Global optimization of mixed-integer ODE constrained network problems using the example of stationary gas transport

Oliver Habeck, Marc E. Pfetsch, Stefan Ulbrich
Research Group Optimization, Department of Mathematics,
TU Darmstadt, Germany
{habeck,pfetsch,ulbrich}@opt.tu-darmstadt.de

October 19, 2017

Abstract

In this paper we propose a new approach for finding global solutions of mixed-integer nonlinear optimization problems with ordinary differential equation constraints on networks. Instead of using a first discretize then optimize approach, we combine spatial and variable branching with appropriate discretizations of the differential equations to derive relaxations of the original problem. To construct the relaxations we derive convex under- and concave over-estimators for the ODE solution operators using numerical discretization schemes. Thereby, we make use of the underlying network structure, where the solutions of the ODEs only need to be known at a finite number of points. This property enables us to adaptively refine the discretization and relaxation without introducing new variables. The incorporation into a spatial branch-and-bound process allows to compute global ε-optimal solutions or decide infeasibility. We prove that this algorithm terminates finitely under some natural assumptions. We then show how this approach works for the example of stationary gas transport and provide some illustrative computational examples.

1 Introduction

In this paper we develop algorithms to globally solve nonlinear optimization problems with ordinary differential equation (ODE) constraints and an underlying network structure. More precisely, we consider problems of the following form:

$$\begin{aligned} \min \quad & C(x, y^0, y^S, z) \\ s.t. \quad & G(x, y^0, y^S, z) \leq 0, \\ & \partial_s y(s) = f\big(s, x, y(s)\big), \qquad s \in [0, S] \\ & y^0 = y(0),\ y^S = y(S), \\ & x \in X,\ y^0 \in Y^0,\ y^S \in Y^S,\ z \in Z, \end{aligned} \tag{$\mathcal{P}_{ode}$}$$

where $X \subset \mathbb{R}^k$ and Y^0, $Y^S \subset \mathbb{R}^n$ are polytopes and $Z \subset \mathbb{Z}^m$ is bounded. Furthermore, the objective function is $C\colon X \times Y^0 \times Y^S \times Z \to \mathbb{R}$ and (possibly nonlinear) constraints are given by $G\colon X \times Y^0 \times Y^S \times Z \to \mathbb{R}^l$. Thus, the variables $y(s)$ are functions that have to solve an ODEs specified by the function $f\colon \mathbb{R} \times \mathbb{R}^k \times \mathbb{R}^n \to \mathbb{R}^n$. Moreover, continuous variables x and integer variables z are present.

The distinguishing feature of $(\mathcal{P}_{ode})$ is that y only needs to be known at a finite number of positions, namely 0 and S. Only the corresponding values y^0 and y^S enter the remaining parts of $(\mathcal{P}_{ode})$ (namely C, G, Y^0, and Y^S). Note that for notational simplicity, we assume that the ODEs are defined on the same interval $[0, S]$; this can be assured by reparametrization. Moreover, we assume that C, G, and f are continuously differentiable.

The particular structure of $(\mathcal{P}_{ode})$ is motivated by stationary gas or water networks. In this case, the differential equations are composed by n one-dimensional ODEs $\partial_s y_i(s) = f_i\big(s, x, y_i(s)\big)$ for $i = 1, \ldots, n$, one for each of the n connections (pipelines) in the corresponding network. The relevant values are the (constant) flows for each connection and the values at the nodes (pressures). These values are coupled by $G(x, y^0, y^S, z) \leq 0$, which represents flow conservation and further network components (compressors/pumps, resistors, valves, ...). The integer variables are used to open or close valves or to turn compressors/pumps on or off. The objective often minimizes the energy for operating compressors/pumps.

We will present a solution method to globally solve $(\mathcal{P}_{ode})$ and we will use the example of stationary gas networks to illustrate the approach, but it can also be applied to stationary water networks.

The contribution of this paper is the development of a method to globally solve optimization problems of the form $(\mathcal{P}_{ode})$. The general approach is to use branch-and-bound to handle the integer variables z and spatial branching for handling nonlinearities. Both approaches are standard in mixed-integer nonlinear programming (MINLP), see, for example the books [22, 16, 19] or overview articles [7, 13]. Spatial branching refers to the technique in which the domain of a (continuous) variable is split into two (nonempty) parts, creating two new child nodes in the branch-and-bound tree. Since the bounds on the variable are tighter in each child node, the hope is that this can be used by other components to further tighten bounds. This process tends to produce better relaxations in the child nodes and hopefully results in a small/finite search tree.

The new contribution of this paper concerns the handling of the ODE constraints. We consider convex underestimators and concave overestimators for the (nonlinear) functions that map one boundary value $y(0)$ or $y(S)$ to the other. In Section 2, we show finite convergence of the corresponding method under natural assumptions. Moreover, a key point of our approach is that these estimators can be easily constructed by using basic discretization schemes. In Section 3, we show that this holds if some conditions are satisfied, for instance, the local truncation error is nonpositive. To demonstrate the approach, we apply the algorithm and estimation techniques to stationary gas transport in Section 4. We have also implemented our approach and provide computational experiments in Section 5.

1.1 Literature Review

The topic of this paper has clear connections to optimization with ODE or partial differential equation (PDE) constraints, see, e.g., [15] for a starting point into this area. Due to its analytical and computational complexity, the focus in this area often lies on the computation of local optima without the consideration of discrete decisions. However, in recent years there has been some effort to approach global and discrete decisions. We review some of the literature in this direction.

A very natural approach is to use the first discretize then optimize approach, i.e., to discretize the state space (usually time and spatial directions) in order to obtain a MINLP, which one typically tries to solve to global optimality; discrete decisions are then often handled by branch-and-bound. A number of articles use this approach – a partial list is as follows. Papamichail and Adjiman [29] consider parametric ODEs and construct approximations via the α-BB approach; the α-BB method for NLPs was introduced by Adjiman et al. [2, 1]. Sager et al. [34, 18] apply the convexification method of [33] to handle discrete decisions over time and show how to efficiently compute feasible solutions. An open source implementation of a general first discretize then optimize approach is available at [44] that uses relaxations based on piecewise-linearization, see Fügenschuh and Vierhaus [8] for a description. Bock et al. [3] consider problems in which discrete decisions depend on the state variables and present a reformulation/solution method for such problems. Diedam and Sager [6] compare single- and multiple-shooting discretizations for the global solution of optimal control problems without integer decisions.

Since the above mentioned approaches use a fixed discretization, the solutions only provide an approximation of the ODEs/PDEs with respect to an a priori fixed accuracy. Moreover, the corresponding MINLPs become very large for high precision. Thus, several approaches to adaptively refine the discretization have been developed. For instance, Sager et al. [33] present a convexification approach for the discrete decisions and use this to set up a solution method that can in principle determine a globally optimal solution. Buchheim et al. [4] present a global approach for solving elliptic mixed-integer PDE problems using outer approximation.

Global approaches are based on tight (convex) relaxations of the solution space. In [12], Hante and Sager extend the convexification approach from [33] to mixed-integer PDE problems and derive a relaxation. A series of publications consider methods to derive relaxations based on interval arithmetic. For instance, Nedialkov et al. [25] review methods for enclosing solutions of initial value problems. Singer and Barton [42, 43] consider relaxations for ODEs and apply Branch-and-Bound. Chachuat et al. [5] use an outer-approximation algorithm for mixed-integer dynamic optimization (MIDO). Scott and Barton [40] (improving [42] and [41]) consider ODEs and Scott, and Barton [39] deal with differential algebraic equations. Further envelopes for (parametric) ODEs are constructed by Neher et al. [26], Lin et al. [20, 21], Sahlodin et al. [35, 36], and Villanueva et al. [45].

As mentioned earlier, we will use stationary gas transport as a running example in this article. We refer to [30], [31], Ríos-Mercado and Conrado Borraz-Sánchez [32], Hante et al. [11] for general information on modelling of and solution methods for gas transport.

A related approach for solving mathematical optimization problems with ODEs in the context of gas transport is described in Gugat et al. [9]. Here, a global decomposition approach is described if the underlying network is a tree. Since in every iteration a mixed-integer linear master optimization problem is solved, this amounts to a "multi-tree" approach, while the method described in this paper works as a "single-tree". Related is the approach of Schmidt et al. [37], who consider the solution of MINLPs with equality constraints using univariate Lipschitz continuous functions for which the constants are known or approximated and the function evaluations may be approximate; this is applied to a stationary gas transport problem in which the underlying network is a tree. Moreover, Gugat et al. [10] present an instantaneous control approach for solving instationary gas transport problems, where a mixed-integer linear problem needs to be solved for each time step.

The technique that we present in this paper is distinct from the approaches mentioned above in the following way: We adaptively refine the discretization, which is not done in the approaches based on first discretize then optimize approaches. Moreover, our method to derive lower and upper bounds exploits the particular network structure and is different from the general-purpose approximations for ODEs and the convexifications mentioned above.

2 Solution Method

In this section we introduce our solution method. We begin with a natural basic assumption on the differential equations, which we assume to hold throughout the paper.

Assumption 1. *The initial value problem*

$$y(0) = y^0, \quad \partial_s y(s) = f\big(s, x, y(s)\big), \quad s \in [0, S] \tag{1}$$

is uniquely solvable for all $x \in X,\ y^0 \in Y^0$.

This assumption can be guaranteed, for example, if f is Lipschitz continuous w.r.t. y. We denote the solution operator by

$$F\colon X \times Y^0 \to \mathbb{R}^n, \quad \big(x, y^0\big) \mapsto y(S),$$

the unique solution of the initial value problem (1) for every $x \in X$ and $y^0 \in Y^0$. With this function we can replace the ODE constraints by

$$y^S - F\big(x, y^0\big) = 0.$$

This yields the equivalent problem

$$\begin{aligned} \min \quad & C(x, y^0, y^S, z) \\ s.t. \quad & G(x, y^0, y^S, z) \le 0, \\ & y^S - F\big(x, y^0\big) = 0, \\ & x \in X,\ y^0 \in Y^0,\ y^S \in Y^S,\ z \in Z. \end{aligned} \tag{$\mathcal{P}$}$$

Since X, Y^0, Y^S are polytopes, Z is bounded and C is continuous, the problem has an optimal solution if the feasible set is nonempty. If there is an analytical formula for F, then we could use spatial branch-and-bound to solve ($\mathcal{P}$). In the following, we assume that this is not the case or that the formula is hard to evaluate.

Our idea is to construct under- and overestimators $F(x, y^0)$ by the right choice of suitable numerical methods. That is, we choose one-step methods which provably yield lower and upper bounds for $y(s)$, respectively. We will later see an example for this approach. For now, we assume the existence of under- and overestimators.

Assumption 2. *There exist functions $F^\ell\colon X \times Y^0 \times \mathbb{N}^n \to \mathbb{R}^n$ and $F^u\colon X \times Y^0 \times \mathbb{N}^n \to \mathbb{R}^n$, which fulfill the inequality*

$$F^\ell\big(x, y^0, N\big) \le F\big(x, y^0\big) \le F^u\big(x, y^0, N\big)$$

for all $x \in X$ and $y^0 \in Y^0$. Furthermore, we assume that on the polytopes X, Y^0 the functions F_i^ℓ and F_i^u converge uniformly to F_i for $N_i \to \infty,\ i = 1, \ldots, n$.

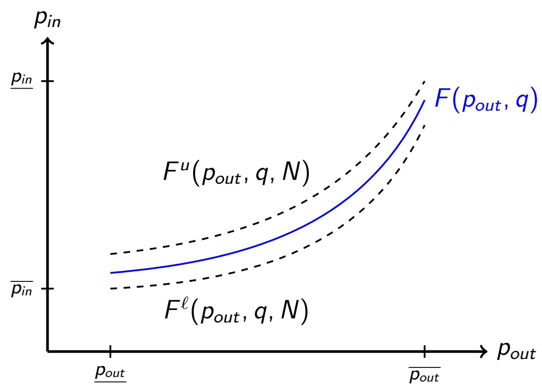

Figure 1: Example from stationary gas transport, see Section 4. Here, for one pipe, the inflow pressure is a convex function of the outflow pressure p_{out} and mass flow q (solid line, $F(p_{out}, q)$) and we can define F^ℓ and F^u (dashed lines) by suitable discretization methods.

For an example of this assumption see Figure 1, which is from our application, the stationary gas transport. There we consider the stationary isothermal Euler equation, which defines a relation between the pressure at the ends of a pipe and the flow. We can show, that given the pressure at the end and the mass flow, one can compute lower and upper bounds on the pressure at start. Thereby, N corresponds to the number of grid points in the discretization.

Next we relax the constraint

$$y^S = F(x, y^0)$$

of the problem $(\mathcal{P})$ by means of the functions F^ℓ and F^u. In this way we derive the relaxation

$$\begin{aligned} \min \quad & C(x, y^0, y^S, z) \\ s.t. \quad & G(x, y^0, y^S, z) \le 0, \\ & F^\ell\big(x, y^0, N\big) \le y^S \le F^u\big(x, y^0, N\big), \\ & x \in X,\ y^0 \in Y^0,\ y^S \in Y^S,\ z \in Z. \end{aligned} \tag{$\mathcal{P}_r(N)$}$$

This is a relaxation of $(\mathcal{P})$, since every feasible point of $(\mathcal{P})$ is feasible for the new constraint

$$F^\ell\big(x, y^0, N\big) \le y^S \le F^u\big(x, y^0, N\big),$$

and the objective function is the same. Note that the problem depends on $N \in \mathbb{N}^n$, as this constraint can vary for different N. Again, the optimal value of this problem is bounded from below, because X, Y^0, Y^S are polytopes, Z is bounded, and C is continuous.

In order to solve $(\mathcal{P}_r(N))$ with spatial branch-and-bound, we need a convex relaxation of the feasible set. Thus, we suppose that we can construct a convex underestimator $\check{F}^\ell$ of F^ℓ and a concave overestimator $\hat{F}^u$ of F^u for every $N \in \mathbb{N}^n$. In addition, let $\check{G}$ and $\check{C}$ be convex underestimators of G and C, respectively, for example, obtained by the αBB approach [2] or McCormick relaxations [24]. Then we obtain the following a convex relaxation of $(\mathcal{P}_r(N))$:

$$\begin{aligned} \min \quad & \alpha \\ s.t. \quad & \check{C}(x, y^0, y^S, z) - \alpha \le 0, \\ & \check{G}(x, y^0, y^S, z) \le 0, \\ & \check{F}^\ell\big(x, y^0, N\big) \le y^S \le \hat{F}^u\big(x, y^0, N\big), \\ & x \in X,\ y^0 \in Y^0,\ y^S \in Y^S,\ z \in \operatorname{conv}(Z). \end{aligned} \tag{$\mathcal{P}_{cv}(N)$}$$

Spatial branch-and-bound will enable us to compute so called (ε,δ)-*optimal solutions* of the relaxation $(\mathcal{P}_r(N))$. For a vector $y \in \mathbb{R}^n$ we denote with $(y)_+$ the vector of the componentwise maxima of y_i and 0.

