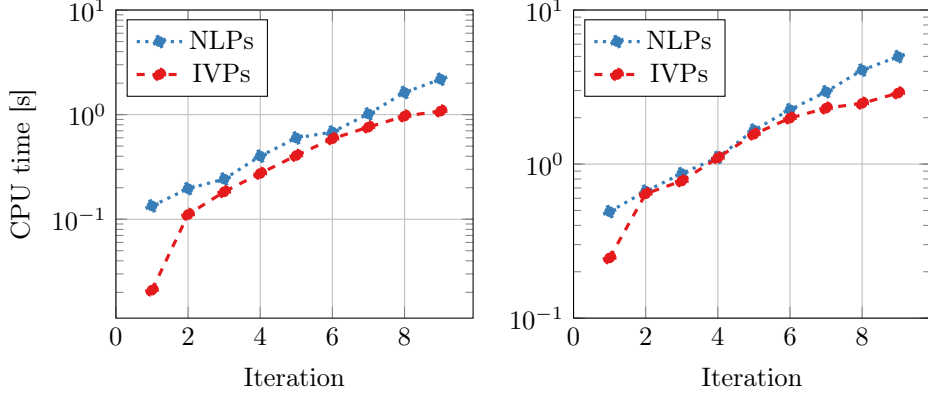
FIGURE 5. Aggregated run times (y-axis; in s) required for solving the nonlinear optimization problems (NLPs) and the initial value problems (IVPs) for the computation of the error estimates. Left: GasLib-40, right: GasLib-135.
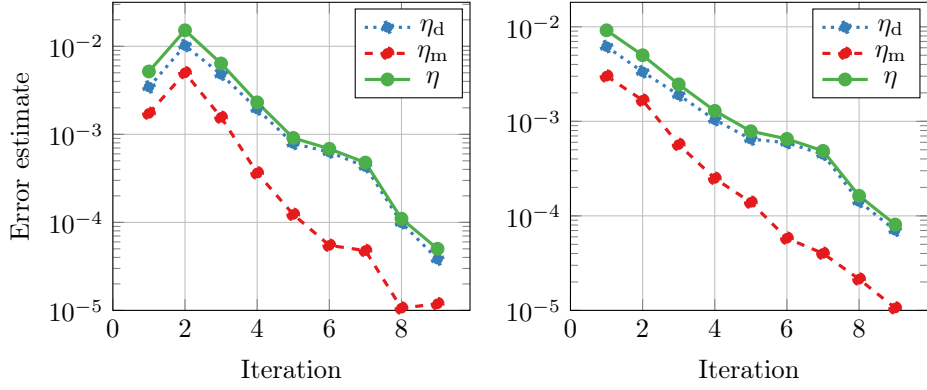

FIGURE 6. Discretization, model and total error estimates (y-axis) over the course of the iterations (x-axis). Left: GasLib-40, right: GasLib-135.

model and discretization error tolerance. The algorithm is shown to be convergent and its performance is illustrated by several numerical results.

The results pave the way for future work in the context of model switching and discretization grid adaptation for nonlinear optimal control. On the one hand, it should be extended to non-isothermal and instationary models of gas transport, in particular, in a port-Hamiltonian formulation. On the other hand, it would be interesting to extend the new technique to mixed-integer nonlinear optimal control.

ACKNOWLEDGEMENTS

REFERENCES

[1] R. Becker, H. Kapp, and R. Rannacher. Adaptive finite element methods for optimal control of partial differential equations: Basic concept. *SIAM Journal on Control and Optimization*, 39(1):113–132, 2000.

[2] L. T. Biegler. *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*, volume 10 of *MOS-SIAM Series on Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2010.

[3] H. G. Bock, M. Diehl, E. Kostina, and J. P. Schlöder. *Constrained Optimal Feedback Control of Systems Governed by Large Differential Algebraic Equations*, chapter 1, pages 3–24. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2007.

[4] S. Brenner and R. Scott. *The mathematical theory of finite element methods*, volume 15. Springer-Verlag New York, 2007.

[5] R. W. Brockett. *Finite dimensional linear systems*, volume 74 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2015.

[6] C. Carstensen and R. Hoppe. Convergence analysis of an adaptive nonconforming finite element method. *Numerische Mathematik*, 103(2):251–266, 2006.

[7] C. Carstensen and R. Hoppe. Error reduction and convergence for an adaptive mixed finite element method. *Mathematics of computation*, 75(255):1033–1042, 2006.

[8] M. Diehl, H. G. Bock, and J. P. Schlöder. A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on Control and Optimization*, 43(5):1714–1736, 2005.

[9] M. Diehl, H. Georg Bock, and J. P. Schlöder. Newton-type methods for the approximate solution of nonlinear programming problems in real-time. In Gianni Di Pillo and Almerico Murli, editors, *High Performance Algorithms and Software for Nonlinear Optimization*, pages 177–200. Springer US, Boston, MA, 2003.

[10] P. Domschke, A. Dua, J. J. Stolwijk, J. Lang, and V. Mehrmann. Adaptive refinement strategies for the simulation of gas flow in networks using a model hierarchy. Technical Report 2017/03, TU Berlin, Institut für Mathematik, 2017.

[11] P. Domschke, B. Hiller, J. Lang, and C. Tischendorf. Modellierung von Gasnetzwerken: Eine Übersicht. Technical report, Technische Universität Darmstadt, 2017.

[12] W. Dörfler. A convergent adaptive algorithm for poisson's equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124, 1996.

