

# Gas Network Extension Planning for Multiple Demand Scenarios

Jonas Schweiger

**Abstract** Today's gas markets demand more flexibility from the network operators which in turn have to invest into their network infrastructure. As these investments are very cost-intensive and long-living, network extensions should not only focus on one bottleneck scenario, but should increase the flexibility to fulfill different demand scenarios. We formulate a model for the network extension problem for multiple demand scenarios and propose a scenario decomposition. We solve MINLP single-scenario sub-problems and obtain valid bounds even without solving them to optimality. Heuristics prove capable of improving the initial solutions substantially. Results of computational experiments are presented.

## 1 Introduction

Recent changes in the regulation of the German gas market are creating new challenges for gas network operators. Especially the unbundling of gas transport and trading reduces the influence of network operators on transportation requests. Greater flexibility in the network is therefore demanded. Traditional, deterministic planning approaches focus on one bottleneck scenario. Stochastic or robust approaches, in contrast, can consider a set of scenarios and therefore lead to more flexible network extensions.

Gas transmission networks are complex structures that consist of passive pipes and active, controllable elements such as valves and compressors. For planning purposes, the relationship of flow through the element and the resulting pressure difference is appropriately modeled by nonlinear functions and the description of the active elements involves discrete decisions (e.g., whether a valve is open or closed) (see [4, 5] for the details of our model and algorithmic approach to solve deterministic models). The resulting model is thus an *Mixed-Integer Nonlinear Program (MINLP)*.

---

J. Schweiger (✉)  
Konrad-Zuse-Zentrum für Informationstechnik Berlin,  
Takustraße 7, 14195 Berlin, Germany  
e-mail: schweiger@zib.de

In this presentation, we focus on additional pipes as extension candidates. A new pipe allows flow but also couples the pressures at the end nodes, possibly rendering previously feasible transport requests (also known as *nominations*) infeasible. An additional valve retains all possibilities of the original network. Opening the valve, corresponds to building the extension pipe and is therefore penalized the cost for the extension. Closing the valve forbids flow over the pipe which effectively removes the pipe from the system. Details on the approach for topology optimization for a single-scenario can be found in [1].

To approach the optimization over a finite set of scenarios (i.e., transport requests), we propose a scenario decomposition. Section 2 describes the model. The decomposition method is presented in Sect. 3 together with some details about primal and dual bounds and results on the ability to reuse solutions from previous optimization runs over the same scenario. Section 4 presents the results of computational experiments. Section 5 provides an outlook on planned future work on the topic.

## 2 Planning for Multiple Demand Scenarios

Assume a gas network, a set of scenarios  $\omega \in \Omega$ , i.e., nominations, and a set of extensions  $\mathcal{E}$  (each extension consisting of a pipe and a valve) is given. We denote the set of characteristic vectors of feasible extension sets for scenario  $\omega$  with

$$\mathcal{F}^\omega = \{x_E \in \{0, 1\}^{\mathcal{E}} \mid \text{Extensions } E \subseteq \mathcal{E} \text{ make } \omega \text{ feasible}\}$$

In our situation, a closed form description of  $\mathcal{F}^\omega$  is not at hand. However, we assume monotonicity in the sense that adding extensions to an element of the set is still feasible:

$$x_1 \in \mathcal{F}^\omega, x_2 \in \{0, 1\}^{\mathcal{E}}, x_2 \geq x_1 \implies x_2 \in \mathcal{F}^\omega.$$

Especially in the context of gas network planning this property cannot be taken for granted but adding valves to all extensions ensures monotonicity in our application.

For a specific scenario  $\omega$  the extension planning problem can now be stated as

$$\begin{aligned} \min c^T x^\omega & \quad (\text{SSTP}) \\ \text{s.t. } x^\omega & \in \mathcal{F}^\omega \end{aligned}$$

This formulation hides the difficulties in describing and optimizing over the set  $\mathcal{F}^\omega$ . Our approach uses problem (SSTP) as sub-problem and assumes a black-box solver to be available (e.g., from [1]).

