

BENJAMIN HILLER, THORSTEN KOCH, LARS SCHEWE,
ROBERT SCHWARZ, JONAS SCHWEIGER

A System to Evaluate Gas Network Capacities: Concepts and Implementation

Zuse Institute Berlin
Takustr. 7
14195 Berlin
Germany

Telephone: +49 30-84185-0
Telefax: +49 30-84185-125

E-mail: bibliothek@zib.de
URL: <http://www.zib.de>

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

A System to Evaluate Gas Network Capacities: Concepts and Implementation

Benjamin Hiller* Thorsten Koch* Lars Schewe†
Robert Schwarz* Jonas Schweiger*

Abstract

Since 2005, the gas market in the European Union is liberalized and the trading of natural gas is decoupled from its transport. The transport is done by so-called *transmissions system operators* or TSOs. The market model established by the European Union views the gas transmission network as a black box, providing shippers (gas traders and consumers) the opportunity to transport gas from any entry to any exit. TSOs are required to offer maximum independent capacities at each entry and exit such that the resulting gas flows can be realized by the network without compromising security of supply. Therefore, evaluating the available transport capacities is extremely important to the TSOs.

This paper gives an overview of the toolset for evaluating gas network capacities that has been developed within the ForNe project, a joint research project of seven research partners initiated by Open Grid Europe, Germany's biggest TSO. While most of the relevant mathematics is described in the book [KHPS15], this article sketches the system as a whole, describes some developments that have taken place recently, and gives some details about the current implementation.

1 Introduction

About 20% of the whole energy demand of Germany (and Europe) is met by natural gas. Until 2005, gas transport and supply in Europe was provided by a handful companies, owning and operating the network to do so. To establish a European gas market, the European Union in 2005 legislated that gas trading and transport had to be done by mutual completely independent companies to ensure discrimination free access to the transport network for all traders [GGH⁺15]. This changed the operation and business model of the gas transport operators who, due to their high investment cost for natural gas pipeline networks, are mostly a natural monopoly. Before, they were part of an integrated organization and could plan the network operation and expansion together with the traders. Since then, they are independent and need to plan under the uncertainty about the gas flow situations resulting from trading. They have to make sure that their network works under these new circumstances. The

*Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany, {hiller, koch, schwarz, schweiger}@zib.de

†Universität Erlangen-Nürnberg, Cauerstrasse 11, 91058 Erlangen, Germany, lars.schewe@math.uni-erlangen.de

goal of this article is to show how mathematical optimization can provide tools to deal with these tasks.

In a simplified view, the EU regulations stipulate that the so-called *entry/exit model* with a virtual trading point is the basis of the capacity market. The idea is that for each network a virtual trading point is introduced. All market participants need to buy capacity from the TSO to access this virtual trading point. Suppliers need to buy entry capacity, demand customers need to buy exit capacity. The preferred capacity product sold by the TSO is known as freely allocable capacity, i.e., the capacity bought by the customer (the *booking*) gives an upper bound on the amount of gas that can be bought resp. sold in day-ahead trading. After trading, the amount of capacity used is communicated to the TSO (the *nomination*). The TSO may assume that the result of market operation is balanced in the sense that the amount nominated at entries is in sum equal to the amount nominated at exits. The requirement for the TSO is now that it is able to transport this gas. In general, this would be impossible for the TSO. To prevent such a situation, the TSO is allowed to restrict the amount of bookable capacity at each point in the network. A TSO may only sell capacity rights for which it can guarantee that each “likely and realistic” [Gas10, §9] gas flow complying with the capacity rights booked by all transport customers can technically be realized. Thus a TSO needs a way to check this requirement. We call this task the *verification of booked capacities*. Our goal for this article is to discuss a system which supports the TSO in checking whether it can sell a given set of booked capacities without risking the security of supply.

The results of this article were achieved as part of the *Forschungskooperation Netzoptimierung* (ForNe) between Germany’s biggest TSO, Open Grid Europe GmbH (OGE, then E.ON Gas Transport) and Zuse Institute Berlin, Technische Universität Darmstadt, Friedrich-Alexander Universität Erlangen-Nürnberg, Universität Duisburg-Essen, Leibniz-Universität Hannover, Humboldt-Universität zu Berlin, Weierstrass-Institut für Angewandte Analysis und Stochastik, and later on the companies develOPT, and atesio. It was mainly funded by OGE and supported by the German Federal Ministry of Economic Affairs and Energy and the Federal Ministry of Education and Research. The principal outcome for OGE is a set of ForNe tools for the verification of booked capacities (about 300 000 lines of code). From the scientific point, the most visible outcome is the book *Evaluating gas network capacities* [KHPS15] which describes most of the research that was done. This article gives an update on this book and outlines some of the more technical details of the toolchain that were not discussed in the book.

The question tackled in this article concerns all TSOs in Europe. These use different techniques to this end, which are only partially described in the literature. From discussions, we know that most TSOs use a similar distinction between generating “challenging” gas flow situations and checking these gas flow situations as we do. Methods to generate “challenging” gas flow situations are described in, e.g., [BEG⁺15, SHDA15, HS16]. Checking for technical feasibility is either done using simulation tools and expert knowledge (augmented by automated tools) or by similar optimization-based methods, e.g., [vdH04]. The physical models underlying these methods are either very detailed, if using simulation tools, or suitably simplified. Our goal here was to provide a unified methodology which allows us to incorporate more and more market information to also deal with future capacity contracts and market-based mechanisms. We

show the flexibility of our approach by showing how another contract type, flow reallocation options, can be incorporated in our framework.

Scope and outline of our approach

The technical and contractual constraints that have to be taken into account for verifying booked capacities are very complex. Apart from the entities of interest, the transmission capacities, many more details need to be considered. Our approach to this question takes into account the following aspects:

capacity contracts Capacity contracts define limits on the amounts of gas that are injected or withdrawn, maybe subject to additional constraints or conditions.

historical gas flows Gas flow measurements may be used for predicting future gas flows where appropriate.

network topology and devices A gas network is modeled as graph with arcs of different types representing pipes, valves, control valves and compressor units.

operational constraints The operation of a network is subject to a range of constraints like bounds for the pressure, joint operation of several network elements, and complex flow bounds for certain paths in the network.

different gas qualities Gas from different sources differs in its composition, resulting e.g., in different calorific values. Since the capacity rights and the gas market are defined in terms of thermal power, the mixing of different calorific values needs to be taken into account to ensure proper power supply at the exits.

flow reallocation options A TSO may have contracted options to adjust the flow at certain nodes in order to support operating the network reliably.

flow rebalancing As we deal with stationary models, adjusting the flow at one node using a flow reallocation option requires an adjustment at other nodes to maintain the balance of injected and withdrawn gas.

Mathematically, there are two major challenges in this problem. First, one has to deal with the *uncertainty* of how transport customers use their capacity contracts, i.e. uncertainty about which gas flow situations have to be served by the network. Second, deciding *feasibility* of a given gas flow situation is, with the detail level asked by OGE, a very difficult mixed-integer nonconvex nonlinear decision problem. Similarly to what is done in industry, we address these two challenges separately.

The uncertainty of how the network is going to be used is handled by considering (a large set of) *nomination scenarios* that describe particular gas flow situations along with technical constraints, flow reallocation options, and rebalancing rules. Each of these nomination scenarios is then separately checked for *technical feasibility*: Is it possible to operate the network within the technical limits, possibly using the flow reallocation options? The problem of answering this question is called *validation of nominations*.

The basic ideas for this approach are described in detail in [KHPS15]. In addition to the problem aspects covered there, the models and methods have been extended to cover the following.

- Dealing with different types of gas with different calorific values [GMSS15]. This is important as the capacity contracts are in terms of thermal power of the transported gas, whereas the technical operation is described in terms of mass flows. To couple these two views, it is necessary to track the mixing of calorific values according to gas flow, which depends on the operation of the network.
- Support for flow reallocation options and flow rebalancing. This is important as flow reallocation options are a flexible mechanism that may be used to offer more transmission capacity without the need for network expansion. The modeling of flow reallocation options and flow rebalancing presented in Section 2.2.3 has not been published before.

The outline of the remaining paper is as follows. Section 2 presents the mathematical models and methods from a high-level perspective, focusing on the interplay of the submodels for different aspects of the problem. In Section 3, we describe the architecture of the software and spotlight some parts of the implementation and the experiences we have made so far. We conclude and provide an outlook in Section 4.

2 Mathematical models and methods

This section summarizes the mathematical concepts proposed in [KHPS15] and outlines the additional ideas developed for the extensions mentioned above. The focus is on providing a basic understanding of all aspects and the description of the interplay of all model components.

The topology of a gas network is modeled by a directed graph $G = (V, A)$. The set of entries is denoted by V_+ , the set of exits by V_- , and their union by V_\pm . The arcs are often referred to as *network elements*.