Definition 1. *We say that a vector $(x, y^0, y^S, z) \in X \times Y^0 \times Y^S \times Z$ is a* δ-feasible solution *of $(\mathcal{P})$ if the condition*

$$\max\left\{\left\|\left(G(x,y^0,y^S,z)\right)_+\right\|_\infty,\ \left\|y^S - F(x,y^0)\right\|_\infty\right\} \le \delta$$

holds. Analogously, we call $(x, y^0, y^S, z) \in X \times Y^0 \times Y^S \times Z$ a δ-feasible solution *of $(\mathcal{P}_r(N))$ if*

$$\max\left\{\left\|\left(G\left(x,y^0,y^S,z\right)\right)_+\right\|_\infty,\ \left\|\left(F^\ell(x,y^0,N) - y^S\right)_+\right\|_\infty,\ \left\|\left(y^S - F^u\left(x,y^0,N\right)\right)_+\right\|_\infty\right\} \le \delta$$

holds. Furthermore, we call $(x, y^0, y^S, z) \in X \times Y^0 \times Y^S \times Z$ an (ε,δ)-optimal solution *of $(\mathcal{P})$ or $(\mathcal{P}_r(N))$ if it is δ-feasible and the objective function satisfies*

$$C(x,y^0,y^S,z) \le C^* + \varepsilon,$$

where $C^* > -\infty$ is the optimal value of $(\mathcal{P})$ or $(\mathcal{P}_r(N))$, or $C^* = \infty$ if their respective feasible set is empty.

Note that this definition is consistent with the definition in the literature, e.g., Locatelli and Schoen [22]. Since our goal is to find (ε,δ)-optimal solutions of $(\mathcal{P})$ by approximatively solving $(\mathcal{P}_r(N))$, we now show how their respective (ε,δ)-optimal solutions are related.

Lemma 2. *Let $(x, y^0, y^S, z) \in X \times Y^0 \times Y^S \times Z$ be an (ε,δ_1)-optimal solution of $(\mathcal{P}_r(N))$ for some $N \in \mathbb{N}^n$. Additionally, let the condition*

$$\left\|F^u(x,y^0,N) - F^\ell(x,y^0,N)\right\|_\infty \le \delta_2 \tag{2}$$

be satisfied for $\delta_2 \ge 0$. Then (x, y^0, y^S, z) is an (ε,δ)-optimal solution of $(\mathcal{P})$ for all $\delta \ge \delta_1 + \delta_2$.

Proof. First, we prove that (x, y^0, y^S, z) is δ-feasible for $(\mathcal{P})$. Because of the δ_1-feasibility for $(\mathcal{P}_r(N))$, we know that

$$\left\|\left(G(x,y^0,y^S,z)\right)_+\right\|_\infty \le \delta_1 \le \delta,$$

as well as

$$\left(y^S - F^u(x,y^0,N)\right)_+ + \left(F^\ell(x,y^0,N) - y^S\right)_+ \le \delta_1$$

holds. Here, we used that $F^u \ge F^\ell$ holds and for all $i = 1,\dots,n$ only one of $y_i^S - F_i^u(x,y^0,N)$ and $F_i^\ell(x,y^0,N) - y_i^S$ can be positive. Thus,

$$\begin{aligned}
\left|y_i^S - F_i(x,y^0)\right| &= \left(y_i^S - F_i(x,y^0)\right)_+ + \left(F_i(x,y^0) - y_i^S\right)_+ \\
&\le \left(y_i^S - F_i^u(x,y^0,N)\right)_+ + \left(F_i^u(x,y^0,N) - F_i(x,y^0)\right)_+ \\
&\qquad + \left(F_i(x,y^0) - F_i^\ell(x,y^0,N)\right)_+ + \left(F_i^\ell(x,y^0,N) - y_i^S\right)_+ \\
&= \left(y_i^S - F_i^u(x,y^0,N)\right)_+ + F_i^u(x,y^0,N) \\
&\qquad - F_i^\ell(x,y^0,N) + \left(F_i^\ell(x,y^0,N) - y_i^S\right)_+ \\
&\le \delta_1 + \delta_2 \le \delta.
\end{aligned}$$

That is, (x, y^0, y^S, z) is δ-feasible for $(\mathcal{P})$.

Next, let $(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z})$ be an optimal solution of $(\mathcal{P})$. As $(\mathcal{P}_r(N))$ is a relaxation of $(\mathcal{P})$, the solution is feasible for $(\mathcal{P}_r(N))$. Hence, there exists an optimal solution $(\tilde{x}, \tilde{y}^0, \tilde{y}^T, \tilde{z})$ of $(\mathcal{P}_r(N))$ with $C(\tilde{x}, \tilde{y}^0, \tilde{y}^T, \tilde{z}) \leq C(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z})$. Since (x, y^0, y^S, z) is an (ε, δ_1)-optimal solution of the relaxation $(\mathcal{P}_r(N))$, we can derive

$$C(x, y^0, y^S, z) \leq C(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}) + \varepsilon \leq C(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}) + \varepsilon,$$

that is, (x, y^0, y^S, z) is an (ε, δ)-optimal solution of $(\mathcal{P})$.

Otherwise, if $(\mathcal{P})$ is infeasible, the condition $C(x, y^0, y^S, z) \leq C^* + \varepsilon = \infty$ is obviously satisfied. □

Algorithm 1: Spatial branch-and-bound for $(\mathcal{P}_r(N))$

Input: Problem $(\mathcal{P}_r(N))$, $\delta > 0$ and $\varepsilon > 0$
Output: (ε, δ)-optimal solution $\big(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}\big)$ of $(\mathcal{P}_r(N))$ or "infeasible"

Upper bound $\mathcal{U} \leftarrow \infty$;
List of active nodes $\mathcal{L} \leftarrow \big\{X \times Y^0 \times Y^S \times Z\big\}$;
while $\mathcal{L} \neq \emptyset$ **do**
  Choose a node $\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z} \in \mathcal{L}$ and set $\mathcal{L} \leftarrow \mathcal{L} \setminus \{\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z}\}$;
  Build the convex relaxation $(\mathcal{P}_{cv}(N))$ w.r.t. $\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z}$;
  if $(\mathcal{P}_{cv}(N))$ *is feasible* **then**
    Let $\big(\tilde{\alpha}, \tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\big)$ be an optimal solution of $(\mathcal{P}_{cv}(N))$;
    if $\tilde{z} \in \mathbb{Z}^m$ **then**
      if $\big(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\big)$ *is δ-feasible for* $(\mathcal{P}_r(N))$ *and* $C\big(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\big) < \mathcal{U}$ **then**
        Set $\mathcal{U} \leftarrow C\big(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\big)$ and $\big(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}\big) \leftarrow \big(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\big)$;
    if $\tilde{\alpha} < \mathcal{U} - \varepsilon$ **then**
      Choose $\tilde{z}_i \notin \mathbb{Z}$ or $\tilde{x}_i, \tilde{y}_i^0, \tilde{y}_i^S$ appearing in a δ-violated constraint or in the ε-violated constraint $C\big(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\big) - \tilde{\alpha} > \varepsilon$;
      Perform branching w.r.t. the chosen variable $\tilde{x}_i$, $\tilde{y}_i^0$, $\tilde{y}_i^S$ or $\tilde{z}_i$ and add nodes to $\mathcal{L}$;

Lemma 2 shows how to generate an (ε, δ)-optimal solution of $(\mathcal{P})$. Since the functions F^ℓ and F^u are uniformly convergent to F, we can choose N such that condition (2) is satisfied for all $(x, y^0) \in X \times Y^0$. Then if the under- and overestimators fulfill some technical conditions, we can compute an (ε, δ_1)-optimal solution of $(\mathcal{P}_r(N))$ with spatial branch-and-bound and therefore an (ε, δ)-optimal solution of $(\mathcal{P})$. This idea yields Algorithm 1.

For proving that Algorithm 1 terminates, we require the following conditions. Suppose the algorithm produces (through branching) an infinite nested sequence of nodes $\mathcal{F}_k = X_k \times Y_k^0 \times Y_k^T \times Z_k$ with $\mathcal{F}_{k+1} \subseteq \mathcal{F}_k$ for all $k \in \mathbb{N}_0$. Then the branching rules have to satisfy the condition

$$\lim_{k \to \infty} \operatorname{diam}(\mathcal{F}_k) = 0, \tag{3}$$

where diam is the diameter of a set U:

$$\operatorname{diam}(U) := \max_{u, u' \in U} \|u - u'\|_2.$$

Next, for every $\mathcal{F}_k$ we need to be able to construct the convex underestimators $\check{C}$, $\check{G}$, $\check{F}^\ell$ and the concave overestimator $\hat{F}^u$ over the set $\mathcal{F}_k$. We denote the dependency on $\mathcal{F}_k$ by the index k, e.g., $\check{C}_k$. Furthermore, the estimators have to satisfy the condition

$$\max_{(x,y^0,y^S,z)\in\mathcal{F}_k} \Big\{ \|G(x,y^0,y^S,z) - \check{G}_k(x,y^0,y^S,z)\|_\infty,\ \|F^\ell(x,y^0,N) - \check{F}^\ell_k(x,y^0,N)\|_\infty, \\ \|\hat{F}^u_k(x,y^0,N) - F^u(x,y^0,N)\|_\infty,\ |C(x,y^0,y^S,z) - \check{C}_k(x,y^0,y^S,z)| \Big\} \to 0 \tag{4}$$

for $k \to \infty$.

Under these conditions, the next theorem proves that Algorithm 1 terminates finitely, see Locatelli and Schoen [22, Theorem 5.26]. Here, we assume for simplicity that Step 5 of Algorithm 1 can be executed exactly, i.e., without rounding errors, otherwise a further approximation error would have to be handled.

Theorem 3. *Let $\varepsilon > 0$, $\delta > 0$ hold. Suppose that the Conditions* (3) *and* (4) *are satisfied. Then Algorithm 1 terminates after a finite number of iterations and either returns an (ε, δ)-optimal solution of $(\mathcal{P}_r(N))$ or that $(\mathcal{P}_r(N))$ is infeasible.*

Note that there can exist (ε, δ)-optimal solutions even if $(\mathcal{P}_r(N))$ is infeasible. In this case, both results of the algorithm are possible. It can happen that Algorithm 1 finds an (ε, δ)-optimal solution or that δ-feasible solutions of $(\mathcal{P}_r(N))$ are infeasible for $(\mathcal{P}_{cv}(N))$ and the algorithm returns "infeasible". This is due to the fact that under- and overestimators are usually tight at some points and cut off δ-feasible solutions. For example, the McCormick estimators for the product of two variables over a square are exact in the corners.

Choosing N big enough, we can now compute (ε, δ)-optimal solutions of $(\mathcal{P})$.

Corollary 4. *Let $\varepsilon > 0$, $\delta > 0$ and suppose that Conditions* (3) *and* (4) *hold. Then we can compute an (ε, δ)-optimal solution of $(\mathcal{P})$ in finite time, or establish the infeasibility of* (1).

Proof. By Assumption 2 we know that F^ℓ and F^u converge uniformly to F for $N_i \to \infty,\ i = 1, \ldots, N$. Therefore, we can choose N such that

$$\left\| F^u\big(x, y^0, N\big) - F^\ell\big(x, y^0, N\big) \right\|_\infty \leq \frac{\delta}{2}$$

holds for all $x \in X$, $Y^0 \in Y^0$.

By Theorem 3 the spatial branch-and-bound algorithm with parameters $\frac{\delta}{2} > 0$ and $\varepsilon > 0$ returns an $(\varepsilon, \frac{\delta}{2})$-optimal solution of $(\mathcal{P}_r(N))$ or that it is infeasible.

Since $(\mathcal{P}_r(N))$ is a relaxation of $(\mathcal{P})$, if the algorithm returns "infeasible", there is no feasible solution of $(\mathcal{P})$. Otherwise, the algorithm returns an $(\varepsilon, \frac{\delta}{2})$-optimal solution of $(\mathcal{P}_r(N))$ and Lemma 2 states that this solution is an (ε, δ)-optimal solution of $(\mathcal{P})$. □

2.1 Adaptive spatial branch-and-bound

A disadvantage of Algorithm 1 is that we have to choose N in advance, such that Condition (2) holds on the whole feasible set. Hence, we might have to select N bigger than it has to be in some parts of the feasible set. This leads to more computational effort, for example, when N corresponds to the number of grid points of a discretization. To circumvent this problem, we replace Step 5 of Algorithm 1 by the following adaptive procedure.

We start with constructing $\check{C}$ and $\check{G}$ by standard methods on the current node. Next, we pick some initial convex relaxation of F^ℓ and F^u, e.g., the relaxation of the parent node during the branch-and-bound process or $X \times Y^0 \times Y^S \times \operatorname{conv}(Z)$ in the root node. Then we solve the convex relaxation. If the relaxation is infeasible, so is the corresponding original problem and we are done. Now, let $\left(\tilde{\alpha}, \tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ be the solution of the relaxation. We check if Condition (2) is satisfied in $\left(\tilde{x}, \tilde{y}^0\right)$ and possibly increase N until (2) holds. If $\left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ is a δ_1-feasible solution of

$$F^\ell\left(\tilde{x}, \tilde{y}^0, N\right) \leq \tilde{y}^S \leq F^u\left(\tilde{x}, \tilde{y}^0, N\right), \tag{5}$$

we stop with the current solution, otherwise we pick the most violated constraint, see Line 11. Then either $\tilde{y}_i^S > F_i^u(\tilde{x}, \tilde{y}^0, N) + \delta_1$, or $\tilde{y}_i^S < F_i^\ell(\tilde{x}, \tilde{y}^0, N) - \delta_1$ holds. Subsequently, we improve the under- or overestimator by cutting off the current solution. If this is not possible, we stop with the current solution and have to perform branching to resolve the infeasibility.

Algorithm 2: Adaptive convex relaxation

Input: Node of Branch-and-Bound tree $\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z}$, $\delta_1, \delta_2 > 0$, $N \in \mathbb{N}^n$, and convex underestimators $\check{C}$, $\check{G}$.

Output: A δ_1-feasible solution of (5), which satisfies (2), "infeasible" or instruction to branch.

1 Choose initial relaxation of (5) (e.g., $\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z}$ or relaxation of the parent node);
2 **for** $k = 1, 2, \ldots$ **do**
3   **if** $(\mathcal{P}_{cv}(N))$ *is feasible* **then**
4   Let $\left(\tilde{\alpha}, \tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ be a solution of the relaxation;
5   **while** $\left\|F^u\left(\tilde{x}, \tilde{y}^0, N\right) - F^\ell\left(\tilde{x}, \tilde{y}^0, N\right)\right\|_\infty > \delta_2$ **do**
6   Increase N_i for all i with $\left|F_i^u\left(\tilde{x}, \tilde{y}^0, N_i\right) - F_i^\ell\left(\tilde{x}, \tilde{y}^0, N_i\right)\right| > \delta_2$;
7   **if** $\left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ *is* δ_1*-feasible for* (5) **then**
8   Stop with solution $\left(\tilde{\alpha}, \tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$;
9   **else**
10   Choose "most violated" constraint i, i.e.,
11         $i \in \arg\max_{j=1,\ldots,n} \max\left\{F_j^\ell\left(\tilde{x}, \tilde{y}^0, N_j\right) - \tilde{y}_j^S,\ \tilde{y}_j^S - F_j^u\left(\tilde{x}, \tilde{y}^0, N_j\right)\right\}$;
12   **if** $\tilde{y}_i^S > F_i^u\left(\tilde{x}, \tilde{y}^0, N_i\right)$ **then**
13   "Improve the overestimator" or stop with the current solution and perform branching;
14   **else if** $\tilde{y}_i^S < F_i^\ell\left(\tilde{x}, \tilde{y}^0, N_i\right)$ **then**
15   "Improve the underestimator" or stop with the current solution and perform branching;
16   **else**
17   Stop with "infeasible";

What do we mean by Steps 13 and 15 of Algorithm 2? We can improve an estimator by adding a linear inequality, which cuts off the current LP-solution and is feasible for $(\mathcal{P}_r(N))$. For example in outer-approximation, current solutions can be cut off by gradient cuts. Another possibility is to add an estimator dynamically, instead of adding all inequalities at once. I.e., if an over- or underestimator consists of multiple inequalities, we only add an inequality if it cuts off the current solution.