In the multi-scenario extension planning problem we seek for a set of extensions of minimal cost such that the resulting network allows a feasible operation in *all* scenarios. We stress that in the different scenarios not all extensions that have been

built have to be actually used; in fact, using them might not even be feasible. The multi-scenario problem can then be stated as:

$$\begin{aligned}
 & \min c^T y && \text{(MSTP\_TS\_Node)} \\
 & \text{s.t. } x^\omega \in \mathcal{F}^\omega && \text{for all } \omega \in \Omega && (1) \\
 & x^\omega \leq y && \text{for all } \omega \in \Omega && (2) \\
 & y \in \{0, 1\}^{\mathcal{E}} && && (3)
 \end{aligned}$$

This is a two-stage stochastic program.  $y$  are the first stage variables which indicate which extensions are built. Finding a feasible operational mode for the scenarios given the extensions selected by  $y$  is the second stage problem.

### 3 Scenario Decomposition

The algorithmic idea is scenario decomposition. First, we solve the scenario sub-problems (SSTP) independently and in parallel. If one scenario sub-problem is infeasible, the multi-scenario problem is infeasible.

Branching on the  $y$  variables is used to coordinate the scenarios. To this end, we identify extensions that are selected in some but not all scenarios. Two sub-problems, i.e., nodes in the Branch&Bound tree, are created: one with the condition  $y_e = 0$  and one with the condition  $y_e = 1$ . In the two nodes, sub-problems have to be modified accordingly. For  $y_e = 0$ , variable  $x_e^\omega$  is fixed to zero. For  $y_e = 1$ , extension  $e$  is built and using it does not incur additional cost.

Each node of the Branch&Bound tree is identified by the sets  $E_0$  and  $E_1$  of extensions that are fixed to 0 and 1, respectively. The modified single-scenario problem for scenario  $\omega$  then reads:

$$\begin{aligned}
 & \min \sum_{e \notin E_1} c_e x_e^\omega + \sum_{e \in E_1} c_e && \text{(SingleScen}_\omega) \\
 & \text{s.t. } x^\omega \in \mathcal{F}^\omega \\
 & x_e^\omega = 0 && \text{for all } e \in E_0
 \end{aligned}$$

The following lemma states that adding more elements to  $E_0$  and  $E_1$  might only deteriorate the objective function value.

**Lemma 1** *Let  $E_0^1 \subseteq E_0^2$  and  $E_1^1 \subseteq E_1^2$  and  $c_i^*$  the optimal value of  $(\text{SingleScen}_\omega)$  with respect to  $(E_0^i, E_1^i)$ . Then  $c_1^* \leq c_2^*$ .*

Dual bounds for the single-scenario problems can be instantly translated into dual bounds for the multi-scenario problem.

**Lemma 2** *Let the objective function coefficients be non-negative, i.e.,  $c \geq 0$ . Then any dual bound for problem (SingleScen $_{\omega}$ ) for any scenario is also a dual bound for problem (MSTP\_TS\_Node).*

We propose three ways to get or enhance feasible solutions: First, by construction the union of all extensions used in the different scenarios constitutes a primal solution for the multi-scenario problem. Therefore, we construct a solution to (MSTP\_TS\_Node) in every node by setting  $y = \max_{\omega \in \Omega} x_e^{\omega}$  where  $x_e^{\omega}$  is taken as the best solution for scenario  $\omega$ .

Second, we observed that checking if a certain subset of extensions is feasible is typically very fast. This observation is used by a 1-opt procedure that takes the best current solution to (MSTP\_TS\_Node), removes one extension, and checks all scenarios for feasibility.

Third, in stochastic programming optimal single-scenario solutions often lack flexibility and do not occur in optimal solutions to the stochastic program (e.g., [7]). To benefit from all solutions the solver provides, we access its solution pool and store all sub-optimal solutions for the scenarios. This has two benefits. The solver might be able to use them as start solutions in the next node. On the other hand, we construct the “best known” solution so far by solving an auxiliary MIP.

### 3.1 Reusing Solutions

The Branch&Bound procedure solves slight modifications of the same problem over and over again. In some important cases, not all scenarios need to be solved again since we already know the optimal solution. As an example, take the extreme case where a scenario is found to be feasible without extensions. Clearly, the procedure should never touch this scenario again.