The mathematical models we developed address two distinct aspects: The *usage of the network* by shippers for their gas trading and the *operation of the network* by the TSO. In principle, network usage may be modeled by a set of gas flow situations. We model a particular gas flow situation as a balanced vector of inflows and outflows $P^{\text{ext}} = (P_u^{\text{ext}})_{u \in V_\pm}$ and call such a vector a *nomination* [KHPS15]. However, considering fixed inflows is not sufficient in view of flow reallocation options and the resulting necessity to model flow rebalancing according to the market. We thus consider *nomination scenarios*, consisting of a nomination and a *market preference vector* $(w_u)_{u \in V_\pm}$ that describes the relative attractivity of entries and exits by weights. The nomination and the market preference vector are the variables of the models and methods used to derive a set of nomination scenarios that capture potential gas flow situations.

For the models and methods for the operation of the network that are used to validate a nomination scenario, the nomination and the market preference vectors are, of course, the parameters. At the exits, the goal is to obtain exactly the outflows specified in the nomination. For the entries, however, flow reallocation options may be used to change the inflows specified by the nomination

network usage variables These variables model the gas flow situation desired by gas transport customers.

$P_u^{\text{ext}}, u \in V_{\pm}$ Desired inflow/outflow at each entry/exit in terms of thermal power.

$w_u, u \in V_{\pm}$ Weight of each entry/exit, describing the attractivity of injecting/withdrawing gas at this particular node. We call the vector $(w_u)_{u \in V_{\pm}}$ the *market preference vector*.

network operation variables These variables model how the network is actually operated to serve the desired gas flow situation. This may include modifying the gas flow situation by using reallocation options.

$P_u^{\text{eff}} \in \mathbb{R}_{\geq 0}, u \in V_+$ Effective inflow at each entry in terms of thermal power after employing reallocation options and flow balancing.

$P_u \in \mathbb{R}_{\geq 0}, u \in V_-$ Outflow at each exit in terms of thermal power according to the flow and calorific value arriving at this exit. The goal is to operate the network such that we have $P_u = P_u^{\text{ext}}$ for all $u \in V_-$.

$q_a \in \mathbb{R}, a \in A$ Mass flow for each arc/network element. A nonnegative value indicates flow in the direction of the arc, a negative value flow in the opposite direction.

$H_{c,u} \in \mathbb{R}_{\geq 0}, u \in V$ Calorific value at each network node.

$p_u \in \mathbb{R}_{\geq 0}, u \in V$ Pressure at each network node.

$s_a \in \{0, 1\}^{k_a}, a \in A_{\text{active}}$ Vector of switching decisions for each active element. The number of switching variables k_a depends on the type of the network element.

Figure 1: Overview on the variables used to model network usage and network operation.

to aid network operation. In this case, flow rebalancing adjusts the flows at the remaining entries such that the overall inflow/outflow balance is maintained and entries with a higher weight are used up to their booked capacities as much as possible. Moreover, there are several other quantities that need to be considered to model the technical and physical constraints of network operation.

Section 2.1 describes how we obtain suitable nomination scenarios and how they are used to verify booked capacities. An overview of the models used to validate a nomination scenario is given in Section 2.2. Fig. 1 shows the variables used to model network usage and network operation, respectively.

2.1 Models and methods for verifying booked capacities

As summarized in the introduction, the entry/exit model enables the gas shippers to easily deal with the task of transporting gas. All they need to do is to book sufficient entry or exit capacities (depending on their role) some time in advance and to sell or buy gas at the virtual trading point. Since the gas is sold to/bought from a market participant in the same network, this transaction is

automatically balanced. From the market perspective, everything is fine.

The TSO, however, needs to provision the network and/or limit the bookable capacities such that the desired gas transport can be realized. To do this, it is necessary to understand and model how the capacities may be used by the shippers and which gas flow situations may arise. The gas withdrawn from most of the exits is consumed and thus follows certain demand patterns. Due to this, the offtake at these exits is quite predictable. The remaining exits are e.g., storages and interconnection points to adjacent networks. As for entries, their flows depend very much on the gas market and are thus hard to predict. Moreover, the flows at these entries and exits may be highly correlated in a way that adversely affects network operation. For instance, it may happen that very few entries supply a large share of the exits and that the gas needs to be transported over long distances, requiring much compression power.

To handle these two types of behavior, our approach [HHH⁺15] combines a stochastic model for the predictable exit flows with an adversarial model for the entry flows and the remaining exit flows. The stochastic model is estimated from historical flow data as outlined in Section 2.1.1. The adversarial model is based on the *market preference vector* $(w_u)_{u \in V_{\pm}}$ that provides a weight for each entry and exit. This weight indicates the relative attractivity of injecting or withdrawing gas at a particular node. These two models are combined to obtain meaningful nomination scenarios that can be used for verifying booked capacities. How this is done is described in more detail in Section 2.1.2.

2.1.1 Estimating distributional models for gas demand

Transmission system operators measure the amount of gas injected to or withdrawn from the network for every entry and exit, respectively. They are legally obliged [Gas10] to use this data in their capacity models to consider “likely and realistic” flow situations for their network. This section summarizes the approach for obtaining distributional models from historical data about gas flows, which is described in full detail in [HHL⁺15a].

The demand of exits usually follows certain patterns and it is thus reasonable to describe it using stochastic models that are then used for forecasting. The exit flows often depend of the *local temperature* in the particular geographical area (see Fig. 2 for examples). To avoid modeling the temperature distributions within Germany, we chose a pragmatic approach and introduced a network-wide *reference temperature*. This reference temperature is determined, for each day, as a weighted average of the local temperatures of several locations in Germany. To further simplify the modeling of the temperature dependency of gas demand, we divide the temperature range in *temperature classes*. Each temperature class should be small enough to treat the temperature as constant within this temperature class. Due to lack of data, this requirement is not fulfilled for very low temperatures, which are thus handled differently.

For each temperature class with sufficiently many measurements, each of the following univariate distributions is fitted to the measurement data of each exit:

- univariate normal distribution,
- univariate log-normal distribution,
- univariate uniform distribution,

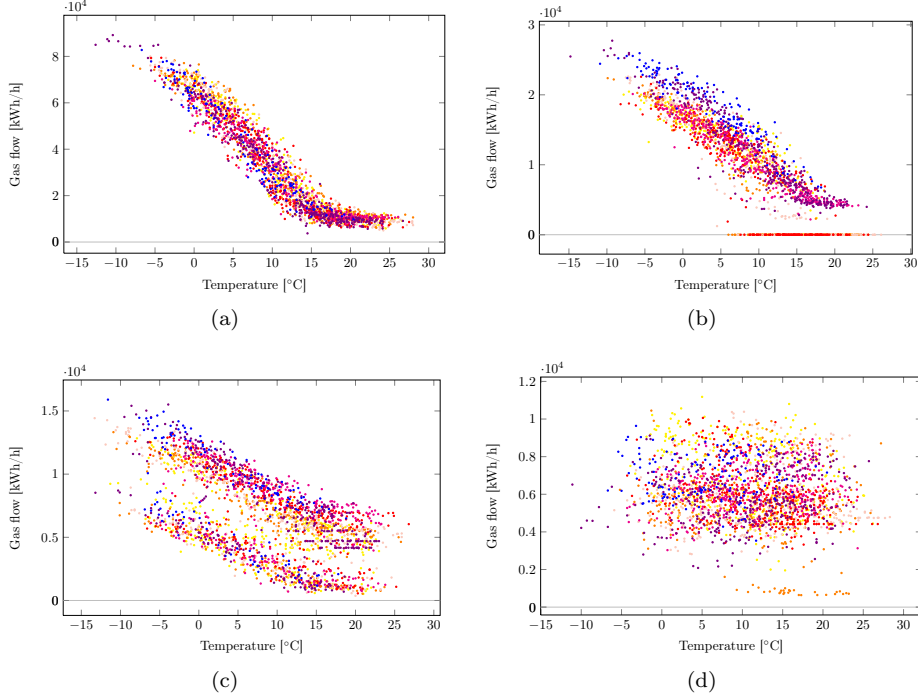


Figure 2: Plots showing the daily gas demand for four selected exits. The indicated temperature is the local temperature at the exit location.

- univariate Dirac distribution.

The fit is done by determining the parameters of each distributions type such that the maximum Kolmogorov distance between the distribution and the empirical distribution is minimized. We select the distribution attaining the overall minimum Kolmogorov distance for modeling the demand of the considered exit.

For the “cold” temperature classes with insufficient data, we employ the following mechanism. The available measurement data of an exit is used to fit a spline function to the means of the data of small temperature intervalls. This spline function is then used to extrapolate estimates for the mean values of the “cold” temperature classes. The distribution type and the remaining parameters are copied from the “coldest” temperature class with sufficient data.