Incorporating Algorithm 2 into the spatial branch-and-bound algorithm results in Algorithm 3. The main change is of course that N need not be constant any more. Note that a δ_1-feasible solution of $(\mathcal{P}_r(N))$ might not be a δ_1-feasible solution of $(\mathcal{P}_r(N'))$ for $N \leq N'$,

Algorithm 3: Adaptive spatial branch-and-bound for $(\mathcal{P})$

Input: Problem $(\mathcal{P})$, $N = N_0 \in \mathbb{N}^n$, $\delta_1, \delta_2 > 0$ and $\varepsilon > 0$
Output: $(\varepsilon, \delta_1 + \delta_2)$-optimal solution $\left(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}\right)$ or "infeasible"

Upper bound $\mathcal{U} \leftarrow \infty$;
List of active nodes $\mathcal{L} \leftarrow \left\{X \times Y^0 \times Y^S \times Z\right\}$;
while $\mathcal{L} \neq \emptyset$ **do**
  Choose a node $\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z} \in \mathcal{L}$ and set $\mathcal{L} \leftarrow \mathcal{L} \setminus \{\tilde{X} \times \tilde{Y}^0 \times \tilde{Y}^S \times \tilde{Z}\}$;
  Construct underestimators $\check{C},\ \check{G}$;
  Run Algorithm 2;
  if *Algorithm 2 stops with a solution of the relaxation* **then**
    Let $\left(\tilde{\alpha}, \tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ be the solution;
    if *the solution is* δ_1*-feasible for* (5) **then**
      if $\left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ *is a* δ_1*-feasible solution of* $(\mathcal{P}_r(N))$ *and* $C\left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right) < \mathcal{U}$ **then**
        Set $\mathcal{U} \leftarrow C\left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$ and $\left(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}\right) \leftarrow \left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right)$;
    if $\tilde{\alpha} < \mathcal{U} - \varepsilon$ **then**
      Choose $\tilde{z}_i \notin \mathbb{Z}$ or $\tilde{x}_i, \tilde{y}_i^0, \tilde{y}_i^S$ appearing in a δ_1-violated constraint $G \leq 0$ or in the possibly ε-violated constraint $C\left(\tilde{x}, \tilde{y}^0, \tilde{y}^S, \tilde{z}\right) - \tilde{\alpha} > \varepsilon$ or in the "most violated" constraint chosen in the last iteration of Algorithm 2;
      Perform branching w.r.t. the chosen variable $\tilde{x}_i$, $\tilde{y}_i^0$, $\tilde{y}_i^S$ or $\tilde{z}_i$ and add nodes to $\mathcal{L}$;

but still is a $\delta_1 + \delta_2$-feasible solution of $(\mathcal{P})$ if it fulfills Condition (2) for N. Therefore, this is an algorithm for solving $(\mathcal{P})$ and not $(\mathcal{P}_r(N))$.

Another big difference is that we do not have to reconstruct $\check{F}^\ell$ and $\hat{F}^u$ in every node, but instead refine them only if needed. Besides this, the algorithm is almost the same as before.

The crucial point in proving that Algorithm 3 terminates, certainly is that Algorithm 2 terminates after a finite number of iterations. As we can not prove this in general, but only for a given construction method of $\check{F}^\ell$ and $\hat{F}^u$, we need the following assumption.

Assumption 3. *If N is fixed, then Algorithm 2 terminates after finitely many iterations.*

Note that we do not suppose that the algorithm stops with a δ_1-feasible solution, it only has to stop with either a solution, "infeasible", or that branching has to be performed. The next Lemma shows that it is enough to demand that this assumption holds such that Algorithm 2 terminates after finitely many iterations.

Lemma 5. *Suppose that Assumption 3 holds. Then Algorithm 2 terminates finitely.*

Proof. Assume that the algorithm does not terminate. Then it produces a sequence of points which are feasible solutions of the convex relaxation but not δ_1-feasible for (5). We denote with $K \subset \mathbb{N}$ the iterations where N has to be increased.

Since F^ℓ and F^u converge uniformly to F w.r.t. N and $\tilde{X} \times \tilde{Y}^0$ is bounded, there exists a $N_0 \in \mathbb{N}^n$ such that

$$\left\|F^u\left(x, y^0, N'\right) - F^\ell\left(x, y^0, N'\right)\right\|_\infty \leq \delta_2$$

is satisfied for all $(x, y^0) \in \tilde{X} \times \tilde{Y}^0$ and all $N' \geq N^0$. That is, each N_i can only be increased a finite number of times until $N_i \geq N_i^0$ holds. Hence, K is either empty or a finite set. Then N

is fixed either from the beginning or after the last iteration $k \in K$. Due to Assumption 3 the algorithm stops after another finite number of iterations. □

Now, we can prove that Algorithm 3 terminates finitely. Again, we consider an infinite nested sequence of nodes $\mathcal{F}_k = X_k \times Y_k^0 \times Y_k^T \times Z_k$ with $\mathcal{F}_{k+1} \subseteq \mathcal{F}_k$ for all $k \geq 0$ produced by Algorithm 3. The branching rules still have to satisfy the condition

$$\lim_{k\to\infty} \operatorname{diam}(\mathcal{F}_k) = 0. \tag{3}$$

Since Algorithm 2 only improves the estimators if (5) is δ_1-violated in the current solution of the relaxation, it might happen that an estimator $\hat{F}^u$ or $\check{F}^\ell$ does not change although

$$\max_{(x,y^0,y^S,z)\in\mathcal{F}_k} \left\{ \|F^\ell(x,y^0,N) - \check{F}_k^\ell(x,y^0,N)\|_\infty, \|\hat{F}_k^u(x,y^0,N) - F^u(x,y^0,N)\|_\infty, \right\} > \delta_1$$

holds. Thus, (4) cannot hold either. Instead, the convex underestimators of C and G have to satisfy

$$\max_{(x,y^0,y^S,z)\in\mathcal{F}_k} \left\{ \|G(x,y^0,y^S,z) - \check{G}_k(x,y^0,y^S,z)\|_\infty,\ |C(x,y^0,y^S,z) - \check{C}_k(x,y^0,y^S,z)| \right\} \to 0 \tag{6}$$

for $k \to \infty$ and we assume that an iteration $k_0 \in \mathbb{N}$ exists such that the optimal solutions $\left(\tilde{\alpha}^k, \tilde{x}^k, \tilde{y}^{0,k}, \tilde{y}^{S,k}, \tilde{z}^k\right)_{k\in\mathbb{N}}$ of the relaxation $\left(\mathcal{P}_{cv}(N^k)\right)$ over $\mathcal{F}_k$ satisfy

$$\max\left\{ \left\|\left(F^\ell(\tilde{x}^k, \tilde{y}^{0,k}, N^k) - \tilde{y}^{S,k}\right)_+\right\|_\infty, \left\|\left(\tilde{y}^{S,k} - F^u(\tilde{x}^k, \tilde{y}^{0,k}, N^k)\right)_+\right\|_\infty \right\} \leq \delta_1 \tag{7}$$

for $k \geq k_0$. With that we have all the conditions and assumptions we need.

Theorem 6. *Suppose that Conditions* (3), (6) *and* (7)*, and the Assumptions 1,2 and 3 hold. Then Algorithm 3 terminates with an* $(\varepsilon, \delta_1 + \delta_2)$*-optimal solution of* $(\mathcal{P})$ *or "infeasible" after a finite number of nodes.*

Proof. Suppose that Algorithm 3 does not terminate. Then it produces at least one infinite nested sequence of nodes $\mathcal{F}_k = X_k \times Y_k^0 \times Y_k^T \times Z_k$ and a sequence $\left(\tilde{\alpha}^k, \tilde{x}^k, \tilde{y}^{0,k}, \tilde{y}^{S,k}, \tilde{z}^k\right)_{k\in\mathbb{N}}$, where each element is the solution of $\left(\mathcal{P}_{cv}(N^k)\right)$ over $\mathcal{F}_k$ during the last iteration of Algorithm 2. Note that the relaxation has to be feasible for every node, otherwise, the node would be pruned and the sequence $\mathcal{F}_k$ ends finitely.

We show that there is a $K \in \mathbb{N}$ such that $\left(\tilde{\alpha}^K, \tilde{x}^K, \tilde{y}^{0,K}, \tilde{y}^{S,K}, \tilde{z}^K\right)$ is a $(\delta_1 + \delta_2)$-feasible solution of $(\mathcal{P})$. By the Conditions (3) and (6) there exists an iteration $k_0 \in \mathbb{N}$ such that $|C(x,y^0,y^S,z) - \check{C}_{\mathcal{F}_k}(x,y^0,y^S,z)| < \varepsilon$ and $\|G(x,y^0,y^S,z) - \check{G}_{\mathcal{F}_k}(x,y^0,y^S,z)\|_\infty < \delta_1$ holds for all $\left(x,y^0,y^S,z\right) \in \mathcal{F}_k$ and all nodes $k \geq k_0$.

For the ODE-relaxation we again notice that there is a $N_0 \in \mathbb{N}^n$ such that Condition (2) is satisfied on the whole domain $X \times Y^0 \times Y^S \times Z$ and all $N \geq N_0$, because of the assumption that F^ℓ and F^u converge uniformly to F with respect to N. Therefore, at some iteration $k_1 \in \mathbb{N}$, N is increased for the last time. Then after $\max\{k_0, k_1+1\}$ nodes the only constraint which can be violated is

$$F^\ell\left(x,y^0,N\right) - \delta_1 \leq y^S \leq F^u\left(x,y^0,N\right) + \delta_1.$$

But by Condition (7), there is an iteration $k_2 \in \mathbb{N}$ such that this condition holds for all solutions $\left\{\left(\tilde{\alpha}^k, \tilde{x}^k, \tilde{y}^{0,k}, \tilde{y}^{S,k}, \tilde{z}^k\right)\right\}_{k\geq k_2}$ of $\left(\mathcal{P}_{cv}(N^k)\right)$ produced by Algorithm 2. Hence,

$(\tilde{x}^K, \tilde{y}^{0,K}, \tilde{y}^{S,K}, \tilde{z}^K)$ with $K = \max\{k_0, k_1, k_2\}$ is a $(\delta_1 + \delta_2)$-feasible solution of $(\mathcal{P})$. Thus, the upper bound $\mathcal{U}$ will be updated if $C(\tilde{x}^K, \tilde{y}^{0,K}, \tilde{y}^{S,K}, \tilde{z}^K) < \mathcal{U}$ holds and the node $\mathcal{F}_K$ will get fathomed, because

$$\tilde{\alpha}^K > C(\tilde{x}^K, \tilde{y}^{0,K}, \tilde{y}^{S,K}, \tilde{z}^K) - \varepsilon \geq \mathcal{U} - \varepsilon$$

is satisfied for $K \geq k_0$. That is, the algorithm does not produce an infinite sequence of nodes and, therefore, terminates finitely.

It remains to show that the output of the algorithm is correct. Suppose Algorithm 3 terminates with upper bound $\mathcal{U} = \infty$. This only happens if every node was fathomed because the relaxations are infeasible. Since the leaf nodes define a partition of the feasible set and the relaxations are infeasible, so has to be the original problem.

Suppose the algorithm terminates with a solution $(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z})$. By construction of the algorithm and Lemma 2, it is clear that the solution is $(\delta_1 + \delta_2)$-feasible for $(\mathcal{P})$. We distinguish two cases:

1. There is an optimal solution of $(\mathcal{P})$ with optimal value $C^* < \infty$.
2. The feasible set of $(\mathcal{P})$ is empty, i.e., $C^* = \infty$.

In the second case, clearly

$$C(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}) - \varepsilon \leq C^*$$

holds and $(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z})$ is ε-optimal. In the first case, let $\mathcal{F}_k = X_k \times Y_k^0 \times Y_k^T \times Z_k$ denote all nodes of the branch-and-bound tree which are fathomed due to $\alpha^k \geq \mathcal{U}^k - \varepsilon$ with optimal solution value α^k of the relaxation and current upper bound $\mathcal{U}^k$. Then $\bigcup_k \mathcal{F}_k$ defines a partition of the feasible set and $\min_k \alpha^k$ is a lower bound for C^*. With $C(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}) \leq \mathcal{U}^k$ we can derive $C(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}) - \varepsilon \leq \mathcal{U}^k - \varepsilon \leq \alpha^k$ and therefore the inequality

$$C(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z}) - \varepsilon \leq \min_k \alpha^k \leq C^*,$$

is true, i.e., $(\bar{x}, \bar{y}^0, \bar{y}^S, \bar{z})$ is ε-optimal. □

3 Existence of bounding schemes

In the previous section, it remained an open question when there exist functions F^ℓ and F^u, which satisfy Assumption 2. In this section, we will investigate how to define such functions based on suitable numerical methods for differential equations. Therefore, we consider an one-dimensional ODE

$$y(0) = y^0, \quad \partial_s y(s) = f(s, y(s)), \quad s \in [0, S]$$

and possibly implicit one-step methods which can be written in the form

$$y_0 = y^0, \quad y_{i+1} = y_i + h_i f_h(s_i, h_i, y_i, y_{i+1}), \quad \forall i = 0, \ldots, N-1, \tag{8}$$

subject to a discretization $0 = s_0 < s_1 < \cdots < s_N = S$. Define $h_i := s_{i+1} - s_i$ for all $i = 0, \ldots, N-1$. The idea is to define F^ℓ and F^u via the execution of a method of the form (8), i.e.,

$$F^\ell \colon y^0 \mapsto y_N \quad \text{or} \quad F^u \colon y^0 \mapsto y_N.$$

The goal is to derive lower and upper bounds on the exact solution $y(L)$ in this way. The *global truncation error* $e_N = y(S) - y_N$ might be a good indicator, but usual techniques only yield estimates on the absolute value of e_N. Therefore, we use the *local truncation error*

$$\tau(s,h) = y(s+h) - y(s) - h\, f_h\big(s, h, y(s), y(s+h)\big).$$

We consider an explicit method with nonnegative local truncation error, i.e.,

$$y(s_{i+1}) - y(s_i) - h_i\, f_h\big(s_i, h_i, y(s_i)\big) \geq 0 \tag{9}$$

for all $i = 0, \ldots, N-1$.