[13] M. Feistauer. *Mathematical Methods in Fluid Dynamics*, volume 67 of *Pitman Monographs and Surveys in Pure and Applied Mathematics Series*. Longman Scientific & Technical, Harlow, 1993.

[14] 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. [23], chapter 2, pages 17–44.

[15] B. Geißler, A. Martin, A. Morsi, and L. Schewe. Using piecewise linear functions for solving MINLPs. In *Mixed integer nonlinear programming*, volume 154 of *IMA Vol. Math. Appl.*, pages 287–314. Springer, New York, 2012.

[16] B. Geißler, A. Morsi, and L. Schewe. A new algorithm for MINLP applied to gas transport energy cost minimization. In *Facets of combinatorial optimization*, pages 321–353. Springer, Heidelberg, 2013.

[17] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[18] M. Gugat, F. M. Hante, M. Hirsch-Dick, and G. Leugering. Stationary states in gas networks. *Networks and Heterogeneous Media*, 10(2):295–320, 2015.

[19] M. Gugat, R. Schultz, and D. Wintergerst. Networks of pipelines for gas with nonconstant compressibility factor: stationary states. *Computational and Applied Mathematics*, 2016.

[20] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations I*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 1993.

[21] 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 Pammy Manchanda, René Lozi, and Abul Hasan Siddiqi, editors, *Industrial Mathematics and Complex Systems: Emerging Mathematical Models, Methods and Algorithms*, Industrial and Applied Mathematics, pages 77–122. Springer Singapore, Singapore, 2017.

[22] I. Joormann, M. Schmidt, M. C. Steinbach, and B. M. Willert. What does "feasible" mean? In Koch et al. [23], chapter 11, pages 211–232.

[23] T. Koch, B. Hiller, M. E. Pfetsch, and L. Schewe, editors. *Evaluating Gas Network Capacities*. SIAM-MOS series on Optimization. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2015.

[24] A. Kröner, K. Kunisch, and B. Vexler. Semismooth Newton methods for optimal control of the wave equation with control constraints. *SIAM J. Control Optimization*, 49(2):830–858, 2011.

[25] D. Leykekhman and B. Vexler. A priori error estimates for three dimensional parabolic optimal control problems with pointwise control. *SIAM J. Control Optimization*, 54(5):2403–2435, 2016.

[26] F. Liu, W. W. Hager, and A. V. Rao. Adaptive mesh refinement method for optimal control using nonsmoothness detection and mesh size reduction. *J. of the Franklin Institute*, 352(10):4081–4106, 2015.

[27] M. V. Lurie. *Modeling of Oil Product and Gas Pipeline Transportation*. Wiley-VCH, Weinheim, 2008.

[28] A. S. Morse, D. Q. Mayne, and G. C. Goodwin. Applications of hysteresis switching in parameter adaptive control. *IEEE Trans. Automat. Control*, 37(9):1343–1354, 1992.

[29] Z. Nagy, S. Agachi, F. Allgöwer, R. Findeisen, M. Diehl, H. G. Bock, and J. P. Schlöder. The tradeoff between modelling complexity and real-time feasibility in nonlinear model predictive control. In *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics, SCI*, 2002.

[30] P. Hr. Petkov, N. D. Christov, and M. M. Konstantinov. *Computational Methods for Linear Control Systems*. Prentice Hall International Ltd., Hertfordshire, UK, 1991.

[31] D. Rose, M. Schmidt, M. C. Steinbach, and B. M. Willert. Computational optimization of gas compressor stations: MINLP models versus continuous reformulations. *Mathematical Methods of Operations Research*, 83(3):409–444, 2016.

[32] L. Schewe, T. Koch, A. Martin, and M. E. Pfetsch. Mathematical optimization for evaluating gas network capacities. In *Evaluating gas network capacities*, volume 21 of *MOS-SIAM Ser. Optim.*, pages 87–102. SIAM, Philadelphia, PA, 2015.

[33] 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. *Data*, 2(4), 2017.

[34] M. Schmidt, M. C. Steinbach, and B. M. Willert. High detail stationary optimization models for gas networks. *Optimization and Engineering*, 16(1):131–164, 2015.

[35] M. Schmidt, M. C. Steinbach, and B. M. Willert. High detail stationary optimization models for gas networks: validation and results. *Optimization and Engineering*, 17(2):437–472, 2016.

[36] J. Stoer and R. Bulirsch. *Introduction to numerical analysis*. Springer-Verlag, New York-Heidelberg, 1980.

[37] J. J. Stolwijk and V. Mehrmann. Error analysis and model adaptivity for flows in gas networks. *Analele Stiintifice ale Universitatii Ovidius Constanta. Seria Matematica*, Accepted for publication, 2017.

[38] J. F. Wilkinson, D. V. Holliday, E. H. Batey, and K. W. Hannah. *Transient Flow in Natural Gas Transmission Systems*. American Gas Association, New York, 1964.

[39] A. Wächter and L. T. Biegler. Line search filter methods for nonlinear programming: Motivation and global convergence. *SIAM Journal on Optimization*, 16(1):1–31, 2005.

[40] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

[1]INSTITUT FÜR MATHEMATIK, MA 4-5, TU BERLIN, STRASSE DES 17. JUNI 136, 10623 BERLIN, GERMANY ({MEHRMANN,STOLWIJK}@MATH.TU-BERLIN.DE); [2]FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DISCRETE OPTIMIZATION, CAUERSTR. 11, 91058 ERLANGEN, GERMANY; [3]ENERGIE CAMPUS NÜRNBERG, FÜRTHER STR. 250, 90429 NÜRNBERG, GERMANY