To account for correlations between the gas flows it is desirable to use multivariate stochastic models. We employ so-called *multivariate normal-like distributions* for this purpose. These are generalizations of multivariate normal distributions that admit normal or log-normal distributions as marginal distributions. The corresponding multivariate stochastic models are obtained by complementing the univariate marginal distributions of the nodes with normal or log-normal distributions with the correlation matrix computed from the data.

2.1.2 Verifying booked capacities using nomination scenarios

As discussed in [HHH⁺15], we are actually interested in estimating the probability that a gas flow situation matching the capacity contracts of a booking is

technically feasible within the gas network. To formalize this, let $\xi: \Omega \rightarrow \mathbb{R}^{V_{\text{stat}}}$ be a random demand vector defined on a suitable probability space $(\Omega, \mathcal{A}, \mathbb{P})$, where V_{stat} is the set of exits for which a stochastic model is available. Moreover, denote by κ_u the capacity booked at node u . Finally, let $\mathcal{B}_=(\xi)$ be the set of all vectors $P^{\text{ext}} = (P_u^{\text{ext}})_{u \in V}$ that

- are *balanced* w.r.t. in- and outflow:

$$\sum_{u \in V_+} P_u^{\text{ext}} = \sum_{u \in V_-} P_u^{\text{ext}}, \quad (1a)$$

- respect the capacity contracts for each node:

$$P_u^{\text{ext}} \leq \kappa_u \quad (1b)$$

- for $u \in V_{\text{stat}}$ coincide with the sampled values:

$$P_u^{\text{ext}} = \xi_u. \quad (1c)$$

We are then interested in estimating the probability

$$\beta := \mathbb{P}(P^{\text{ext}} \in \mathcal{B}_=(\xi) \mid P^{\text{ext}} \text{ is technically feasible}). \quad (2)$$

This expresses the idea that a given (stochastic) exit demand can be met by many supply patterns and that it should be very likely that *any* supply pattern results in technically feasible operation.

Our method [HHH⁺15] estimates β using suitable nomination scenarios as follows. Since validating a nomination scenario is computationally expensive, the goal is to construct a small yet representative set of nomination scenarios to validate.

1. Sample a (large) set of demand vectors according to ξ . In order to obtain fast convergence even for relatively few samples we use a randomized quasi-Monte Carlo method. As a result, we obtain a set $\{s_1, \dots, s_M\}$ of sampled demand vectors.
2. Use scenario reduction techniques to obtain a smaller, yet representative set of demand vectors. The result is a subset $\{s_1, \dots, s_N\}$, $N < M$, of the original sampled demand vectors, with potentially nonuniform probabilities $\{p_1, \dots, p_N\}$.
3. For each demand vector s , obtain a set of market preference vectors that describe potential supply patterns to satisfy this demand. In our method, we draw these market preference vectors uniformly at random from the unit sphere $\mathcal{S}^{V_{\pm}}$ to model a diverse set of potential market behaviors (supply patterns). For each of these market preference vectors, solve the optimization problem

$$\min \left\{ \sum_{u \in V_{\pm}} w_u P_u^{\text{ext}} \mid P_u^{\text{ext}} \text{ satisfies (1)} \right\} \quad (3)$$

to obtain a balanced vector P^{ext} of desired inflows and outflows.

Perform nomination validation for each nomination scenario (P^{ext}, w) .

4. Estimate the probability β by

$$\sum_{\{s_i | \text{all nomination scenarios for } s_i \text{ are feasible}\}} p_i.$$

The probability estimated by the method may be used by network planners to assess the risk of a particular set of capacity contracts.

2.2 Models and methods for nomination validation

The task in the nomination validation problem is to determine whether there is a feasible control for the gas network serving a nomination scenario given by the vector $(P_u^{\text{ext}})_{u \in V_{\pm}}$ of desired in- and outflows, the market preference vector $(w_u)_{u \in V_{\pm}}$, and the set of applicable flow reallocation options.

To solve this very complex problem, we address each of the following three problem aspects by a separate layer of modeling.

1. The operational capabilities of a gas network are described by the topology of the network and the characteristics of each of the network elements. These aspects can conveniently be modeled in terms of the mass flow q_a of each network element, the pressure p_u at each node, and the switching decisions s_a for each active elements. The basic assumption here is that the gas flow situation in terms of mass flows is given and fixed.
2. The next layer deals with the fact that the capacity contracts are not concluded in terms of mass flow $(q_u^{\text{ext}})_{u \in V_{\pm}}$, but instead in terms of thermal power $(P_u^{\text{ext}})_{u \in V_{\pm}}$. This makes it necessary to model a simple form of gas mixture.
3. Using flow reallocation options and the resulting re-balancing are market-based mechanisms. These effects are modeled as a bilevel problem with a linear second level problem, providing effective inflows $(P_u^{\text{eff}})_{u \in V_+}$ at each entry. This bilevel model depends on the market preference vector $(w_u)_{u \in V_{\pm}}$ that governs the re-balancing and the desired in- and outflow vector $(P_u^{\text{ext}})_{u \in V_{\pm}}$.

These three model layers are solved by an integrated method that is summarized in Fig. 3. We will use the rest of this subsection to outline the different parts.

2.2.1 Technical and physical network operation: Nomination validation for flow nominations

The modeling of physical behavior of natural gas and the technical constraints of the used network elements is described in great detail in [KHPS15], especially in [FGG⁺15] or a shorter version in [SKMP15]. We briefly outline the most important aspects of our modeling to indicate the complexity of the MINLP model used. Our basic assumption concerning the gas physics are the following.

- We consider a stationary gas flow situation given as a vector of balanced in- and outflows in terms of mass flow.

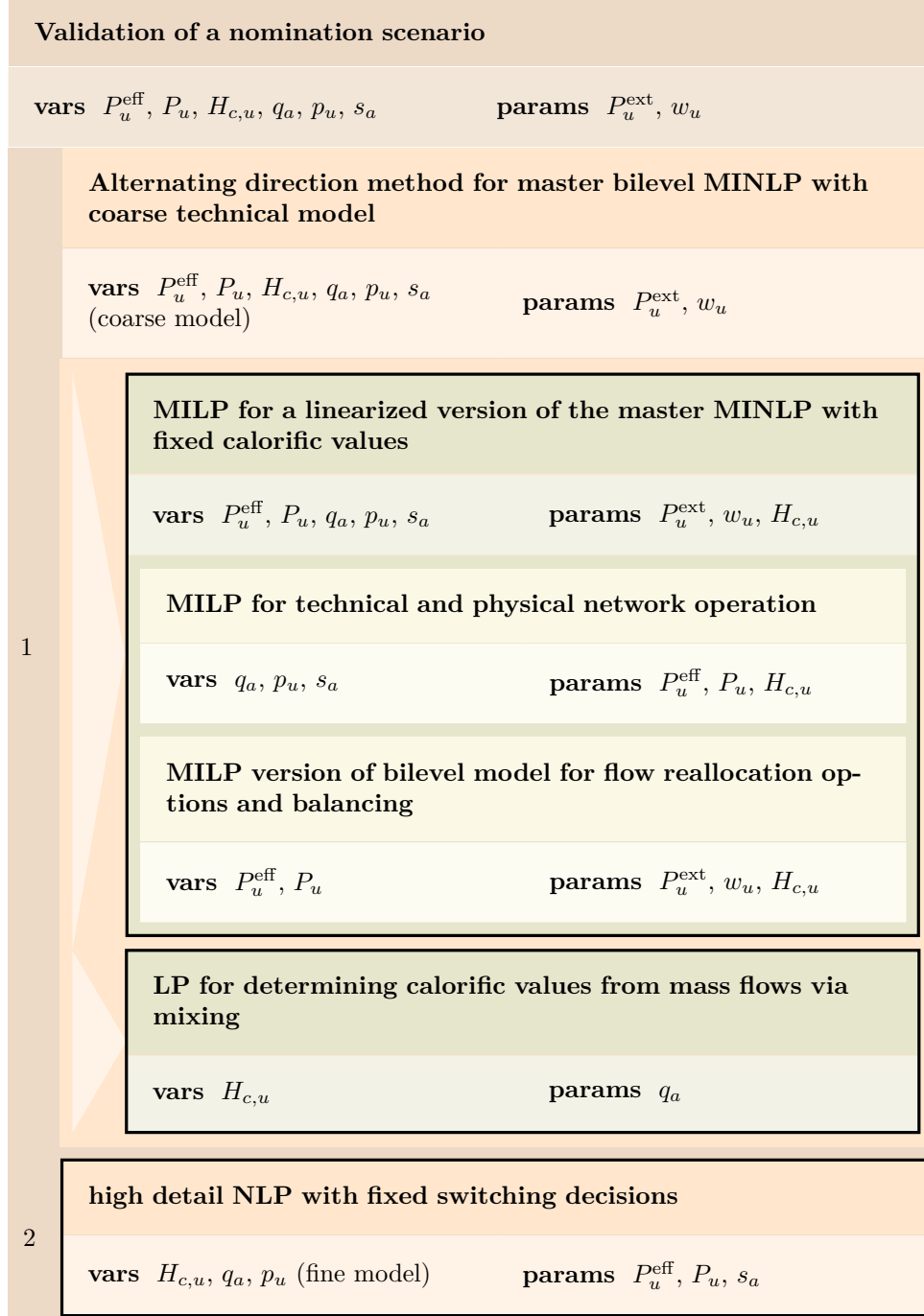


Figure 3: Structure of the models and methods used by nomination validation for checking a single nomination scenario. The models marked with thick lines are solved by an off-the-shelf solver as part of the overall method.