Example. From (9), we can immediately derive

$$y(s_1) - y_1 = y(s_1) - y(s_0) - h_0\, f_h\big(s_0, h_0, y(s_0)\big) \geq 0.$$

Nevertheless, this does not guarantee $y(s_2) - y_2 \geq 0$. For example, let $\partial_s y(s) = -y(s)$ and $y(0) = 1$. For this ODE the explicit Euler method (i.e., $f_h(s_i, h_i, y_i, y_{i+1}) = f(s_i, y_i)$) with equidistant step size has nonnegative truncation error and produces the solution $y_i = (1-h)^i$ for all i. Thus, with $h = 2$ we have $y_{2i} \leq y(s_{2i})$ and $y(s_{2i+1}) \leq y_{2i+1}$ for all i. Whereas, $y_i \leq y(s_i)$ with $0 < h \leq 1$ holds for all i.

This example suggests that a signed local truncation error and small step sizes are sufficient for producing lower and upper bounds. In fact, by (9) we have

$$y(s_{i+1}) \geq y(s_i) + h\, f_h\big(s_i, h, y(s_i)\big)$$

and the inequality

$$\geq \quad y_i \quad + h\, f_h\big(s_i, h, \quad y_i \quad\big) = y_{i+1}.$$

holds if $y + h\, f_h(s, h, y)$ is nondecreasing w.r.t. y. Therefore, we can derive $y(s_{i+1}) \geq y_{i+1}$.

Lemma 7. *Consider a method of the form* (8) *for a scalar ODE, i.e.,* $y(s) \in \mathbb{R}$*, and let the local truncation error of the method be nonnegative, i.e., the inequality*

$$y(s+h) - y(s) - h\, f_h\big(s, h, y(s), y(s+h)\big) \geq 0$$

holds for all $s \in [0, S]$ *and* $h \geq 0$ *with* $s + h \leq S$*. Suppose the derivatives satisfy*

$$b \leq \partial_y f_h(s, h, y, \tilde{y}) \quad \textit{and} \quad \partial_{\tilde{y}} f_h(s, h, y, \tilde{y}) \leq B$$

for constants b*,* $B \in \mathbb{R}$*. Then if*

$$0 < h_i \leq h_{max} = \begin{cases} \infty & \text{if } b \geq 0 \text{ and } B \leq 0, \\ \frac{1}{\max\{-b, B\}} & \text{otherwise}, \end{cases}$$

for all $i = 1, \ldots, N$*, the one-step method produces a lower bound on the solution* $y(s)$*, i.e.,*

$$y_i \leq y(s_i), \quad i = 1, \ldots, N.$$

If on the other hand the local truncation error of the method is nonpositive, i.e., the inequality

$$y(s+h) - y(s) - h\, f_h\big(s, h, y(s), y(s+h)\big) \leq 0$$

holds for all $s \in [0, S]$ *and* $h \geq 0$ *with* $s + h \leq S$*, then we obtain under the same assumptions*

$$y_i \geq y(s_i), \quad i = 1, \ldots, N.$$

Proof. We prove the lemma by induction on the number of grid points. Consider the function

$$R(s,h,y,\tilde{y}) = \tilde{y} - y - h\, f_h(s,h,y,\tilde{y}).$$

Then one step of (8) is given by $R(s_i,h_i,y_i,y_{i+1}) = 0$. By the assumption on the derivatives of f_h, we get

$$\partial_{\tilde{y}} R(s,h,y,\tilde{y}) = 1 - h\,\partial_{\tilde{y}} f_h(s,h,y,\tilde{y}) \geq 1 - h\,B.$$

Obviously, R is nondecreasing w.r.t. $\tilde{y}$ if $B \leq 0$ holds, or if all step sizes satisfy $h \leq \frac{1}{B}$. Since the local truncation error is nonnegative and $y_0 = y(0)$, we can derive the inequality

$$R\big(0,h_1,y(0),y(s_1)\big) \geq 0 = R\big(0,h_1,y_0,y_1\big).$$

Therefore, we gain the inequality

$$y(s_1) \geq y_1$$

if either $B \leq 0$ or $h_1 \leq \frac{1}{B}$ holds. We assume that $y(s_i) \geq y_i$ is satisfied. Since $\partial_y f_h$ is bounded by b, we know

$$\partial_y R(s,h,y,\tilde{y}) = -1 - h\,\partial_y f_h(s,h,y,\tilde{y}) \leq -1 - h\,b$$

holds. Thus, R is nonincreasing w.r.t. y if either $b \geq 0$ or $h \leq -\frac{1}{b}$ holds. Again, using that the local truncation error is nonnegative we derive

$$R\big(s_i,h_i,y(s_i),y(s_{i+1})\big) \geq 0 = R\big(s_i,h_i,y_i,y_{i+1}\big).$$

Furthermore, the monotonicity w.r.t. the third argument and $y(s_i) \geq y_i$ results in

$$R\big(s_i,h_i,y_i,y(s_{i+1})\big) \geq R\big(s_i,h_i,y(s_i),y(s_{i+1})\big) \geq R\big(s_i,h_i,y_i,y_{i+1}\big)$$

and consequently

$$y(s_{i+1}) \geq y_{i+1}$$

holds. Then induction yields that the one-step method produces a lower bound on $y(S)$.

The case of nonpositive truncation error can be treated in the same way. □

Note that for most one-step methods like the Runge-Kutta methods, the assumption on the right-hand side f of the ODE to be Lipschitz-continuous already ensures that the partial derivatives $\partial_y f_h$ and $\partial_{\tilde{y}} f_h$ are bounded.

Remark. If we consider an explicit one-step method, i.e., $f_h(s,h,y,\tilde{y})$ is independent of $\tilde{y}$, that is $y_{i+1} = y_i - h_i\, f_h(s_i,h_i,y_i)$, then the previous lemma yields that we can choose

$$h_{\max} = \begin{cases} \infty & \text{if } b \geq 0, \\ -\frac{1}{b} & \text{else.} \end{cases}$$

Remark. If we consider an "end value problem" instead of an initial value problem, that is $\partial_s y(s) = f(s,y(s))$ holds for $s \in [0,S]$ and $y(S) = y^S$, then Lemma 7 still holds true with the modification, that the bounds are now reversed, i.e., positive truncation error now yields upper bounds and negative truncation error now yields lower bounds.

Remark. In the autonomous case (f_h is independent of s), the sign condition of Lemma 7 for the local truncation error is also necessary in the following sense. If there exist s, $y(s)$ and $\bar{h} > 0$ such that (9) does not hold for all $0 < h \leq \bar{h}$, then the scheme (8) does not produce lower bounds for all initial data y^0. In fact, in the autonomous case, without restriction, there exists $y(0) = y^0$ such that (9) does not hold at $s = 0$ for all $0 < h \leq \bar{h}$. Now for $0 < h_1 < \bar{h}$ small enough, a strict version of the second assertion of Lemma 7 is applicable, yielding $y_1 > y(h_1)$.

For the case of a system of ODEs, we have the following analogue of Lemma 7 that can be proven in a similar way.

Lemma 8. *Consider a method of the form* (8) *for a system of ODEs with* $f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$. *Let the local truncation error of the method be nonnegative, i.e., the inequality*

$$y(s+h) - y(s) - h\, f_h\big(s, h, y(s), y(s+h)\big) \geq 0$$

holds for all $s \in [0, S]$ *and* $h \geq 0$ *with* $s+h \leq S$. *Define with* $\partial_y f_h(s, h, y, \tilde{y})$ *and* $\partial_{\tilde{y}} f_h(s, h, y, \tilde{y})$ *the mean value derivatives*

$$\hat{D}_{y/\tilde{y}} f_h(s, h, y, \tilde{y}, z, \tilde{z}) = \int_0^1 \partial_{y/\tilde{y}} f_h(s, h, y + \tau(z-y), \tilde{y} + \tau(\tilde{z} - \tilde{y}))\, \mathrm{d}\tau.$$

Suppose there are $h_{max} > 0$ *and* $d_{max} > 0$ *such that*

$$\Big(I - h\hat{D}_{\tilde{y}} f_h(s, h, y(s), y(s+h), z, \tilde{z})\Big)^{-1} \Big(I + h\hat{D}_y f_h(s, h, y(s), y(s+h), z, \tilde{z})\Big)$$

has nonnegative entries for all $0 < h \leq h_{max}$, $s \in [0, S-h]$, $\|z - y(s)\| \leq d_{max}$, *and* $\|\tilde{z} - y(s+h)\| \leq d_{max}$. *Then for all* $0 < h \leq h_{max}$ *such that the solution of the scheme* (8) *satisfies* $\|y_i - y(s_i)\| \leq d_{max}$, $i = 0, \ldots, N$, *one has*

$$y_i \leq y(s_i), \quad i = 1, \ldots, N.$$

If on the other hand the local truncation error is nonpositive then we obtain under the same assumptions

$$y_i \geq y(s_i), \quad i = 1, \ldots, N.$$

It still remains to formulate sufficient conditions for the existence of one-step methods, which have nonpositive or nonnegative local truncation error.

Example. Suppose that $\partial_s y(s)$ is concave. In the autonomous case, i.e., $f\big(s, y(s)\big) = f\big(y(s)\big)$, this holds if $f\big(y(s)\big) \leq 0$, $\partial_{yy} f\big(y(s)\big) \leq 0$ for all $s \in [0, S]$, and if f is Lipschitz-continuous with respect to y. Then the inequality

$$\begin{aligned} y(\tilde{s}+h) - y(\tilde{s}) &= \int_{\tilde{s}}^{\tilde{s}+h} f\big(s, y(s)\big)\, \mathrm{d}s \\ &\geq \int_{\tilde{s}}^{\tilde{s}+h} f\big(\tilde{s}, y(\tilde{s})\big) + \frac{f\big(\tilde{s}+h, y(\tilde{s}+h)\big) - f\big(\tilde{s}, y(\tilde{s})\big)}{h}(s - \tilde{s})\, \mathrm{d}s \\ &= \frac{h}{2}\big[f\big(\tilde{s}+h, y(\tilde{s}+h)\big) + f\big(\tilde{s}, y(\tilde{s})\big)\big] \end{aligned}$$

holds for all $h > 0$. Hence, the trapezoidal rule with $f_h(s_i, h_i, y_i, y_{i+1}) = \frac{1}{2}\big[f(s_i, y_i) + f(s_{i+1}, y_{i+1})\big]$ has nonnegative truncation error. Since f is Lipschitz-continuous, we derive that the conditions of Lemma 7 are satisfied. Thus, the trapezoidal rule produces a lower bound on the solution $y(S)$.

4 Application – Stationary Gas Transport

Within this section we show that our solution method can be applied to the example of stationary gas transport.

4.1 The Model

Let a gas network be given by a directed graph $\mathcal{G} = \big(\mathcal{V}, \mathcal{A}\big)$, where the nodes in $\mathcal{V}$ are entries, exits and junctions of the network, and $\mathcal{A}$ are the network elements like valves, resistors, compressors and pipes. As a basis for the gas flow we consider the stationary isothermal Euler-equations, which is a one-dimensional ordinary differential equation in space. In the following we concentrate on the differential equation, whereas we refer to [31, Chapter 6] for the models of the other network elements.

In terms of pressure p and mass flow rate q the differential equation reads

$$\partial_x p(x) \left(1 - \frac{c^2 q^2}{A^2 p(x)^2}\right) = -\frac{\lambda c^2}{2DA^2} q|q| \frac{1}{p(x)} - \frac{g}{c^2} s_{lope}\, p(x), \quad 0 < x < L. \tag{10}$$

The ODE describes the pressure of gas flowing along a single cylindrical pipe of length L. Note that in stationary gas transport, the mass flow rate is constant along each pipe, but is a variable in our consideration. The constants are the speed of sound c, the cross sectional area of the pipe A, the friction coefficient λ, the diameter of the pipe D, the gravitational acceleration g, and the slope s_{lope} of the pipe.

For simplicity, we assume the pipe to be horizontal, i.e., $s_{lope} = 0$. Furthermore, we assume that the gas travels with subsonic velocity v, for instance, we can require $\frac{v}{c} \leq 0.8$. In practice this is no limitation, since the gas velocity is typically much less than the speed of sound. Thus, using the relation $q = A\rho v$ of mass flow rate and velocity with density ρ and the relation $\rho c^2 = p$, we derive $\frac{v}{c} = \frac{c|q|}{Ap} \leq 0.8$. With these assumptions, we can rewrite (10) to

$$\partial_x p(x) = -\frac{\lambda\, c^2\, q|q|\, p(x)}{2D\big(A^2 p(x)^2 - c^2 q^2\big)}.$$

Here, we only used $c\, q < A\, p$, which is implied by $\frac{v}{c} \leq 0.8$, see also Note 13.

We now define the right-hand side as a function of pressure and mass flow:

$$\varphi(p, q) := -\frac{\lambda\, c^2\, q|q|\, p}{2D(A^2 p^2 - c^2 q^2)}. \tag{11}$$

With this, we can now formulate our optimization model:

$$\begin{aligned}
\min \quad & C(p, q, z) \\
\text{s.t.} \quad & G(p, q, z) \leq 0, \\
& \partial_x p_a(x) = \varphi_a\big(p_a(x), q_a\big) \quad \forall\, a \in \mathcal{A}_{pipe} \subseteq \mathcal{A}, \\
& p_u = p_a(0),\ p_v = p_a(L_a) \quad \forall\, a = (u, v) \in \mathcal{A}_{pipe}, \\
& p \in P,\ q \in Q,\ z \in Z.
\end{aligned} \tag{12}$$

Here, we have pressure variables p_v for all nodes $v \in \mathcal{V}$, functions $p_a(x)$ for each pipe $a \in \mathcal{A}_{pipe} \subseteq \mathcal{A}$, flow variables q_a for all edges $a \in \mathcal{A}$, and binary variables $z \in \{0, 1\}^m$ for the discrete decisions like to open or close a valve, or turn a compressor on or off. The sets $P \subset \mathbb{R}^{\mathcal{V}}$ and $Q \subset \mathbb{R}^{\mathcal{A}}$ are given by variable bounds $0 < \underline{p}_v \leq p_v \leq \overline{p}_v$ for all $v \in \mathcal{V}$ and $\underline{q}_a \leq q_a \leq \overline{q}_a$ for all $a \in \mathcal{A}$, respectively. Just like before, the ODEs only define a coupling between the pressure and flow variables for a single pipe, whereas C, G only depend on the pressure at the nodes. Furthermore, note that φ depends on $a \in \mathcal{A}_{pipe}$ since the pipes might differ in length, diameter or friction coefficient.

The constraint $G(p, q, z) \leq 0$ represents the models for the different network elements, as well as the flow conservation given by

$$\sum_{a \in \delta^+(v)} q_a - \sum_{a \in \delta^-(v)} q_a = q_v^{\pm} \tag{13}$$

for all nodes $v \in \mathcal{V}$, where $\delta^+(v)$ denotes the outgoing arcs of v, $\delta^-(v)$ denotes the ingoing arcs and $q_v^{\pm}$ is the in- or outflow at node v. Furthermore, $G(p, q, z) \leq 0$ contains the inequalities

$$\begin{aligned} 5\,c\,q_a - 4A\,p_v &\leq 0, \\ -5\,c\,q_a - 4A\,p_u &\leq 0 \end{aligned}$$

for all pipes $a = (u, v)$ such that the subsonic flow condition is satisfied. Note that it is sufficient to demand $5\,c\,|q| \leq 4A\,p$ at the end of the pipe where the gas flows out, since the pressure drops in the direction of the flow.