- We consider the isothermal case, i.e., we neglect the change of temperature due to the various physical effects involved. The model is based on a constant mean temperature calculated as a flow-weighted average according to the injected gas qualities.
- We ignore the mixing of gas and assume all parameters (molar mass, calorific value, norm density, pseudocritical pressure and temperature, molar heat capacity) of the gas to be constant with the value given by the respective flow-weighted average as for the temperature. (Calorific values will be handled specially by the method described in Section 2.2.2.)
- The so-called z -factor describing the deviation from a real gas is assumed to be constant along a network element.

At each node $u \in V$, there are constraints for bounds on the pressure p_u and (mass) flow conservation. For each network element $a = (u, v)$, there are bounds for the mass flow q_a . The various network elements are modeled as follows.

Pipes are assumed to be cylindrical with a constant slope. Together with the assumption of a constant (mean) z -factor this leads to the following well-known model for the pressure loss along a pipe

$$p_v^2 = \left(p_u^2 - \Lambda |q| q \frac{e^S - 1}{S} \right) e^{-S} \quad (4)$$

with suitable constants Λ and S (see [KHPS15, p. 27]).

Resistors are virtual elements to approximately model the pressure loss due to e.g., complex piping. There are two models for resistors, both of which incur a pressure loss in the direction of flow. This is handled via a binary variable for the flow direction. In the first form a resistor causes a nonlinear pressure loss according to a type of Darcy–Weisbach formula:

$$p_{\text{in}} - p_{\text{out}} = C \frac{q^2}{p_{\text{in}}}, \quad (5)$$

for some constant C depending on the parameters of the resistor (see [KHPS15, p. 28]). In the second and simpler form, the pressure loss is given by a fixed constant.

Valves are used to route the gas flow. A valve can be in two states, *open* and *closed*. An open valve admits arbitrary flow without changing the state of the gas, whereas a closed valve blocks the flow and decouples the gas flow at its ends. We model this behavior using a binary variable s_a and the constraints

$$\begin{aligned} s_a = 0 &\implies q_a = 0, \\ s_a = 1 &\implies p_u = p_v. \end{aligned} \quad (6)$$

A *control valve* is used to reduce the pressure in a controlled way. The working direction of a control valve is fixed. A control valve operates in one of three states: *active*, *closed*, or *bypass*. When *closed*, a control valve behaves like a closed valve; likewise, the *bypass* state corresponds to an open valve. If the control valve is *active*, the pressure at its outlet can be reduced to a given controllable value by at least $\underline{\Delta}_a$ and at most $\overline{\Delta}_a$. We use two binary variables to represent the states of a control valve: s_a^{ac} indicates whether it is active, and

s_a^{bp} whether it is bypassed. The closed state is encoded by $s_a^{\text{ac}} = s_a^{\text{bp}} = 0$. For a control valve *with remote access*, the state can be switched by the network operators. Such a control valve is modeled by the constraints

$$\begin{aligned} s_a^{\text{ac}} + s_a^{\text{bp}} = 0 &\implies q_a = 0, \\ s_a^{\text{ac}} = 1 &\implies \underline{\Delta}_a \leq p_u - p_v \leq \overline{\Delta}_a, q_a \geq 0, \\ s_a^{\text{bp}} = 1 &\implies p_u = p_v, \\ s_a^{\text{ac}} + s_a^{\text{bp}} &\leq 1. \end{aligned} \tag{7}$$

For control valves *without remote access*, there is a preset downstream pressure p_a^{set} and the state of the control valve is determined by the upstream and downstream pressure: If the upstream pressure is at least p_a^{set} and the downstream pressure is not greater than this value, then the flow is regulated such that the downstream pressure attains the value of p_a^{set} . Once the downstream pressure rises above p_a^{set} , the control valve without remote access closes automatically. If the downstream pressure is less than or equal to p_a^{set} and the upstream pressure drops below this threshold value, the control valve without remote access opens fully and is in bypass. This state-dependent behavior is modeled by the following additional constraints:

$$\begin{aligned} s_a^{\text{ac}} + s_a^{\text{bp}} = 0 &\implies p_v \geq p_a^{\text{set}}, \\ s_a^{\text{ac}} = 1 &\implies p_v = p_a^{\text{set}}, \\ s_a^{\text{bp}} = 1 &\implies p_v \leq p_a^{\text{set}}. \end{aligned} \tag{8}$$

To compensate for the pressure loss in the network, the TSO uses *compressors*. Typically the TSO operates a number of compressor stations at various points in the network, where a number of compressors are placed. The energy expenditure in these compressor stations accounts for the majority of the operating costs of the TSO. In coarse models, the only parameters of interest are bounds on the in- and outflow of the station and the maximum absolute and relative pressure increase that can be obtained. For a detailed physical model, one needs take into account that a compressor consists of a compressor machine, which compresses the gas and a compressor drive, which powers the compressor machine. We do not go into detail of compressor modeling in this section, but refer to the chapter [FGG⁺15] and to the report [HW16]. We note, however, that a detailed compressor model leads to a nonlinear nonconvex system of constraints.

In addition to the constraints for operating each network element, there are further constraints for the joint operation of the active elements. We model these via so-called *subnetwork operation modes* which specify a set of joint switching states for each element in a subnetwork. Typically, such a subnetwork corresponds to a bigger physical entity like a compressor or control valve station.

Another operational constraint arises from contractual restrictions. Due to the fact that some pipeline systems are co-owned by several TSOs, there is a need to share their transmission capacity between the owners. This is often done by limiting the so-called *transport moment* (or *payload distance*) along the co-owned pipeline. The transport momentum measures the amount of work necessary to move the gas and is computed as

$$\sum_{i=1}^n q_{a_i} L_{a_i}, \tag{9}$$

where (a_1, \dots, a_n) is a path given as a sequence of arcs and L_{a_i} denotes the length of the corresponding pipe.

The model sketched is a very complex MINLP model combining many non-convex constraints for the pressure drop on pipes/resistors and the operating ranges of compressor machines and drives with sophisticated switching decisions for the active elements and the subnetwork operation modes. In this integrated form, the model is not tractable yet. We therefore use a two-stage approach to tackle it:

1. In the first stage, we use a simplified physical and technical model to obtain switching decisions for operating the network.
2. The second stage uses these switching decisions to determine refined controls in the detailed and accurate physical and technical model.

As the second stage does not involve any integer variables any more, it is addressed by the NLP method described in [SSW15c]. The first stage, however, is still necessarily a nonconvex MINLP as the nonlinear pressure loss in pipes cannot be meaningfully simplified further. We therefore developed the following four distinct approaches exploiting different properties to tackle the hard problem of finding feasible switching decisions.

MILP approach This approach [GMMS15] transforms the MINLP into an MILP model using a general technique given by [GMMS12] building on prior work. This MILP is a relaxation of our original MINLP. Advantages of this technique include: a priori guarantees on the error of the solution with respect to the underlying MINLP model, certification of infeasibility of the original MINLP, independence of the MILP solver used. Extensions of this technique have been discussed in e.g., [GMS13, Gei11, Mor13, DGK⁺11, GKL⁺11].

Specialized MINLP approach This approach [HFH⁺15] uses a somewhat coarser MINLP formulation. This allows to use a specialized algorithm that is implemented in the general-purpose MINLP solver framework SCIP [Ach09, VG16]. Advantages of this technique include: certification of infeasibility with respect to the coarsened MINLP, short computation times due to tight integration with the solver.

Reduced NLP heuristic This heuristic [GSS15] reformulates the MINLP as a coarse NLP. The formulation chosen leads to very small NLPs which allows to check a large number of discrete settings in a short time. To reduce the number of settings to test, first a transshipment problem is solved on an aggregated graph and then analyzed. The advantages of this technique include: Adaptability to a specific network, computation speed. Extensions of this technique to the case of random loads have been discussed in [GHHS16].

MPEC heuristic This heuristic [SSW15b] reformulates all discrete decisions as complementarity constraints and uses standard regularization approaches to solve the resulting NLPs. Advantages of this technique include: computation speed, very detailed physical model. Extensions of this technique have been discussed in e.g., [RSSW16, Sch13, Sch15, SSW13].