Like in the abstract problem setting, we suppose that in the branch-and-bound framework $C(p, q, z)$ and $G(p, q, z)$ can be treated by standard techniques, such that we can focus on the differential equations.

4.2 Lower and Upper Bounds for the Inflow Pressure

Within this subsection we interpret the differential equation as an initial value problem with the pressure at the end of the pipe as “initial” value. We will derive lower and upper bounds on the inflow pressure. Therefore, we assume from now on that the flow is nonnegative, i.e., $q \geq 0$. During the branch-and-bound process we can ensure this by branching on the flow w.r.t. $q = 0$.

Let $\mathcal{D} := \{(p, q) \in \mathbb{R}^2 \mid 0 < \underline{p} \leq p \leq \overline{p},\ 0 \leq \underline{q} \leq q \leq \overline{q},\ 5c\,q \leq 4A\,p\}$ be the domain of φ given by variable bounds and the subsonic flow condition. Simple differentiating and computing the eigenvalues of the Hessian yields the following properties of φ.

Lemma 9. *The function $\varphi\colon \mathcal{D} \to \mathbb{R}$ is nondecreasing in p, nonincreasing in q, and concave in $(p, q) \in \mathcal{D}$. The second derivatives satisfy $\partial_{pp}\varphi(p, q) \leq 0$, $\partial_{pq}\varphi(p, q) \geq 0$ and $\partial_{qq}\varphi(p, q) < 0$. Furthermore,*

$$\varphi(p, q) = \partial_p\varphi(p, q) = \partial_q\varphi(p, q) = \partial_{pp}\varphi(p, q) = \partial_{pq}\varphi(p, q) = 0$$

holds, if and only if $q = 0$.

This leads to the following properties of the differential equation.

Corollary 10. *The ordinary differential equation*

$$p(L) = p^0, \quad \partial_x p(x) = \varphi\big(p(x), q\big), \quad 0 \leq x \leq L \tag{14}$$

has a unique solution $p(x)$ for all $(p^0, q) \in \mathcal{D}$. Furthermore, $p(x)$, as well as $\partial_x p(x)$, is nonincreasing and concave.

Proof. For $q = 0$ the ODE has the only solution $p(x) = p^0$, because $\varphi(p, 0) = 0$ holds. For fixed $q > 0$, the right-hand side $\varphi\big(p(x), q\big)$ is negative, i.e., $p(x)$ is nonincreasing and the pressure is bounded from below by p^0. Thus, $\partial_p\varphi(p,q)$ is bounded by $\partial_p\varphi\big(\frac{5cq}{4A},q\big) \geq \partial_p\varphi(p,q) > 0$, i.e., φ is Lipschitz-continuous w.r.t. p. Hence, there is a unique solution of the ODE.

The remaining statements follow by differentiating $\partial_x p(x) = \varphi\big(p(x),q\big)$ and using the properties of φ. $\square$

Next, we show that the explicit midpoint method and the implicit trapezoidal rule define lower and upper bounds on $p(0)$, as illustrated in Figure 1. Therefore, let a discretization $0 = x_N < \ldots < x_1 < x_0 = L$ be given. For simplicity, we assume that discretization is equidistant with step size $h = \frac{L}{N}$. Then the explicit midpoint method is defined by

$$p_0^\ell = p^0, \quad p_{i+1}^\ell = p_i^\ell - h\,\varphi\big(p_i^\ell - \tfrac{h}{2}\varphi(p_i^\ell,q),q\big), \qquad \forall i = 0,\ldots,N-1, \tag{15a}$$

and the implicit trapezoidal rule is

$$p_0^u = p^0, \quad p_{i+1}^u = p_i^u - \tfrac{1}{2}h\left[\varphi(p_i^u,q) + \varphi(p_{i+1}^u,q)\right], \quad \forall i = 0,\ldots,N-1. \tag{15b}$$

Note that we apply the methods in opposite direction of the flow such that we can guarantee the existence of solutions p_N^ℓ and p_N^u with $5c\,q \leq 4A\,p_N^\ell$ and $5c\,q \leq 4A\,p_N^u$.

In order to make use of Lemma 7, we write the schemes in the form of (8). Therefore, we define

$$\varphi_h^\ell\big(x_i,h,p_i^\ell,p_{i+1}^\ell\big) := -\varphi\big(p_i^\ell - \tfrac{h}{2}\varphi(p_i^\ell,q),q\big)$$

and

$$\varphi_h^u\big(x_i,h,p_i^u,p_{i+1}^u\big) := -\tfrac{1}{2}\varphi(p_i^u,q) - \tfrac{1}{2}\varphi(p_{i+1}^u,q).$$

To keep the notation simple, we leave out the unnecessary variables, i.e., we write $\varphi_h^\ell(h,p_i^\ell)$ and $\varphi_h^u(p_i^u,p_{i+1}^u)$. With that, we can prove that the methods yield lower and upper bounds on $p(0)$.

Corollary 11. *The explicit midpoint method* (15a) *with step size* $0 < h \leq \frac{81}{328}\frac{D}{\lambda}$ *defines a lower bound on the solution* $p(0)$ *of* (14).

Proof. We have to show that $\partial_{p_i}\varphi_h^\ell$ is bounded from below and that the local truncation error of (15a) is nonnegative. Then Lemma 7 and its Remark for "end value problems" show that the inequality

$$p(0) \geq p_N^\ell$$

holds.

First we show that the derivative is bounded. Differentiating φ_h^ℓ yields

$$\partial_{p_i}\varphi_h^\ell\big(h,p_i\big) = -\partial_{p_i}\varphi\big(p_i - \tfrac{h}{2}\varphi(p_i,q),q\big)\Big(1 - \tfrac{h}{2}\partial_{p_i}\varphi(p_i,q)\Big).$$

For $q = 0$ we have $\partial_{p_i}\varphi_h^\ell = 0$, since $\partial_p\varphi(p,0) = 0$ holds. Otherwise, we can choose h such that $1 > 1 - \frac{h}{2}\partial_p\varphi(p,q) > 0$ holds. For $(p,q) \in \mathcal{D}$ we know

$$1 - \tfrac{h}{2}\partial_p\varphi\big(p,q\big) \geq 1 - \tfrac{h}{2}\partial_p\varphi\big(\tfrac{5}{4}\tfrac{cq}{A},q\big) = 1 - h\,\tfrac{164}{81}\tfrac{\lambda}{D},$$

i.e., we have to choose $h < \frac{81}{164}\frac{D}{\lambda}$. Hence, with $-\partial_{p_i}\varphi \leq 0$ we can derive

$$0 > \partial_{p_i}\varphi_h^\ell\big(h,p_i\big) = -\partial_{p_i}\varphi\big(p_i - \tfrac{h}{2}\varphi(p_i,q),q\big)\Big(1 - \tfrac{h}{2}\partial_{p_i}\varphi(p_i,q)\Big) > -\partial_{p_i}\varphi\big(p_i - \tfrac{h}{2}\varphi(p_i,q),q\big).$$

Furthermore, $p - \frac{h}{2}\varphi(p,q) > p \geq \frac{5}{4}\frac{cq}{A}$ and $-\partial_{pp}\varphi(p,q) \geq 0$ holds and, thus,

$$-\partial_{p_i}\varphi\big(p_i - \tfrac{h}{2}\varphi(p_i,q),q\big) > -\partial_{p_i}\varphi\big(p_i,q\big) \geq -\partial_{p_i}\varphi\big(\tfrac{5}{4}\tfrac{cq}{A},q\big) = -\tfrac{328}{81}\tfrac{\lambda}{D},$$

i.e., all in all we obtain that $\partial_{p_i}\varphi_h^\ell$ is bounded by

$$0 \geq \partial_{p_i}\varphi_h^\ell\big(h,p_i\big) \geq -\tfrac{328}{81}\tfrac{\lambda}{D}.$$

Next, we show that the local truncation error is nonnegative. Using concavity of $\partial_x p(x)$, we obtain the inequality

$$\begin{aligned}
p(x_i) - p(x_{i+1}) &= \int_{x_{i+1}}^{x_i} \partial_x p(x)\,\mathrm{d}x \\
&\leq \int_{x_{i+1}}^{x_i} \partial_x p\big(\tfrac{x_i+x_{i+1}}{2}\big) + \partial_{xx} p\big(\tfrac{x_i+x_{i+1}}{2}\big)\,\big(x - \tfrac{x_i+x_{i+1}}{2}\big)\,\mathrm{d}x \\
&= h\,\partial_x p\big(\tfrac{x_i+x_{i+1}}{2}\big) + \partial_{xx} p\big(\tfrac{x_i+x_{i+1}}{2}\big)\underbrace{\int_{x_{i+1}}^{x_i} \big(x - \tfrac{x_i+x_{i+1}}{2}\big)\,\mathrm{d}x}_{=0} \\
&= h\,\varphi\big(p\big(\tfrac{x_i+x_{i+1}}{2}\big),q\big).
\end{aligned}$$

Then using concavity of $p(x)$, we get $p\big(\frac{x_i+x_{i+1}}{2}\big) \leq p(x_i) - \frac{h}{2}\varphi\big(p(x_i),q\big)$, and with $\partial_p\varphi \geq 0$ we can derive

$$p(x_{i+1}) - p(x_i) \geq -h\,\varphi\big(p\big(\tfrac{x_i+x_{i+1}}{2}\big),q\big) \geq -h\,\varphi\big(p(x_i) - \tfrac{h}{2}\varphi\big(p(x_i),q\big),q\big) = h\,\varphi_h^\ell\big(h,p(x_i)\big).$$

Therefore, the local truncation error is nonnegative and we can apply Lemma 7, which proves that the explicit midpoint method produces a lower bound on $p(0)$. □

Corollary 12. *The trapezoidal rule* (15b) *with step size* $0 < h \leq \frac{81}{164}\frac{D}{\lambda}$ *defines an upper bound on the solution* $p(0)$ *of* (14).

Proof. As in the previous proof, we have to show that the derivatives of φ_h^u are bounded and that the local truncation error of the trapezoidal rule is nonpositive.

We have already seen in Example 3 that the inequality

$$p(x+h) - p(x) - \frac{h}{2}\,\big(\varphi\big(p(x),q\big) + \varphi\big(p(x+h),q\big)\big) \geq 0$$

holds, since $\partial_x p(x)$ is concave. Thus, we get $p(x_i) - p(x_{i+1}) + h\,\varphi_h^u\big(p(x_i),p(x_{i+1})\big) \geq 0$, i.e., the local truncation error is nonpositive.

Next, we show that the derivatives are bounded. Differentiating φ_h^u yields $\partial_{p_i}\varphi_h^u(p_i,p_{i+1}) = -\frac{1}{2}\partial_{p_i}\varphi(p_i,q)$ and $\partial_{p_{i+1}}\varphi_h^u(p_i,p_{i+1}) = -\frac{1}{2}\partial_{p_{i+1}}\varphi(p_i,q)$. Therefore, it suffices to show that $\partial_p\varphi(p,q)$ is bounded.

For $q = 0$ we have $\partial_p\varphi(p,q) = 0$. Otherwise, if $q > 0$, we know that $\partial_p\varphi$ is positive and decreasing w.r.t. p. Thus, for $(p,q) \in \mathcal{D}$ we obtain

$$0 < \partial_p\varphi\big(p,q\big) \leq \partial_p\varphi\big(\tfrac{5}{4}\tfrac{cq}{A}\big) = \tfrac{328}{81}\tfrac{\lambda}{D},$$

i.e., we get $-\frac{164}{81}\frac{\lambda}{D} \leq \partial_{p_i}\varphi_h^u(p_i,p_{i+1}) \leq 0$ and $-\frac{164}{81}\frac{\lambda}{D} \leq \partial_{p_{i+1}}\varphi_h^u(p_i,p_{i+1}) \leq 0$. Hence, by choosing $h \leq \frac{81}{164}\frac{D}{\lambda}$ and applying Lemma 7 we see that the trapezoidal rule produces an upper bound on $p(0)$. □

Note 13. *Note that for Corollaries 11 and 12 to hold, it is essential that* $\frac{cq}{Ap}$ *has an upper bound that is strictly lower than* 1*. Otherwise, the derivatives* $\partial_{p_i}\varphi_h^\ell$ *and* $\partial_{p_i}\varphi_h^u$ *would not be bounded from below, and we could not apply Lemma 7.*

With these two schemes in mind, we define two functions $p^\ell, p^u \colon \mathcal{D} \times \mathbb{N} \to \mathbb{R}$ through the computation of (15a) and (15b). That is

$$p^\ell(p, q, N) := p_N^\ell \qquad \text{and} \qquad p^u(p, q, N) := p_N^u,$$

where $p_0^\ell = p_0^u = p$.

We can derive the following properties for p^ℓ and p^u.

Lemma 14. *Let N be big enough such that the condition $h = \frac{L}{N} \leq 0.16\frac{D}{\lambda}$ holds. Then the functions p^ℓ and p^u are nondecreasing, continuously differentiable w.r.t. pressure and mass flow, and convex in (p, q). Furthermore, every solution $\bar{p}(x)$ of the differential equation $\partial_x p(x) = \varphi\big(p(x), q\big)$ with $q \geq 0$ satisfies the inequality*

$$p^\ell\big(\bar{p}(L), q, N\big) \leq \bar{p}(0) \leq p^u\big(\bar{p}(L), q, N\big). \tag{16}$$

Proof. The inequalities in (16) follow from Corollaries 11 and 12.

To show the differentiability, monotonicity and convexity of p^ℓ, $p^u : \mathcal{D} \to \mathbb{R}$, we only discuss the more involved case of the midpoint rule (15a). With $p_0^\ell = p_0^\ell(p, q) = p$, we can write (15a) as

$$p_{i+1}^\ell(p, q) = F^\ell\big(p_i^\ell(p, q), q, h\big) := p_i^\ell(p, q) - h\,\varphi\big(p_i^\ell(p, q) - \tfrac{h}{2}\varphi(p_i^\ell(p, q), q), q\big), \quad i = 0, \ldots, N-1.$$

Differentiating yields $\partial_p\, p_0^\ell(p, q) = 1$, $\partial_q\, p_0^\ell(p, q) = 0$ and

$$\begin{aligned}
\partial_p\, p_{i+1}^\ell(p, q) &= \partial_p F^\ell\big(p_i^\ell(p, q), q, h\big)\ \partial_p\, p_i^\ell(p, q),\\
\partial_q\, p_{i+1}^\ell(p, q) &= \partial_p F^\ell\big(p_i^\ell(p, q), q, h\big)\ \partial_q\, p_i^\ell(p, q) + \partial_q F^\ell\big(p_i^\ell(p, q), q, h\big),
\end{aligned}$$

where $\partial_p F^\ell$ and $\partial_q F^\ell$ denotes the partial derivative of F^ℓ with respect to the first and second argument, respectively. Moreover, we have $D^2 p_0^\ell(p, q) = 0$ and

$$D^2 p_{i+1}^\ell(p, q) = D\begin{pmatrix} p_i^\ell(p, q) \\ q \end{pmatrix}^{\top} D^2 F^\ell\big(p_i^\ell(p, q), q, h\big) D\begin{pmatrix} p_i^\ell(p, q) \\ q \end{pmatrix} + \partial_p F^\ell\big(p_i^\ell(p, q), q, h\big) D^2 p_i^\ell(p, q).$$

Hence, we obtain by induction that $\partial_p\, p_{i+1}^\ell(p, q) \geq 0$ and $D^2 p_{i+1}^\ell(p, q)$ is positive semidefinite, if $\partial_p F^\ell\big(p_i^\ell(p, q), q, h\big) \geq 0$ and $D^2 F^\ell\big(p_i^\ell(p, q), q, h\big)$ is positive semidefinite. Moreover, if in addition $\partial_q F^\ell\big(p_i^\ell(p, q), q, h\big) \geq 0$ holds then also $\partial_q\, p_{i+1}^\ell(p, q) \geq 0$ follows.