To validate these approaches, a highly detailed NLP model was implemented [SSW15c, SSW15a, SSW16]. There are, however, difficulties when certifying infeasibility [JSSW15, JOP16]. As reported in [HHL⁺15b], all four approaches have their strengths and combining them provides a very powerful solver for the nomination validation problem. When considering the extensibility for different calorific values and the use of flow reallocation options (and hence nonconstant in- and outflows in terms of mass flow), however, the MILP approach turned out to be most flexible and robust.

2.2.2 Handling of different gas sources: Nomination validation for power nominations

So far, the model for nomination validation assumed that in- and outflows are specified in terms of mass flow $(q_u^{\text{ext}})_{u \in V_{\pm}}$. However, as described in the introduction, the desired in- and outflows $(P_u^{\text{ext}})_{u \in V_{\pm}}$ are specified in terms of thermal power. As the entries in our network, however, have distinct gas qualities, the mass flow needed to satisfy the exit demand is not known a priori. This means that we cannot simply assume that all gas is equal, but we have to take into account the mixing of different gas types. Such mixing problems are, without taking into account the gas physics, known as (generalized) pooling problems [MF09], which are notoriously hard. The corresponding models involve two new major difficulties:

- The coupling between the thermal power, the mass flow, and the calorific value introduces (many) nonconvex bilinear constraints.
- In the considered gas networks, for many arcs the flow direction is not known beforehand and depends on the network control. This leads to nonsmoothness, as the mixing condition discriminates between ingoing and outgoing arcs (w.r.t. flow), which is to be decided by the model.

To tackle this additional layer of complexity, a tailored method was developed, which we outline here. For more details, we refer to [GMSS15]. The method has also been extended to use more detailed compressor models [GMSS16b].

For thermal power, the gas quality parameter we need to track is the calorific value H_c . We assume that for each node and each arc we have introduced a variable $H_{c,u}$ for all $u \in V$ and $H_{c,a}$ for all $a \in A$. The main mixing condition for calorific values is then given by

$$H_{c,u} = \frac{H_{c,u}^{\text{ext}} q_u^{\text{ext}} + \sum_{a \in \mathcal{I}(u)} H_{c,a} |q_a|}{q_u^{\text{ext}} + \sum_{a \in \mathcal{I}(u)} |q_a|} \quad \text{for all } u \in V. \quad (10)$$

Here, we assume constant molar mass in the network and define $H_{c,u}^{\text{ext}} = q_u^{\text{ext}} = 0$ for all inner and exit nodes and $q_u^{\text{ext}} = q_u$ at entry nodes. The calorific value $H_{c,a}$ of an arc a is then given by

$$H_{c,a} = H_{c,u} \quad \text{for all } u \in V, a \in \mathcal{O}(u). \quad (11)$$

In our case, it is helpful to rewrite these conditions using variables for the thermal power P_u for all $u \in V$. Then using the relation $P_u = q_u H_{c,u}$ and the

assumption that our gas has constant molar mass, we can rewrite condition (10) and obtain

$$\sum_{a \in \mathcal{O}(u)} |P_a| - \sum_{a \in \mathcal{I}(u)} |P_a| = P_u \quad \text{for all } u \in V. \quad (12)$$

This is the constraint we use to model the mixing of thermal power. The resulting master MINLP is not efficiently solvable with the techniques described so far. Nevertheless, an alternating direction method (ADM) can be used to provide feasible solutions. Here, we only sketch the method, for details, we refer to [GMSS15, GMSS16b]. The method alternatingly solves the following two models that arise as restrictions of the master MINLP model. In this model, we use as objective function the deviation of the P_u from the desired outflows P_u^{ext} .

- In the first model, the calorific values $H_{c,u}$ are fixed and we optimize with respect to the network controls.
- In the second model, all other variables are fixed and we optimize with respect to the calorific values $H_{c,u}$.

The first model is a slight variation of the MINLP described above, which can be solved with the MILP approach. The second model is an LP which can be readily solved using standard solvers. To achieve satisfactory performance, further tweaks of the method are needed. The success of the method is maybe not that surprising, if one takes into account that it is closely related to feasibility pump heuristics [GMSS16a], which have been shown to be very successful in practice.

2.2.3 Flow reallocations and market behavior

The task of the TSO is the operation of the gas network infrastructure to satisfy the transportation demand. This involves the configuration of the technical components, such as valves and compressor. In addition, the TSO may use non-technical measures to help avoid bottlenecks. These include the *flow reallocation options* that the TSO can conclude with its transportation customers, i.e., the shippers. If any flow reallocation options are available, the nominated amount of gas injected into the network may be changed in favor of the TSO.

We consider three types of flow reallocation options, all acting on entry nodes only:

1. Minimum: the TSO may increase the amount of flow above the nominated value.
2. Reduction: the TSO may decrease the amount of flow below the nominated value.
3. Distribution: the TSO may reduce the flow at one entry and simultaneously increase the flow at another entry by the same amount, staying balanced.

These changes potentially introduce imbalance between injection and withdrawal, leading to infeasibility in stationary models of gas flow. Therefore, the nomination is *rebalanced* according to the preferences of the shippers (here understood as a unified agent). Here, the capacity contracts need to be taken

into account. Furthermore, the use of flow reallocation options *protects* certain nodes from the influence of the shippers.

In this operational view, decisions happen at three stages, sequentially, where each decision restricts the choices for the next stage.

1. TSO decides about usage of flow reallocation options.
2. Shippers decide about rebalancing of nomination.
3. TSO decides about technical configuration.

One can now canonically formulate a multi-level optimization problem consisting of three levels, corresponding to the stages above. A generic version is given by:

$$\begin{aligned}
& \underset{x,y,z}{\text{minimize}} && f_0(x,y,z) \\
& \text{subject to} && f_1(x) \leq 0 \\
& && \underset{y,z}{\text{maximize}} && g_0(y,z) \\
& && \text{subject to} && g_1(x,y) \leq 0 \\
& && && \underset{z}{\text{minimize}} && h_0(z) \\
& && && \text{subject to} && h_1(x,y,z) \leq 0
\end{aligned} \tag{13}$$

In our case, the first and third level are decided by the same agent and the objectives are identical: $f_0(x,y,z) = h_0(z)$, depending only on the technical network state z . The constraints $f_1(x)$ restrict the usage of the flow reallocation options x . The objective $g_0(y,z)$ is really a function of y only and represents the shippers preference for the nomination values. The second-level constraints $g_1(x,y)$ include both general rebalancing constraints, as well as restrictions stemming from the application of flow reallocation options through x . Finally, the third-level constraints $h_1(x,y,z)$ describe the physical constraints with respect to the technical control z based on the actually resulting nomination y and do not depend directly on the first-level decisions x . The simplified model reads:

$$\begin{aligned}
& \underset{x,y,z}{\text{minimize}} && h_0(z) \\
& \text{subject to} && f_1(x) \leq 0 \\
& && \underset{y,z}{\text{maximize}} && g_0(y) \\
& && \text{subject to} && g_1(x,y) \leq 0 \\
& && && \underset{z}{\text{minimize}} && h_0(z) \\
& && && \text{subject to} && h_1(y,z) \leq 0
\end{aligned} \tag{14}$$

This approach implies, however, that the shippers take into account the third-level problem. In reality, they are ignorant of the technical details of the gas network configuration and only know about the contractual constraints (that is, the non-physical network connecting the entries and exits via the virtual trading point). Therefore, the constraints $h_1(y,z)$ should be moved to the first level and the third level be eliminated altogether.

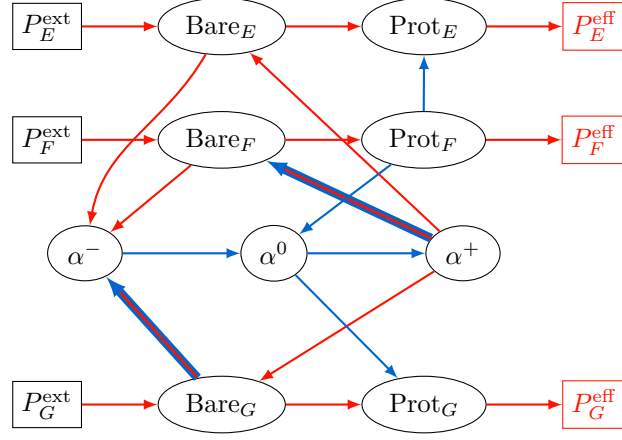


Figure 4: Flow network for follower problem in (15). Shown are three entries E, F, G with all types of flow reallocation options: reduction at F , minimum flow at G , and distribution $F \rightarrow E$. Rectangular nodes represent sources and sinks. The P^{ext} nodes on the left are sources with a fixed supply corresponding to the nominated value. The P^{eff} nodes on the right are sinks with a variable demand, as chosen by the shippers. The other nodes have no supply and demand. Three different types of arcs are used to model the flow redistribution. Follower arcs (in red) have a free flow variable of the follower ($y_a \geq 0$). Leader arcs (in blue) are fixed by a leader variable ($x_a = y_a \geq 0$). Finally, bounded arcs (red within blue) have a flow variable of the follower that is bounded by leader variables ($0 \leq x_a^{lb} \leq y_a \leq x_a^{ub}$).

This corrected formulation has only two levels:

$$\begin{aligned}
& \underset{x, y, z}{\text{minimize}} && h_0(z) \\
& \text{subject to} && f_1(x) \leq 0 \\
& && h_1(y, z) \leq 0 \\
& && \underset{y}{\text{maximize}} && g_0(y) \\
& && \text{subject to} && g_1(x, y) \leq 0
\end{aligned} \tag{15}$$

Within this formulation (15), the problem of nomination validation as described in Sections 2.2.1 and 2.2.2 is represented by the objective $h_0(z)$ and constraints $h_1(y^*, z)$, given a fixed y^* .

The “follower” problem of (15) has the form of a network flow in the y variables, with x acting as arc capacities. Figure 4 shows the structure of the artificial network that represents the distribution and rebalancing of flow injected at the entry nodes.

The objective of the follower (to be maximized) is a linear combination of the P^{eff} (demand) variables in the network 4. The coefficients are given by the market preference vector $(w_u)_{u \in V_{\pm}}$.

The bilevel problem in (15) can be transformed to a single-level optimization problem via the equivalent KKT reformulation of the follower problem [CMS07].

Consider the follower problem in the generic linear programming form, given by

$$\begin{aligned} & \underset{y}{\text{minimize}} && e^T y \\ & \text{subject to} && By = f - Ax \\ & && y \geq 0 \end{aligned} \tag{16}$$

We replace it with the corresponding optimality conditions

$$\begin{aligned} e - \lambda + \mu^T B &= 0 \\ By &= f - Ax \\ \lambda^T y &= 0 \\ y, \lambda &\geq 0 \end{aligned} \tag{17}$$

The complementarity constraint $\lambda_a \perp y_a$ can be expressed as a special ordered set of type 1 in the context of MIP solvers.

The leader variables x that model the choice and application of flow reallocation options are subject to additional constraints. For the case of flow reallocation options, consider the entry G in Fig. 4. We introduce a binary variable $x_G^{\text{useFC}} \in \{0, 1\}$ as an indicator for the use of the flow reallocation option. If $x_G^{\text{useFC}} = 1$, then the constraints $x_a^{\text{lb}} = x_a^{\text{ub}} = 0$ become active, where a is the arc from Bare_G to α^- . These “protect” the flow from rebalancing via α . Otherwise ($x_G^{\text{useFC}} = 0$), the constraints $x_b = 0$ (where b is the arc from α^0 to Prot_G), $x_a^{\text{lb}} = 0$, $x_a^{\text{ub}} = P_G^{\text{ext}} - \text{FirmCap}^{\text{lb}}$ are active. This means that no flow goes from other nodes (via α) to G and the nominated flow may be used for rebalancing.

3 Architecture of the ForNe System and current implementation

In this section we discuss the requirements and the solution approach taken in the ForNe System. The ForNe System consists of two parts of software: The computing core and the user interface. The computing core implements the aforementioned models, providing algorithms for the validation of flow and power nominations. The user interface offers several means to interact with the ForNe System and orchestrates the solution of high-level subproblems within the validation of flow nominations and the verification of booked capacities.

The ForNe System has the dual purpose of being a platform to allow computations in scientific quality and in a production setting at OGE. With scientific quality, we mean that the experiments should be as reproducible as possible. Thus in this setting only machines with identical hardware and software are used. Furthermore, every physical machine processes only one job at a time to avoid random effects such as cache misses which cause random noise in the computing times. In the production setting, the focus is on best possible use of the available resources and reliable operation.

3.1 Requirements and features

The ForNe System is able to compute solutions for several kinds of problems with the main objective to verify booked capacities. For this, the validation

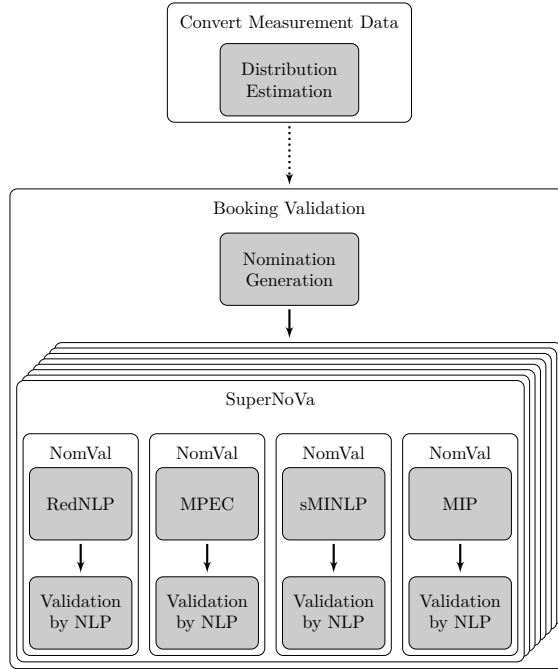


Figure 5: Overview of the architecture. Components with a white background are handled by the user interface and components with a gray background by the computing core.

of nominations is an important subproblem. As described in Section 2.2, the validation of a nomination is a two stage process. The first stage computes discrete decisions and a starting solution that is then refined by a highly detailed NLP model. For power nominations the alternating direction method can be used for validation or the nomination can be transformed into a flow nomination assuming a uniform calorific value of the gas at all entries and exists. For flow nomination validation, the four approaches described in Section 2.2.1 have been implemented and can be run in parallel where the first algorithm to find a feasible solution or proving that no such solution exists decides the result. We refer to this parallel nomination validation of flow nominations as SuperNoVa. The validation of a single flow or power nomination as well as the detailed NLP are implemented in the computing core where the respective models are built, solved and analyzed. The user interface is responsible to orchestrate the solution of the different models as well as the parallel solution within a SuperNoVa and the validation of nominations within a booking validation.

Figure 5 provides an overview of the process of the verification of booked capacities assuming the nominations are validated using the SuperNoVa approach. In a first step, a distribution model is estimated from the measured data as described in Section 2.1.1. In this step, the user interface only serves as a wrapper to submit a job to the cluster as the estimation of the distribution is done in the computing core. The resulting distribution is written to a file and can be used to verify different variants of booked capacities, i.e., in the assessment of which capacities can be published and which capacity requests accepted. The

dashed arrow depicts this human interaction. The validation of a booking is a process in two steps which is managed by the user interface. First, a number of nomination scenarios is generated as described in Section 2.1.2; a feature that is also implemented in the computing core. Then, these nominations are validated in parallel.

The results of all subproblems that are solved by the computing core as well as the building blocks of a booking validation are exposed to the users for in-depth analysis and to test and validate the approaches. Furthermore, straight forward mechanisms are in place to repeat the computation of the subproblems outside of the overall solution approach. To ensure maintainability and testability, the same code is called disrespecting if the call is done directly by the user or within the solution of a higher level problem such as booking validation.

Several interfaces are available to upload input data, start, stop, query the status and download the results of jobs. This includes a HTML-based web interface for human interaction and an XML-based web API allows an integration into OGE’s processes and infrastructure. The HTML interface also supports the upload of simple line based batch files where every line specifies one problem instance.

Most of the jobs are rather long running; ranging from several hours for the validation of a nomination to several days for the detailed validation of a booking. Proper job management has to ensure that the status and, upon termination, the results of a job are available and users are notified. Furthermore, the job has to support the abortion and in some cases also continuation of jobs. Proper termination and restart has to be ensured for the problem types where several components are solved individually and in parallel such as SuperNoVa or booking validation.

3.2 Implementation Details

In this section, we provide some more technical details about the ForNe System.

The computing core is implemented in C++ using the framework Lamatto++ [GMM]. It includes appropriate data structures, parsing facilities, advanced data checks on a logical level that go far beyond pure schema validation, preprocessing (e.g., flow and pressure propagation and network simplification), abilities to build various kinds of optimization models and algorithms developed in the project, and solution analysis. It contains interfaces to several solvers, including CPLEX [IBM], Gurobi [Gur], SCIP [Ach09, VG16], and Ipopt [WB06] (through GAMS [GAM]). Apart from the problems that are in the focus of this paper, the computing core contains algorithms for several related problems such as for gas network topology extension for a single [FHH⁺11, Hum14] and multiple scenarios [Sch17, SL16], and capacity maximization [Hay16].

The user interface is implemented in Python and uses Cherryypy [Che] as web framework. A class hierarchy for problem types enables code reuse. Each problem class contains a list of parameters and expected outputs from which the various interfaces and documentation is generated. This approach ensures high flexibility in the addition, modification and deletion of parameters which can easily become time consuming and error prone without such an infrastructure. Each problem class performs lightweight tasks directly and submits jobs to solve subproblems with the computing core to the cluster. For example, for

SuperNoVa four jobs are submitted to the cluster. Each of them determines the status of the SuperNoVa upon termination and possibly kills the remaining jobs as soon as the result of the SuperNoVa has been decided.