Since $\varphi(p, q) \leq 0$ on $\mathcal{D}$, we obtain by (15a) that $p_{i+1}^\ell(p, q) \geq p_i^\ell(p, q)$ and thus $(p_i^\ell(p, q), q) \in \mathcal{D}$ for $i = 0, \ldots, N$. Moreover, (15a) yields

$$\partial_p F^\ell(p_i^\ell, q, h) = 1 - h\,\partial_p\,\varphi\big(p_i^\ell - \tfrac{h}{2}\varphi(p_i^\ell, q), q\big)\big(1 - \tfrac{h}{2}\partial_p\,\varphi(p_i^\ell, q)\big).$$

By Lemma 9 and its proof we have $\partial_p\varphi \geq 0$, $\partial_{pp}\varphi \leq 0$ on $\mathcal{D}$ and $1 - \frac{h}{2}\partial_p\,\varphi(p_i^\ell, q) \geq 0$ for $0 < h \leq \frac{81D}{164\lambda}$. This shows that

$$\partial_p F^\ell(p_i^\ell, q, h) \geq 1 - h\,\partial_p\,\varphi\big(p_i^\ell - \tfrac{h}{2}\varphi(p_i^\ell, q), q\big) \geq 1 - h\,\partial_p\,\varphi(p_i^\ell, q) \geq 0 \quad \forall\, 0 < h \leq \tfrac{81D}{328\lambda}.$$

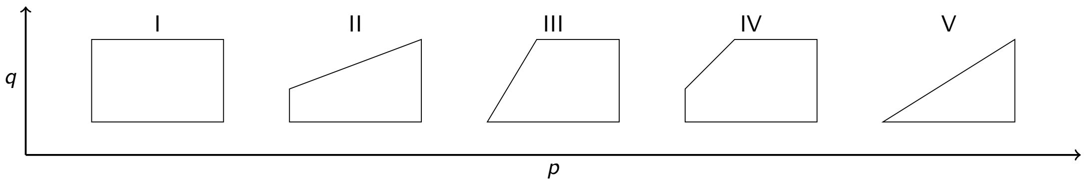

Figure 2: Five possible shapes of the domain $\mathcal{D}$ of the functions p^ℓ and p^u.

Moreover, one can verify that $D^2F^\ell(p_i^\ell, q)$ is singular and is thus positive semidefinite on $\mathcal{D}$ if $\partial_{pp}F^\ell(p_i^\ell, q) \geq 0$ on $\mathcal{D}$.

To show the latter, we observe that $\partial_{pp}F^\ell(p_i^\ell, q, h)$ is a rational function in p_i^ℓ and h with positive denominator on $\mathcal{D}$. The numerator is a polynomial in p_i^ℓ whose value and all its derivatives are nonnegative at $p_i^\ell = \frac{5cq}{4A}$ for all $0 < h \leq 0.16\frac{D}{\lambda}$. Hence, the numerator – and thus $\partial_{pp}F^\ell(p_i^\ell, q, h)$ – is nonnegative for all $(p_i^\ell, q) \in \mathcal{D}$. As already observed, this implies by induction that $\partial_p p_{i+1}^\ell(p, q) \geq 0$ and that $D^2 p_{i+1}^\ell(p, q)$ is positive semidefinite for all $(p, q) \in \mathcal{D}$.

Finally, $\partial_p F^\ell(p, 0, h) = 0$ and $\partial_{qq}F^\ell(p, q, h) \geq 0$ on $\mathcal{D}$, since $D^2F^\ell(p, q, h)$ is positive semidefinite on $\mathcal{D}$. Thus also $\partial_q F^\ell(p_i^\ell(p, q), q, h) \geq 0$ and we deduce $\partial_q p_{i+1}^\ell(p, q) \geq 0$.

The proof for the trapezoidal rule (15b) is similar, but easier. □

The functions p^ℓ and p^u can be used to relax the ODE constraints, compare $(\mathcal{P}_r(N))$. For deriving a convex relaxation we only have to construct a concave overestimator of p^u since p^ℓ is already convex. The domain $\mathcal{D}$ of p^ℓ and p^u is given by the intersection of the box $[\underline{p}, \overline{p}] \times [\underline{q}, \overline{q}]$ with nonnegative lower bounds and the inequality $5c\,q \leq 4A \leq p$. The resulting possible shapes are shown in Figure 2. As the concave envelope of a convex function over a polytope is defined by the vertices of the polytope (see, e.g., [16, Theorem IV.6]), it consists of up to two linear inequalities in the cases I-III, up to three inequalities in the case IV, and only one inequality in case V.

Consequently, choosing the number of grid points N_a for each pipe $a \in \mathcal{A}$ sufficiently big such that the condition

$$\left| p_a^u(p, q, N_a) - p_a^\ell(p, q, N_a) \right| \leq \delta_1$$

is fulfilled (compare with Lemma 2), we can apply Algorithm 1 and produce (ε, δ)-optimal solutions for the problem (12).

4.3 Adaptive LP-Relaxation

For our implementation we had to circumvent the problem, that the definition of p^ℓ is not given by an explicit formula. Furthermore, we use an LP-based branch-and-bound method. Therefore, we designed an adaptive approach, which has the advantage that the evaluation of p^ℓ and p^u is only needed "on demand". Furthermore, the number of grid points need not satisfy the condition (2) up front for the whole domain, but only in specific points.

Within the branch-and-bound process, we approximate p^ℓ with an outer-approximation approach. Moreover, we do not add the complete concave overestimator of p^u at once. Instead, inequalities are added dynamically if they cut off the current solution of the relaxation.

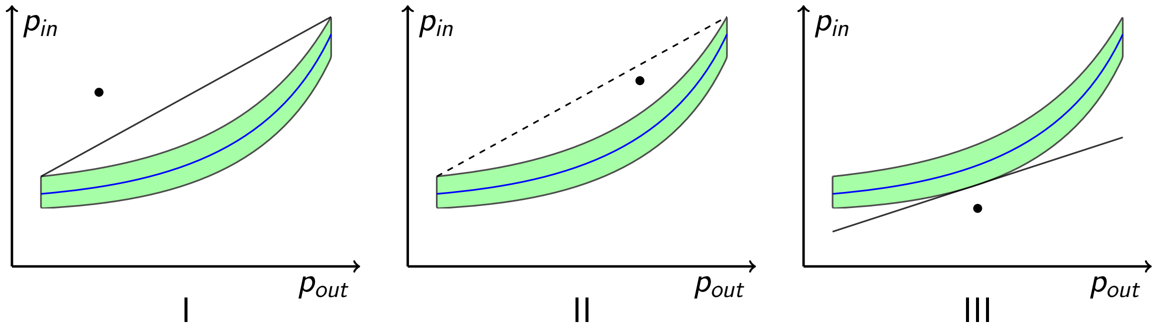

Figure 3: Three different cases of infeasibility of a pair (p_{in}, p_{out}) for fixed mass flow rate.

We proceed as in Algorithm 2, but adjusted to the gas model, see Algorithm 4. In the root node of the branch-and-bound tree we start with the minimal number of grid points N_a, such that the condition $\frac{L_a}{N_a} \leq 0.16\frac{D_a}{\lambda_a}$ is satisfied for all pipes $a \in \mathcal{A}_{pipe}$, and $P \times Q$ as the initial relaxation of the differential equation. In every other node, we use the relaxation of the parent node. Next, we solve the LP-relaxation. Thus, we obtain a triple $(\tilde{p}_{in}, \tilde{p}_{out}, \tilde{q})$ for every pipe $a \in \mathcal{A}_{pipe}$. We then compute $p_a^\ell(\tilde{p}_{out}, \tilde{q}, N_a)$ and $p_a^u(\tilde{p}_{out}, \tilde{q}, N_a)$. If the difference does not satisfy (2), we increase N_a and recompute p_a^ℓ, p_a^u until they do (Lines 7 and 8). We then check whether the triples $(\tilde{p}_{in}, \tilde{p}_{out}, \tilde{q})$ are feasible for

$$p_a^\ell(\tilde{p}_{out}, \tilde{q}, N_a) \leq \tilde{p}_{in} \leq p_a^u(\tilde{p}_{out}, \tilde{q}, N_a). \tag{17}$$

If all triples are feasible, Algorithm 4 returns the current solution to the branch-and-bound process. In the case that at least one triple is not feasible, we pick the pipe a with the greatest deviation of $\tilde{p}_{in}$ from the next bound p_a^ℓ, p_a^u, see Line 12. Next, we check whether the direction of the flow is fixed on this pipe. If not, we perform branching with respect to q_a to fix the flow direction (Line 14); this step is not performed in the general Algorithm 2, but particular to our context here. If the flow direction is already fixed, we try to cut off the solution. Thereby, we distinguish three different cases, as shown in Figure 3.

In the first case (Line 16), $\tilde{p}_{in}$ is greater than the concave envelope of p_a^u. Here, we add a linear inequality, which separates $\tilde{p}_{in}$ from the feasible region. In the second case, we have $p_a^u(\tilde{p}_{out}, \tilde{q}, N_a) \leq \tilde{p}_{in} \leq \hat{p}_a^u(\tilde{p}_{out}, \tilde{q}, N_a)$, i.e., we cannot cut off the current solution with the concave envelope. Instead, we have to resolve the infeasibility by branching. In the last case, (Line 18), when $\tilde{p}_{in}$ is less than $p_a^\ell(\tilde{p}_{out}, \tilde{q}, N_a)$, we make use of the convexity of p_a^ℓ and cut off the solution with a gradient cut

$$p_{in} \geq p_a^\ell(\tilde{p}_{out}, \tilde{q}, N_a) + \nabla p_a^\ell(\tilde{p}_{out}, \tilde{q}, N_a)^\top \begin{pmatrix} p_{out} - \tilde{p}_{out} \\ q - \tilde{q} \end{pmatrix}.$$

In the first and last case, we then iterate and again solve the relaxation. In the third case, Algorithm 4 stops and instructs the branch-and-bound process to perform branching. After the convex relaxation algorithm terminates with a solution, we carry on like in the spatial branch-and-bound Algorithm 3.

We now show that Assumption 3 holds for this example.

Lemma 15. *Suppose that the vector of grid points $N \in \mathbb{N}^{\mathcal{A}_{pipe}}$ stays constant during the execution of Algorithm 4, i.e., the condition*

$$\|p^u(p_{out}^k, q^k, N) - p^\ell(p_{out}^k, q^k, N)\|_\infty < \delta_2$$

Algorithm 4: Adaptive convex relaxation of gas flow

Input: Node of branch-and-bound tree, $\delta_1, \delta_2 > 0$ and $N = N_0 \in \mathbb{N}^{\mathcal{A}_{pipe}}$.

Output: δ_1-feasible solution of the Euler-equation, "infeasible" or instruction to branch.

choose initial convex relaxation: In the root node take the box $P \times Q$, else take the relaxation of the parent node;
for $k = 1, 2, \ldots$ **do**
  solve the convex relaxation;
  if *the relaxation is feasible* **then**
    for each pipe $a \in \mathcal{A}_{pipe}$ let $\left(p_{in}^k, p_{out}^k, q^k\right)_a$ be the solution of the relaxation;
    foreach $a \in \mathcal{A}_{pipe}$ **do**
      while $\left|p_a^u\left(p_{out}^k, q^k\right) - p_a^\ell\left(p_{out}^k, q^k, N_a\right)\right| > \delta_2$ **do**
        increase N_a;
    if *all* $\left(p_{in}^k, p_{out}^k, q^k\right)_a$ *are* δ_1*-feasible for* (17) **then**
      stop with solution $\left(p_{in}^k, p_{out}^k, q^k\right)_a$;
    choose "most infeasible" pipe $a \in \mathcal{A}_{pipe}$, i.e.,
      $a \in \arg\max_{a \in \mathcal{A}_{pipe}} \max\left\{p_{in}^k - p_a^u\left(p_{out}^k, q^k, N_a\right),\ p_a^\ell\left(p_{out}^k, q^k, N_a\right) - p_{in}^k\right\}$;
    if $\underline{q}_a < 0 < \overline{q}_a$ **then**
      fix orientation of flow on pipe a via branching;
    if $p_{in}^k > p_a^u\left(p_{out}^k, q^k, N_a\right)$ **then**
      try to cut off the solution with one inequality of the concave envelope; if this fails, then stop and perform branching w.r.t. p_{in}, p_{out} or q;
    else if $p_{in}^k < p_a^\ell\left(p_{out}^k, q^k, N_a\right)$ **then**
      add a gradient cut;
  else
    stop with "infeasible";

is satisfied for all produced pairs $\left(p_{out}^k, q^k\right)$. *Then Algorithm 4 terminates after a finite number of iterations.*

Proof. In order to keep the notation simple, we assume that there is only one pipe. One can straightforwardly extend this proof to an arbitrary number of pipes.

Suppose that the algorithm does not terminate, that is, it produces an infinite sequence of points which are feasible for the convex relaxation but not δ_2-feasible for (17). Since the algorithm does not terminate, the orientation of flow already has to be fixed. Hence, we can distinguish between input pressure p_{in} and output pressure p_{out}.

Let $\left(p_{in}^k, p_{out}^k, q^k\right)_{k \in \mathbb{N}}$ denote the sequence of solutions produced by the algorithm. We divide the iterations into two sets. With $\mathcal{O} \subseteq \mathbb{N}$ we denote the set of iterations with

$$p_{in}^k > p^u\left(p_{out}^k, q^k, N\right),$$

and with $\mathcal{L} = \mathbb{N} \setminus \mathcal{O}$ we denote the set of iterations with

$$p_{in}^k < p^\ell\left(p_{out}^k, q^k, N\right).$$

We will show that both sets have to be finite and, therefore, the algorithm terminates after a finite number of iterations.

We first consider the subsequence $\mathcal{O}$. Since the function $p^u\big(p_{out}, q\big)$ is convex, the concave envelope over the feasible set

$$\mathcal{F} := \big[\underline{p_{out}}, \overline{p_{out}}\big] \times \big[\underline{q}, \overline{q}\big] \cap \big\{(p_{out}, q) \,\big|\, 5c\, q \leq 4A\, p_{out}\big\}$$

consists of up to three linear inequalities. Thus, after at most three iterations $k \in \mathcal{O}$ the concave envelope is fully added to the convex relaxation. Any further point $\big(p_{in}^k, p_{out}^k, q^k\big)$ with $k \in \mathcal{O}$ cannot be separated from p^u with a linear inequality. Thus, the algorithm would terminate with the instruction to do branching. Therefore, $\mathcal{O}$ can have at most three elements (in general, three times the number of pipes).