The computing cluster is hosted at the Zuse Institute Berlin and is managed by the cluster resources management system SLURM [YJG03]. Dependencies between jobs are used to trigger job evaluations and email notifications. Due to the limited length of the job queue (50000 jobs in our installation), an additional job queue is implemented in the interface as this quickly becomes a bottleneck, e.g., if several bookings are to be validated in parallel.

An instance of a problem is typically described by several input files (describing e.g., the network, the compressors, the contractual situation, the nomination scenario) and parameters (e.g., the formula for the real gas factor or the number of threads a MIP solver can use). As input files, in particular for rather static data such as network and compressor descriptions, are often used in multiple instances, all files have to be uploaded before actually starting the instance. Input files are stored on a network file system also accessible by the cluster nodes. Comparison of hash values ensures that the content of the files did not change between submitting and actually starting the jobs. Result files from the computation are first stored on the local disks of the cluster nodes and only copied to the network file system when the job is finished as otherwise the permanent writing of log files to the network file system becomes a bottleneck.

A SHA1 hash is computed for each instance taking into account the hashes of all input files and all parameter values. Considering the hash of the entire file content instead of the name of input files ensures that different file contents yield different instances which is important if automated tools are used. Composite problem types like SuperNoVa and booking validation know the hash values of their subinstances, allowing to easily display or download the results or to rerun a subinstance after possible code changes. This transparency was a major asset during development and debugging.

Some of the solvers used in the computing core are also capable of using multiple threads for parallel computations. In the scientific setting, only one job per cluster node is allowed anyway which may then use all its cores. In the production setting, we found it beneficial to limit the number of cores to be able to run more instances at a time. Especially in SuperNoVa this increases the throughput as it suffices if one of the four approaches decides the instance and it is beneficial to start more jobs simultaneously. SLURM ensures that the total number of threads used by all jobs scheduled on a computing node does not exceed its number of cores and that only one job is assigned to each node in the scientific setting.

4 Conclusions and outlook

The software system described has been used to conduct extensive computational experiments, comprising several months of computing time on a cluster with more than 1500 cores located at ZIB. All in all, millions of nomination scenarios were generated and solved in parallel. The system proved to be reliable and robust enough to handle this large-scale workload.

In a first phase, the goal was to establish the fitness for purpose on a relatively few selected instances as well as the necessary performance and scal-

ability for real-world use cases, where a single booking validation is supposed to be supported by 1000 or more nomination scenarios. To assess the fitness for purpose for nomination validation, OGE provided a small set of nomination scenarios that had been validated using standard simulation software. For testing the overall workflow for performance and scalability, OGE provided data for historical gas flows, capacity contracts and reallocation options, and the topology and technical parameters of the network both for their L-gas and H-gas network, respectively. This data was taken from the existing simulation-based process established at OGE [BEG⁺15], which was semi-automatically converted to the respective input formats of the ForNe system. (Due to the complexity of some of the models and the input data, some of the input had to be generated manually.)

A second phase addressed the fitness for purpose of the entire workflow of estimating distributions from the historical gas flows, generating nomination scenarios from sampled gas flows and capacity contracts and validating these nomination scenarios. It turned out that the initial quality of the input data was not sufficient to obtain results matching those of the established process. This resulted in many iterations with successively improved data. For finding and repairing data errors, researchers' experience and intimate knowledge of the implemented models and algorithms turned out to be invaluable.

Regarding the practical use of tools built on highly complex mathematical models, there are two conclusions from this observation. First of all, quality assurance for complex input data is crucial to obtain meaningful results and should be supported by tools (not just the data sanity checks of the optimization code) as far as possible. For use in practice, such tools need to be usable and appealing to the planners, i.e., without requiring a deep understanding of the underlying math. Secondly, it is necessary to provide, as far as possible, easily comprehensible means to understand the working of the underlying models and algorithms. Of course, this is very hard to achieve for general MIP/MINLP solvers, but the problem-specific high-level design of the algorithms often allows to derive insights from intermediate results. For instance, initial bound tightening combines input data from several sources, sometimes detecting infeasibility. With some additional effort, it is possible to indicate the set of data that lead to certain bounds, helping to pinpoint the data inconsistency.

All in all, the computational experiments established the feasibility and power of the proposed approach. However, they also showed several possibilities for improvement. Most importantly, the physical-technical model used in the first stage of nomination validation to determine the discrete control decisions turned out to be often too optimistic, i.e., it fails to detect situations that are technically infeasible. Detailed investigations revealed the following potential sources for this behavior:

- The MILP relaxations used to model pressure-loss in pipes and resistors are sometimes quite weak. Improvements in the initial bound tightening process may help to overcome this.
- The compressor modeling in the first stage of nomination validation neglects the drives' maximum power that sometimes limits the network capacity. This issue is addressed by novel compressor models [HW16] developed in a subproject of the recently established CRC/Transregio 154

- The relaxation arising from the alternating direction scheme for handling calorific values may be too coarse. In particular, due to its design this scheme is not very powerful for proving technical infeasibility as it is not an exact approach. More sophisticated decomposition schemes might be useful to tackle this issue.

The running time necessary for validating a single nomination scenario should also be improved substantially: It is usually well below one hour, but sometimes exceeds 4 days. This may be improved, for instance, by incorporating the ideas from the mentioned heuristic approaches in the current solver. There is also room for improving the overall running time of a booking validation by improving the order in which the corresponding nomination scenarios are validated.

We hope that our methods can be extended to provide a full analysis of the entry/exit system in the future and allow for a fully automated calculation of welfare-optimal technical capacities. For steps in this direction, see [Hay16]. Another relevant and challenging research direction is to extend these methods (or devise entirely new ones) towards network design. The goal is to extend or adapt the existing network such that it can accomodate the gas flow situations expected in the future.

5 Acknowledgements

First of all, we would like to thank Open Grid Europe GmbH for initiating and funding this research project. We also like to thank all our many colleagues who have been involved in ForNe, both from academia and OGE: Dagmar Bargmann, Mirko Ebberts, Armin Fügenschuh, Björn Geißler, Ralf Gollmer, Claudia Gotzes, Uwe Gotzes, Christine Hayn, Nina Heinecke, Holger Heitsch, René Henrion, Jesco Humpola, Imke Joormann, Veronika Kühn, Thomas Lehmann, Ralf Lenz, Hernan Leövey, Alexander Martin, Radoslava Mirkov, Andris Möller, Antonio Morsi, Antje Pelzer, Marc E. Pfetsch, Werner Römisch, Jessica Rövekamp, Martin Schmidt, Rüdiger Schultz, Klaus Spreckelsen, Marc C. Steinbach, Isabel Wegner-Specht, Bernhard M. Willert.

This work was supported by the *Research Campus Modal* funded by the German Federal Ministry of Education and Research (fund number 05M14ZAM), by the ICT COST Action TD1207 *Mathematical Optimization in the Decision Support Systems for Efficient and Robust Energy Networks*, and by DFG Collaborative Research Centre CRC/Transregio 154, *Mathematical Modelling, Simulation and Optimization Using the Example of Gas Networks* Subprojects: A04, B06, B07, B08, and Z01.

References

- [Ach09] Tobias Achterberg. SCIP: Solving Constraint Integer Programs. *Mathematical Programming Computation*, 1(1):1–41, 2009.
- [BEG⁺15] Dagmar Bargmann, Mirko Ebberts, Nina Geißler, Thorsten Koch, Veronika Kühn, Antje Pelzer, Marc E. Pfetsch, Jessica Rövekamp,

- and Klaus Spreckelsen. State of the art in evaluating gas network capacities. In Koch et al. [KHPS15].
- [Che] CherryPy – A Minimalist Python Web Framework. <http://cherrypy.org/>.
- [CMS07] Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Annals of operations research*, 153(1):235–256, 2007.
- [DGK⁺11] P. Domschke, B. Geißler, O. Kolb, J. Lang, A. Martin, and A. Morsi. Combination of nonlinear and linear optimization of transient gas networks. *INFORMS Journal on Computing*, 23(4):605–617, 2011.
- [FGG⁺15] Armin Fügenschuh, Björn Geißler, Ralf Gollmer, Antonio Morsi, Marc E. Pfetsch, Jessica Rövekamp, Martin Schmidt, Klaus Spreckelsen, and Marc C. Steinbach. Physical and technical fundamentals of gas networks. In Koch et al. [KHPS15], chapter 2, pages 17–43.
- [FHH⁺11] A. Fügenschuh, B. Hiller, J. Humpola, T. Koch, T. Lehman, R. Schwarz, J. Schweiger, and J. Szabó. Gas Network Topology Optimization for Upcoming Market Requirements. *IEEE Proceedings of the 8th International Conference on the European Energy Market (EEM), 2011*, pages 346–351, 2011.
- [GAM] GAMS. www.gams.com.
- [Gas10] Verordnung über den Zugang von Gasversorgungsnetzen (Gasnetzzugangsverordnung – GasNZV), 2010. Version released Sep. 03, 2010.
- [Gei11] B. Geißler. *Towards Globally Optimal Solutions for MINLPs by Discretization Techniques with Applications in Gas Network Optimization*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2011.
- [GGH⁺15] Nina Geißler, Uwe Gotzes, Benjamin Hiller, Jessica Rövekamp, and Thorsten Koch. Regulatory rules for gas markets in Germany and Europe. In Koch et al. [KHPS15].
- [GHHS16] Claudia Gotzes, Holger Heitsch, René Henrion, and Rüdiger Schultz. On the quantification of nomination feasibility in stationary gas networks with random load. *Mathematical Methods of Operations Research*, 84(2):427–457, 2016.
- [GKL⁺11] B. Geißler, O. Kolb, J. Lang, G. Leugering, A. Martin, and A. Morsi. Mixed integer linear models for the optimization of dynamical transport networks. *Mathematical Methods of Operations Research*, 73(3):339–362, 2011.
- [GMM] Björn Geißler, Alexander Martin, and Antonio Morsi. Lamatto++. Information available at <http://www.mso.math.fau.de/edom/projects/lamatto.html>.

- [GMMS12] Björn Geißler, Alexander Martin, Antonio Morsi, and Lars Schewe. Using piecewise linear functions for solving MINLPs. In Jon Lee and Sven Leyffer, editors, *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 287–314. Springer New York, 2012.
- [GMMS15] Björn Geißler, Alexander Martin, Antonio Morsi, and Lars Schewe. The MILP-relaxation approach. In Koch et al. [KHPS15].
- [GMS13] Björn Geißler, Antonio Morsi, and Lars Schewe. A new algorithm for MINLP applied to gas transport energy cost minimization. In Michael Jünger and Gerhard Reinelt, editors, *Facets of combinatorial optimization*, pages 321–353. Springer, Heidelberg, 2013.
- [GMSS15] Björn Geißler, Antonio Morsi, Lars Schewe, and Martin Schmidt. Solving power-constrained gas transportation problems using an mip-based alternating direction method. *Computers & Chemical Engineering*, 82, 2015.
- [GMSS16a] Björn Geißler, Antonio Morsi, Lars Schewe, and Martin Schmidt. Penalty alternating direction methods for mixed-integer optimization: A new view on feasibility pumps. Technical report, 2016. submitted.
- [GMSS16b] Björn Geißler, Antonio Morsi, Lars Schewe, and Martin Schmidt. Solving highly detailed gas transport MINLPs: Block separability and penalty alternating direction methods. Technical report, 2016. submitted.
- [GSS15] Ralf Gollmer, Rüdiger Schultz, and Claudia Stangl. The reduced NLP heuristic. In Koch et al. [KHPS15].
- [Gur] Gurobi. www.gurobi.com.
- [Hay16] Christine Hayn. *Computing maximal entry and exit capacities of transportation networks*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2016.
- [HFH⁺15] Jesco Humpola, Armin Fügenschuh, Benjamin Hiller, Thorsten Koch, Thomas Lehmann, Ralf Lenz, Robert Schwarz, and Jonas Schweiger. The specialized MINLP approach. In Koch et al. [KHPS15].
- [HHH⁺15] Benjamin Hiller, Christine Hayn, Holger Heitsch, René Henrion, Hernan Leövey, Andris Möller, and Werner Römis. Methods for verifying booked capacities. In Koch et al. [KHPS15].
- [HHL⁺15a] Holger Heitsch, René Henrion, Hernan Leövey, Radoslava Mirkov, Andris Möller, Werner Römis, and Isabel Wegner-Specht. Empirical observations and statistical analysis of gas demand data. In Koch et al. [KHPS15].

- [HHL⁺15b] Benjamin Hiller, Jesco Humpola, Thomas Lehmann, Ralf Lenz, Antonio Morsi, Marc E. Pfetsch, Lars Schewe, Martin Schmidt, Robert Schwarz, Jonas Schweiger, Claudia Stangl, and Bernhard M. Willert. Computational results. In Koch et al. [KHPS15].
- [HS16] Kai Hennig and Robert Schwarz. Using bilevel optimization to find severe transport situations in gas transmission networks. Technical Report 16-68, ZIB, Takustr.7, 14195 Berlin, 2016.
- [Hum14] J. Humpola. *Gas Network Optimization by MINLP*. PhD thesis, Technische Universität Berlin, 2014.
- [HW16] Benjamin Hiller and Tom Walther. Modelling compressor stations in gas network optimization. ZIB Report 16-62, Zuse Institute Berlin, 2016.
- [IBM] IBM CPLEX Optimizer. <http://www.cplex.com>.
- [JOP16] Imke Joormann, James B. Orlin, and Marc E. Pfetsch. A characterization of irreducible infeasible subsystems in flow networks. *Networks*, 68(2):121–129, 2016.
- [JSSW15] Imke Joormann, Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. What does “feasible” mean? In Koch et al. [KHPS15], chapter 11, pages 211–232.
- [KHPS15] Thorsten Koch, Benjamin Hiller, Marc Pfetsch, and Lars Schewe, editors. *Evaluating Gas Network Capacities*. MOS-SIAM Series on Optimization. SIAM, 2015.
- [MF09] R. Misener and C. A. Floudas. Advances for the pooling problem: Modeling, global optimization, and computational studies. *Appl. Comput. Math.*, 8:3–22, 2009.
- [Mor13] A. Morsi. *Solving MINLPs on Loosely-Coupled Networks with Applications in Water and Gas Network Optimization*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2013.
- [RSSW16] Daniel Rose, Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. Computational optimization of gas compressor stations: MINLP models versus continuous reformulations. *Mathematical Methods of Operations Research*, 83(3):409–444, 2016.
- [Sch13] Martin Schmidt. *A Generic Interior-Point Framework for Nonsmooth and Complementarity Constrained Nonlinear Optimization*. PhD thesis, Gottfried Wilhelm Leibniz Universität Hannover, 2013.
- [Sch15] Martin Schmidt. An interior-point method for nonlinear optimization problems with locatable and separable nonsmoothness. *EURO J. Comput. Optim.*, 3(4):309–348, November 2015.
- [Sch17] J. Schweiger. *Exploiting structure in nonconvex quadratic optimization and gas network planning under uncertainty*. PhD thesis, Technische Universität Berlin, 2017.

- [SHDA15] Jarig J. Steringa, Marco Hoogwerf, Harry Dijkhuis, and ET AL. A systematic approach to transmission stress tests in entry-exit systems. In *PSIG Annual Meeting*. Pipeline Simulation Interest Group, 2015.
- [SKMP15] Lars Schewe, Thorsten Koch, Alexander Martin, and Marc E. Pfetsch. Mathematical optimization for evaluating gas network capacities. In Koch et al. [KHPS15], chapter 5, pages 87–102.
- [SL16] J. Schweiger and F. Liers. A decomposition approach for optimum gas network extension with a finite set of demand scenarios. submitted, 2016.
- [SSW13] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. A Primal Heuristic for Nonsmooth Mixed Integer Nonlinear Optimization. In Michael Jünger and Gerhard Reinelt, editors, *Facets of Combinatorial Optimization*, pages 295–320. Springer Berlin Heidelberg, 2013.
- [SSW15a] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. High detail stationary optimization models for gas networks. *Optimization and Engineering*, 16(1):131–164, 2015.
- [SSW15b] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. An MPEC based heuristic. In Koch et al. [KHPS15].
- [SSW15c] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. The precise NLP model. In Koch et al. [KHPS15].
- [SSW16] Martin Schmidt, Marc C. Steinbach, and Bernhard M. Willert. High detail stationary optimization models for gas networks: Validation and results. *Optimization and Engineering*, 17(2):437–472, 2016.
- [vdH04] Tom van der Hoeven. *Math in Gas and the art of linearization*. PhD thesis, Rijksuniversiteit Groningen, 2004.
- [VG16] Stefan Vigerske and Ambros Gleixner. SCIP: Global Optimization of Mixed-Integer Nonlinear Programs in a Branch-and-Cut Framework. Technical Report 16-24, Zuse Institute Berlin, 2016.
- [WB06] A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.
- [YJG03] Andy B. Yoo, Morris A. Jette, and Mark Grondona. SLURM: Simple Linux Utility for Resource Management. In Dror Feitelson, Larry Rudolph, and Uwe Schwiegelshohn, editors, *Job Scheduling Strategies for Parallel Processing: 9th International Workshop, JSSPP 2003, Seattle, WA, USA, June 24, 2003. Revised Paper*, pages 44–60. Springer Berlin Heidelberg, 2003.