Next, we show that the sequence $\mathcal{L}$ is finite, too. In every iteration $k \in \mathcal{L}$ an inequality of the form

$$p_{in} \geq p^u\big(p_{out}^k, q^k, N\big) + \nabla p^u\big(p_{out}^k, q^k, N\big)^\top \begin{pmatrix} p_{out} - p_{out}^k \\ q - q^k \end{pmatrix}$$

gets added to the relaxation. Since p^u is convex and continuously differentiable, it is Lipschitz continuous on the compact set $\mathcal{F}$. Hence, there is a radius r such that for all $k \in \mathcal{L}$ and all points $\big(p_{out}, q\big) \in B_r\big(p_{out}^k, q^k\big)$, where $B_r\big(p_{out}^k, q^k\big)$ is the ball around $\big(p_{out}^k, q^k\big)$ with radius r, the inequality

$$0 \leq p^u\big(p_{out}, q, N\big) - p^u\big(p_{out}^k, q^k, N\big) - \nabla p^u\big(p_{out}^k, q^k, N\big)^\top \begin{pmatrix} p_{out} - p_{out}^k \\ q - q^k \end{pmatrix} < \delta_1$$

is satisfied. That is, any point $\big(p_{in}^{\tilde{k}}, p_{out}^{\tilde{k}}, q^{\tilde{k}}\big)$ with $k < \tilde{k} \in \mathcal{L}$ and $\big(p_{out}^{\tilde{k}}, q^{\tilde{k}}\big) \in B_r\big(p_{out}^k, q^k\big)$ would be δ_1-feasible for (17) and the algorithm would terminate. So $\big(p_{out}^{\tilde{k}}, q^{\tilde{k}}\big) \in B_r\big(p_{out}^k, q^k\big)$ cannot be true. But as the feasible set $\mathcal{F}$ is compact, there exists a $K \in \mathcal{L}$ with

$$\bigcup_{\substack{k\in\mathcal{L},\\ k\leq K}} B_r\big(p_{out}^k, q^k\big) = \mathcal{F}.$$

After iteration K, every feasible point of the convex relaxation is δ_1-feasible for the inequality $p_{in} \geq p^u\big(p_{out}, q, N\big)$. Consequently, $\mathcal{L}$ does not contain any $k > K$ and is finite. □

This Lemma proves that Assumption 3 holds here and by Lemma 5, we derive the following Corollary.

Corollary 16. *Algorithm 4 terminates after a finite number of iterations.*

In order to make use of Algorithm 3 and Theorem 6, it remains to show that Condition (7) holds.

Proposition 17. *Suppose that Algorithm 3 applied to problem* (12) *produces an infinite nested sequence of nodes. Then the solutions of the convex relaxation produced by Algorithm 4 satisfy Condition* (7)*.*

Proof. Again, we only consider a single pipe $a = (u, v)$. Suppose that Algorithm 3 produces an infinite nested sequence of bounding boxes

$$\mathcal{F}_k = \big[\underline{p}_u^k, \overline{p}_u^k\big] \times \big[\underline{p}_v^k, \overline{p}_v^k\big] \times \big[\underline{q}_a^k, \overline{q}_a^k\big]$$

and let (p_u^k, p_v^k, q_a^k) be the last solution of the relaxation produced by Algorithm 4 for node k. Since our first priority is to fix the direction of the flow, we can assume that q_a is restricted to nonnegative values. Then p_u is the inflow-pressure and p_v is the outflow-pressure.

We have to prove that there is a $\tilde{k} \in \mathbb{N}$ such that the condition

$$\max\left\{\left(p_u^k - p_a^u(p_v^k, q_a^k, N)\right)_+,\ \left(p_a^\ell(p_v^k, q_a^k, N) - p_u^k\right)_+\right\} \leq \delta_1$$

holds for all $k \geq \tilde{k}$.

Note that the algorithm only returns a point (p_u^k, p_v^k, q_a^k) with $p_a^\ell(p_v^k, q_a^k, N) - p_u^k > \delta_1$ if there is another pipe, where infeasibility cannot be resolved by adding a cut. Otherwise, by construction of Algorithm 4 a gradient cut, which cuts off the current solution, would be added. Hence, it suffices to show that $\tilde{k} \in \mathbb{N}$ with $p_u^k - p_a^u(p_v^k, q_a^k, N) \leq \delta_1$ for all $k \geq \tilde{k}$ exists.

For simplicity, we assume that $5c\,\underline{q}_a^k < 4A\,\underline{p}_v^k$ and $5c\,\overline{q}_a^k < 4A\,\overline{p}_v^k$ is true for all nodes $\mathcal{F}_k$. Otherwise, we could adjust the variable bounds such that they do or, in the following, replace $\underline{p}_v^k$ by $\max\{\underline{p}_v^k, \frac{5c\,\underline{q}_a^k}{4A}\}$ and $\overline{q}_a^k$ by $\min\{\overline{q}_a^k, \frac{4A\,\overline{p}_v^k}{5c}\}$. Then due to monotonicity the inequality

$$p_a^u\big(\underline{p}_v^k, \underline{q}_a^k, N\big) \leq p_a^u\big(p_v, q_a, N\big) \leq p_a^u\big(\overline{p}_v^k, \overline{q}_a^k, N\big)$$

is fulfilled for all nodes k and all feasible points $(p_v, q_a) \in \big[\underline{p}_v^k, \overline{p}_v^k\big] \times \big[\underline{q}_a^k, \overline{q}_a^k\big]$. With the condition $\lim_{k\to\infty} \operatorname{diam} \mathcal{F}_k = 0$, the continuity of $p_a^u(p_v, q_a)$, and the fact that N is increased only finitely often, we can derive that $\tilde{k} \in \mathbb{N}$ exists with $p_a^u\big(\overline{p}_v^k, \overline{q}_a^k, N\big) - p_a^u\big(\underline{p}_v^k, \underline{q}_a^k, N\big) < \delta_1$ for $k \geq \tilde{k}$, i.e., for every solution (p_u^k, p_v^k, q_a^k) with $p_u^k - p_a^u(p_v^k, q_a^k, N) > \delta_1$ the inequality $p_u^k > p_a^u(\overline{p}_v^k, \overline{q}_a^k, N)$ holds. Since the constant function $p_a^u(\overline{p}_v^k, \overline{q}_a^k, N)$ is a concave overestimator, these points can be cut off with a linear inequality. Hence, the algorithm only produces δ_1-feasible solutions for all nodes $k \geq \tilde{k}$. □

With Lemma 15 and Proposition 17 we see that our construction of under- and overestimators for the Euler-equation satisfies the necessary requirements of Theorem 6.

Corollary 18. *Suppose that Conditions* (3) *and* (6) *hold. Then for* $\varepsilon > 0$ *and* $\delta_1, \delta_2 > 0$, *Algorithm 3 applied to problem* (12) *terminates after a finite number of iterations with an* $(\varepsilon, \delta_1 + \delta_2)$*-optimal solution of* (12) *or the conclusion that the problem is infeasible.*

This Corollary shows that our approach and Algorithm 3 works for the example of stationary gas transport.

5 Numerical Results

In this section, we demonstrate the behavior of the methods discussed in this paper on three examples of stationary gas networks. To this end, we implemented our approach using the LP-based branch-and-cut framework SCIP version 4.0, see [23, 38]. We keep the description of the implementation extremely short and refer to [30, 31] for a detailed discussion of modeling issues in stationary gas transport.

Recall from Section 4 that a gas network is given by the graph $\mathcal{G} = \big(\mathcal{V}, \mathcal{A}\big)$. The mass flows q_a, $a \in \mathcal{A}$, have to satisfy flow conservation (13). Moreover, flow and pressures have to satisfy given lower and upper bounds $(\underline{q}_a \leq q_a \leq \overline{q}_a,\ \underline{p}_v \leq p_v \leq \overline{p}_v)$. We consider several different network elements, which are handled as follows:

- Pipes are handled in the way described in Section 4. We determine the friction coefficient λ with the formula of Nikuradse [27, 28], i.e.,

$$\lambda = \Big(2\log_{10}\Big(\frac{D}{k}\Big) + 1.138\Big)^{-2},$$

  where k is the roughness of the pipe.
- Valves are modeled using binary variables. If the valve is open, the pressures on both sides are equal. If the valve is closed, the flow is set to 0 and the pressures are decoupled.
- Compressors allow to increase gas pressure. Compressor stations consist of several compressors that are connected by piping and valves. We use a rather simple model in which we approximate the operation states by a polyhedron, see Hiller et al. [14]. Two binary variables are used to decide whether the compressor is turned on/off or if its bypass is open/closed.
- A network may also contain control valves and resitors, which are modeled as described in [30, 31].

As objective function, we consider the following options:

- Minimize the number of compressors running. This can be seen as a proxy for the energy used. Note that we cannot express the consumed energy to run the compressors with the currently used compressor model.
- Maximize the sum of all pressures.
- Minimize the power loss. We minimize the function

$$\sum_{a=(u,v)\in\mathcal{A}} (p_u - p_v)\, q_a = \sum_{v\in\mathcal{V}} \Big(\sum_{a=(v,w)\in\mathcal{A}} q_a - \sum_{a=(u,v)\in\mathcal{A}} q_a \Big) p_v = \sum_{v\in\mathcal{V}} q_v^{\pm}\, p_v.$$

  This can be seen as a measure for the power loss in the network, since the change in pressure times the flow is proportional to the change in energy.

We implemented bound propagation based on the numerical methods (15a) and (15b). Since the input pressure is nondecreasing in output pressure and mass flow, we can derive an upper bound on the input pressure by computing $p^u_{in}(\overline{p_{out}}, \overline{q}_a)$. Similarly, $p^\ell_{in}(\underline{p_{out}}, \underline{q}_a)$ defines a lower bound on the input pressure, if $\underline{q}_a \geq 0$ holds. Note that we can also apply the methods in the direction of the flow and try to compute lower and upper bounds on the output pressure. Thereby, the bounds produced by the methods are reversed, i.e., the explicit midpoint method now produces upper bounds and the trapezoidal rule produces lower bounds. During this propagation step, we also check whether $p_i = p_i^\ell$ or $p_i = p_i^u$ violate the inequality $5c\,\overline{q}_a \leq 4A\,p_i$. In the worst case, when propagating the lower input pressure, $c\,\overline{q}_a > A\,p_i^\ell$ holds and p_N^u is a feasible output pressure. Thus, $p_{i+1}^\ell > p_i^\ell$ follows, although the pressure is nonincreasing! In this case, we cannot strengthen bounds and neither decide whether $(\underline{p_{in}}, \overline{q}_a)$ is at all feasible. This may happen in the beginning of the algorithm, since the bounds are large and the inequality $5c\,q \leq 4A\,p$ may have a nonempty intersection with the current feasible region; in the final solution, however, the slack of this inequality will be large.

All computations are performed with a precision of $\delta_1 = \varepsilon = 10^{-6}$ (SCIP default values) and $\delta_2 = 10^{-4}$, see Corollary 18.

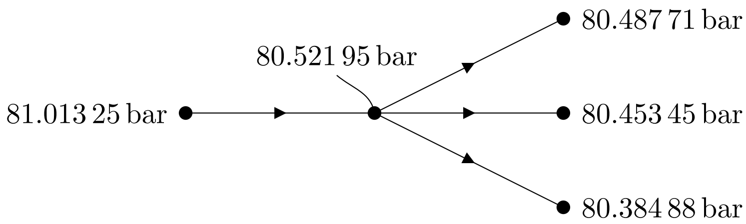

Figure 4: A small tree network with one entry on the left and three exits on the right side. The pressures shown are the solution to maximize the pressure with inflow of $150\,\mathrm{kg\,s^{-1}}$ and outflow of $50\,\mathrm{kg\,s^{-1}}$ at each exit.

5.1 A Tree Network

The first example is a simple tree which consists of five nodes and four pipes, see Figure 4. The pipes only differ in their length; their diameter is 1 m and the roughness is $k = 0.01\,\mathrm{mm}$. The pipe from the source to the innode is 15 km long and the pipes to the exits are, from top to bottom as seen in Figure 4, 10 km, 20 km and 40 km long. In this network, the flows on the pipes are already determined by the in- and outflow at the entry and the exits. Therefore, no branching on the flow occurs and the under- and overestimators can already be added in the root node of the branch-and-bound tree. As the network does not contain a compressor, we only tested maximizing all pressures and minimizing the power loss.

During presolving, some pressure bounds are improved, based on bound propagation. Then in both tests, one gradient cut and one overestimating linear inequality for each pipe get added to the LP relaxation. After that, the branch-and-bound process finishes with an optimal solution in the root node.

As indicated by the small amount of under- and overestimators added by the algorithm, we observe that for small mass flow rates the relation between the input pressure and the output pressure is almost linear. See also Table 1, which shows the pressure at the innode of the network for some pressures at the entry with fixed inflow of $150\,\mathrm{kg\,s^{-1}}$.

Table 1: Pressures at the entry and the innode of the tree network.

pressure at entry (bar)	50.00	55.00	60.00	65.00	70.00	75.00	80.00
pressure at innode (bar)	49.20	54.27	59.33	64.39	69.43	74.47	79.50

5.2 Diamond Graph

In the next example, the graph has a diamond shape, see Figure 5. It consists of one entry s, one exit e, four innodes $n_1, \ldots, n_4$ and seven pipes. Here, the pipes vary in diameter and length. The diameter is either 1 m or 1.30 m and the length varies between 14 km and 40 km. The roughness is again $k = 0.01\,\mathrm{mm}$.

During presolving, the flows q_2 and q_5 are replaced by $q_1 - q_3$ and $q_7 - q_6$, which can be done due to the flow conservation in the nodes n_1 and n_3 and because the flows q_1 and q_7 are already determined by the in- and outflow at s and e. Furthermore, the pressure bounds for s, e, n_1 and n_3 are improved during presolving. In contrast, the pressure bounds on n_2 and n_4 are not improved, because the bounds of $q_2, \ldots, q_6$ are too large, and positive as

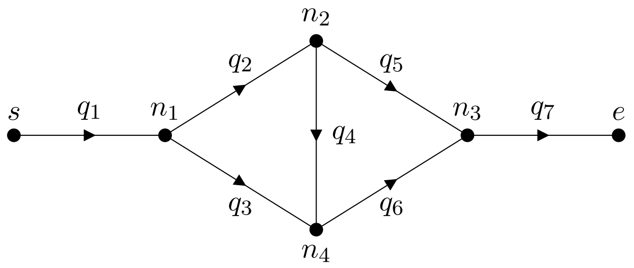

Figure 5: A small network with one entry (s) on the left, one exit (e) on the right and two overlapping circles.

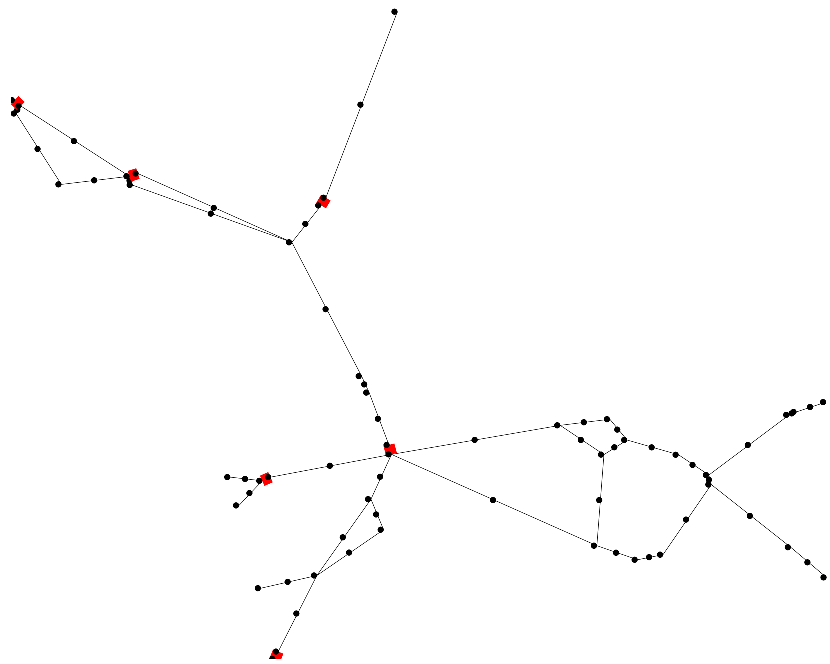

Figure 6: Picture of the GasLib-40, rotated counter clockwise by 90 degrees.

well as negative flow is possible in the beginning. Nevertheless, bound propagation is more important for this instance. When maximizing pressures, a pressure bound was improved 307 times, and pressure bounds were improved 467 times when minimizing power loss.

The branch-and-bound tree contains 16 nodes with 7 leaves when maximizing the pressure and 21 nodes with 8 leaves while minimizing the power loss. In the first case, branching on the flow variables only occurred for fixing the flow direction on the pipes from 2 to 6. Additionally, it took one branching step on a pressure variable. The ODEs on the pipes are approximated by 55 gradient cuts, varying from 0 on pipe 1 to 18 on pipe 2, and only 14 overestimators, varying from 0 to 4 on each pipe. In the second case, except of fixing the flow direction, there was one branching step on a flow variable and one branching step on two pressure variables each. Here, 65 gradient cuts were applied, varying from 5 to 16 per pipe, and 25 overestimating cuts with 0 to 10 per pipe.

Table 2: Statistics on the GasLib-40 instance for the three different objective functions.

objective	max $\sum_{v\in\mathcal{V}} p_v$	min $\sum_{v\in\mathcal{V}} q_v^{\pm} p_v$	min $\sum_{a\in\mathcal{A}_{cs}} z_a$
solving time (seconds)	877.74	542.48	65.08
processed Nodes	12 444	2790	263
branchings on flow (%)	46.2	81.0	84.2
branchings on pressure (%)	52.9	0.0	8.8
branchings on binary variables (%)	0.9	19.0	7.0
leaves	5900	1412	73
cut offs by propagation	5297	1149	38
bound changes by propagation	32 444	63 317	6051
added overestimators	7362	2385	985
added underestimators	6360	4281	488

5.3 GasLib-40

The final example is the GasLib-40 instance [17], see Figure 6. This instance consists of 40 nodes, 39 pipes and 6 compressors. Thus, it contains 12 binary variables that determine whether a compressor is turned on or is in bypass. For this instance the roughness varies from 0.012 mm to 0.05 mm, the length varies from 3 km to 86 km and the diameter varies between 40 cm and 1 m.

Table 2 shows some statistics on the work of Algorithm 3 for the three different objective functions. As one can see, the branch-and-bound trees are of very different size. When maximizing pressures, more than half of the branching steps were on the pressure variables, while otherwise branching on pressure variables only played a minor role. For both other objectives, the algorithm mainly branches on flow variables. Overall, branching on binary variables amounts only to a small part of the branching steps, because branching on them is preferred over continuous variables. Therefore, branching on the binary variables happens early in the tree and will not be repeated in the deeper regions.

For this instance, bound propagation plays a much more important role, because in some cases a node can be cut off due to bound propagation, e.g., when propagating an upper pressure bound yields a value lower than the lower bound. Up to almost 90% of the leaves are cut off by bound propagation. Furthermore, there were up to 63 317 bound changes due to bound propagation, when minimizing the power loss.

The diamond graph and the GasLib-40 with minimization of the power loss suggest that adding underestimators is more important than adding overestimators. This is contradicted by the GasLib-40 with the other objectives. So it is not clear, yet, what has more impact on the performance.

Table 3 shows statistics on the final discretizations, used to compute (15a) and (15b). The discretizations are initialized with a minimum number of 179 grid points and maximal 7961 grid points on a single pipe such that the requirements of Lemma 14 are satisfied. The number of grid points corresponds to step sizes between 5.12 m and 19.2 m. Furthermore, when minimizing the power loss, 363 507 additional variables would have been needed in a first discretize then optimize approach with the same precision.

Table 3: Statistics on the final discretizations of the pipes.

objective	max $\sum_{v\in\mathcal{V}} p_v$	min $\sum_{v\in\mathcal{V}} q_v^{\pm} p_v$	min $\sum_{a\in\mathcal{A}_{cs}} z_a$
minimal number of grid points	179	371	179
maximal number of grid points	43 889	63 681	43 889
smallest step size	0.32 m	0.32 m	0.32 m
largest step size	19.2 m	15.18 m	19.2 m
overall number of grid points	190 350	363 547	264 929

6 Outlook

In this paper, we have investigated an adaptive method to construct relaxations for mixed-integer optimization problems with ODE constraints and its integration in a branch-and-bound framework. The method relies on the fact that the values of the ODE solution are only required at a priori fixed positions. If the ODE has favorable properties – like in water or gas transport – we have derived an effective way to produce lower and upper bounds based on discretization methods, leading to finite convergence.

For gas transport, the interplay of branching on integer variables and spatial branching needs further investigation. Moreover, the question is whether non-horizontal pipes can also be treated. Finally, the implementation can be extended by considering more elaborate compressor models and additional presolving techniques. The performance could be improved by considering additional branching rules and primal heuristics.

It would be interesting to extend the approach to instationary network problems, which leads to mixed-integer PDE-constrained optimal control problems.

Acknowledgement

This work was supported by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) Research Grant CRC/Transregio 154 within Project A01. We thank Benjamin Hiller and Tom Walther for computing the polyhedral approximations of compressor stations.

References

[1] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas. A global optimization method, αBB, for general twice differentiable NLPs – II. Implementation and computational results. *Computers and Chemical Engineering*, 22:1159–1179, 1998.

[2] C. S. Adjiman, S. Dallwig, C. A. Floudas, and A. Neumaier. A global optimization method, αBB, for general twice differentiable NLPs – I. Theoretical advances. *Computers and Chemical Engineering*, 22:1137–1158, 1998.

[3] H. G. Bock, C. Kirches, A. Meyer, and A. Potschka. Numerical solution of optimal control problems with implicit switches. Technical report, Optimization Online, 2016.

[4] C. Buchheim, R. Kuhlmann, and C. Meyer. Combinatorial optimal control of semilinear elliptic PDEs. Technical report, Optimization Online, 2015.

[5] B. Chachuat, A. B. Singer, and P. I. Barton. Global mixed-integer dynamic optimization. *AIChE Journal*, 51(8):2235–2253, 2005.

[6] H. Diedam and S. Sager. Global optimal control with the direct multiple shooting method. *Optimal Control Applications and Methods*, pages 1–22, 2017.

[7] C. A. Floudas and C. E. Gounaris. A review of recent advances in global optimization. *Journal of Global Optimization*, 45(1):3–38, 2008.

[8] A. Fügenschuh and I. Vierhaus. A global approach to the optimal control of system dynamics models. In *Proceedings of the 31st International Conference of the System Dynamics Society*, 2013.

[9] M. Gugat, G. Leugering, A. Martin, M. Schmidt, M. Sirvent, and D. Wintergerst. Towards simulation based mixed-integer optimization with differential equations. Technical report, Optimization Online, 2016.

[10] M. Gugat, G. Leugering, A. Martin, M. Schmidt, M. Sirvent, and D. Wintergerst. Mip-based instantaneous control of mixed-integer pde-constrained gas transport problems. Technical report, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2017. Available at `https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/docId/140`.

[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. Technical report, TRR 154, 2016. available at `https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/docId/121`.

[12] F. M. Hante and S. Sager. Relaxation methods for mixed-integer optimal control of partial differential equations. *Computational Optimization and Applications*, 55(1):197–225, May 2013.

[13] R. Hemmecke, M. Köppe, J. Lee, and R. Weismantel. Nonlinear integer programming. In M. Jünger, T. M. Liebling, D. Naddef, G. L. Nemhauser, W. R. Pulleyblank, G. Reinelt, G. Rinaldi, and L. A. Wolsey, editors, *50 Years of Integer Programming 1958–2008 – From the Early Years to the State-of-the-Art*, pages 561–618. Springer, 2010.

[14] B. Hiller, R. Saitenmacher, and T. Walther. Analysis of operating modes of complex compressor stations. In *Proceedings of Operations Research 2016*, 2017. accepted for publication.

[15] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE constraints*, volume 23 of *Mathematical Modelling: Theory and Applications*. Springer, New York, 2009.

[16] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. Springer, Berlin, 3. rev and enlarged ed. edition, 1996.

[17] J. Humpola, I. Joormann, D. Oucherif, M. E. Pfetsch, L. Schewe, M. Schmidt, and R. Schwarz. GasLib – A Library of Gas Network Instances. Technical report, Optimization Online, 2015. Available at `http://www.optimization-online.org/DB_HTML/2015/11/5216.html`.

[18] M. N. Jung, G. Reinelt, and S. Sager. The Lagrangian relaxation for the combinatorial integral approximation problem. *Optimization Methods and Software*, 30(1):54–80, 2015.

[19] J. Lee and S. Leyffer, editors. *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*. Springer-Verlag, New York, 2012.

[20] Y. Lin, J. A. Enszer, and M. A. Stadtherr. Enclosing all solutions of two-point boundary value problems for ODEs. *Computers & Chemical Engineering*, 32(8):1714–1725, 2008.

[21] Y. Lin and M. A. Stadtherr. Validated solutions of initial value problems for parametric ODEs. *Applied Numerical Mathematics*, 57(10):1145–1162, 2007.

[22] M. Locatelli and F. Schoen. *Global Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2013.

[23] S. J. Maher, T. Fischer, T. Gally, G. Gamrath, A. Gleixner, R. L. Gottwald, G. Hendel, T. Koch, M. E. Lübbecke, M. Miltenberger, B. Müller, M. E. Pfetsch, C. Puchert, D. Rehfeldt, S. Schenker, R. Schwarz, F. Serrano, Y. Shinano, D. Weninger, J. T. Witt, and J. Witzig. The SCIP Optimization Suite 4.0. Technical report, Optimization Online, 2017. Available at `http://www.optimization-online.org/DB_HTML/2017/03/5895.html`.

[24] G. P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I — convex underestimating problems. *Mathematical Programming*, 10(1):147–175, 1976.

[25] N. Nedialkov, K. Jackson, and G. Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105(1):21–68, 1999.

[26] M. Neher, K. R. Jackson, and N. S. Nedialkov. On taylor model based integration of ODEs. *SIAM Journal on Numerical Analysis*, 45(1):236–262, 2007.

[27] J. Nikuradse. *Strömungsgesetze in rauhen Rohren*. Forschungsheft auf dem Gebiete des Ingenieurwesens. VDI-Verlag, Düsseldorf, 1933.

[28] J. Nikuradse. *Laws of Flow in Rough Pipes*, volume Technical Memorandum 1292. National Advisory Committee for Aeronautics Washington, 1950.

[29] I. Papamichail and C. S. Adjiman. A rigorous global optimization algorithm for problems with ordinary differential equations. *Journal of Global Optimization*, 24(1):1–33, Sep 2002.

[30] M. E. Pfetsch, A. Fügenschuh, B. Geißler, N. Geißler, R. Gollmer, B. Hiller, J. Humpola, T. Koch, T. Lehmann, A. Martin, A. Morsi, J. Rövekamp, L. Schewe, M. Schmidt, R. Schultz, R. Schwarz, J. Schweiger, C. Stangl, M. C. Steinbach, S. Vigerske, and B. M. Willert. Validation of nominations in gas network optimization: models, methods, and solutions. *Optimization Methods and Software*, 30(1):15–53, 2015.

[31] M. E. Pfetsch, T. Koch, L. Schewe, and B. Hiller, editors. *Evaluating Gas Network Capacities*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2015.

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

[33] S. Sager, H. G. Bock, and G. Reinelt. Direct methods with maximal lower bound for mixed-integer optimal control problems. *Mathematical Programming*, 118(1):109–149, 2009.

[34] S. Sager, M. Jung, and C. Kirches. Combinatorial integral approximation. *Mathematical Methods of Operations Research*, 73(3):363–380, 2011.

[35] A. Sahlodin and B. Chachuat. Convex/concave relaxations of parametric ODEs using taylor models. *Computers & Chemical Engineering*, 35(5):844–857, 2011. Selected Papers from ESCAPE-20 (European Symposium of Computer Aided Process Engineering - 20), 6-9 June 2010, Ischia, Italy.

[36] A. M. Sahlodin and B. Chachuat. Discretize-then-relax approach for convex/concave relaxations of the solutions of parametric ODEs. *Applied Numerical Mathematics*, 61(7):803–820, 2011.

[37] M. Schmidt, M. Sirvent, and W. Wollner. A decomposition method for MINLPs with Lipschitz continuous nonlinearities. Technical report, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2017. Available at `https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/docId/145`.

[38] SCIP. Solving Constraint Integer Programs. `http://scip.zib.de/`.

[39] J. K. Scott and P. I. Barton. Convex and concave relaxations for the parametric solutions of semi-explicit index-one differential-algebraic equations. *Journal of Optimization Theory and Applications*, 156(3):617–649, Mar 2013.

[40] J. K. Scott and P. I. Barton. Improved relaxations for the parametric solutions of ODEs using differential inequalities. *Journal of Global Optimization*, 57(1):143–176, Sep 2013.

[41] J. K. Scott, B. Chachuat, and P. I. Barton. Nonlinear convex and concave relaxations for the solutions of parametric ODEs. *Optimal Control Applications and Methods*, 34(2):145–163, 2013.

[42] A. B. Singer and P. I. Barton. Bounding the solutions of parameter dependent nonlinear ordinary differential equations. *SIAM Journal on Scientific Computing*, 27(6):2167–2182, 2006.

[43] A. B. Singer and P. I. Barton. Global optimization with nonlinear ordinary differential equations. *Journal of Global Optimization*, 34(2):159–190, Feb 2006.

[44] I. Vierhaus and R. Gottwald. SD-SCIP – system dynamics scip: A SCIP plug-in for solving system dynamics optimization problems. `http://sdscip.zib.de/`, 2017.

[45] M. E. Villanueva, B. Houska, and B. Chachuat. Unified framework for the propagation of continuous-time enclosures for parametric nonlinear ODEs. *Journal of Global Optimization*, 62(3):575–613, Jul 2015.