In order to decide whether a solution from a previous node can be reused, we need to take the fixations under which the solution was computed and the current fixations into account. In addition to the current fixations  $E_0$  and  $E_1$ , we define the sets  $E_0^S$  and  $E_1^S$  as the extensions that were fixed to the respective values when solution  $S$  was computed. We assume  $E_i^S \subseteq E_i$ , i.e., currently more extensions are fixed than when solution  $S$  was computed. Abusing notation, we identify the solution with the set of extensions it builds.

We start with the simple observation, that if all the extensions in a solution are already built (i.e.,  $y_e$  is fixed to 1), then the solution is optimal for the restricted problem:

**Lemma 3** *If  $S \in \mathcal{F}^{\omega}$  and  $S \subseteq E_1$ , then  $S$  an optimal solution for (SingleScen $_{\omega}$ ) for fixings  $E_0$  and  $E_1$ .*

If a solution is optimal for  $(E_0^S, E_1^S)$  and all extensions in  $E_1$  are part of the solution, then the solution is still optimal.

**Lemma 4** *Let  $S \in \mathcal{F}^\omega$  be an optimal solution to  $(SingleScen_\omega)$  given the fixations  $E_0^S$  and  $E_1^S$ . If  $E_1 \subseteq S$  and  $S \cap E_0 = \emptyset$ , then  $S$  is an optimal solution to  $(SingleScen_\omega)$  for fixings  $E_0$  and  $E_1$ .*

This is the situation, for example, after branching in the root node. In the 1-branch, a scenario which uses this extension does not need to be recomputed. In the 0-branch, solutions that did not use the extension remain optimal.

The situation becomes tricky if a solution does not use extensions that are already built but still uses unfixed extensions. The following lemma generalizes Lemma 4.

**Lemma 5** *Let  $S \in \mathcal{F}^\omega$  be an optimal solution given the fixations  $E_0^S$  and  $E_1^S$ . If  $E_1 \setminus E_1^S \subseteq S$  and  $S \cap E_0 = \emptyset$ , then  $S$  is an optimal solution to  $(SingleScen_\omega)$  for fixings  $E_0$  and  $E_1$ .*

## 4 Computational Results

We tested our approach on realistic instances from the gaslib-582 testset of the publicly available GASLIB [2, 5]. The GASLIB contains network data and flow scenarios that are distorted versions of the real data from the German gas network operator Open Grid Europe GmbH. The approach is implemented in the framework Lamatto++ [3]. Methods to solve the single-scenario problems and to generate suitable extension candidates were developed in the FORNE project. We used a time limit of 600s for the sub-problems which is reduced to 300s in the 1-opt heuristic. The total timelimit for set to 10h. The experiments were performed on Linux computers with two 64 bit Intel Xeon X5672 CPUs at 3.20GHz having 4 cores each such that up to 8 threads were used to solve the single-scenario problems in parallel.

Instances are composed from a pool of 126 infeasible instances that in single-scenario optimization find feasible solutions in the first 10 min. Table 1 summarizes the results. All but 3 instance are solved to proven optimality. The 3 instances that run into timeout each solve 3 nodes and then arrive to a point, where all extensions are fixed, but the single-scenario subproblem can neither find a feasible solution nor prove infeasibility.

**Table 1** Summary of computational results

Scenarios	Instances	Status		Nodes		Time (s)	
		Optimal	Timelimit	Avg	Max	Avg	Max
4	186	184	2	1.1	3	475	36017
8	90	89	1	1.3	3	660	36001
16	42	42	0	2.0	15	676	4598
32	18	18	0	3.9	14	1890	6080

## 5 Outlook

We presented a method for capacity planning with multiple demand scenarios. The computational experiments show the potential of our approach. Even though developed in the context of gas network planning, the few assumptions on the problem structure suggest the generalization to other capacity planning problems.

In the future, we also want to consider active elements (compressors, which can increase the pressure, and control valves, which can reduce it) as extension candidates. They possess 3 states: active, bypass, and closed. In case the element is not used in active mode, the abilities needed can be covered by a much cheaper valve. Then the binary “build”-“not build” decision is replaced by the three possibilities “build as active element”, “build as valve”, and “do not build”.

Last, we want to mention that Singh et al. [6] present an approach for capacity planning under uncertainty based on Dantzig-Wolfe decomposition. A comparison of our approach to theirs is future work.

**Acknowledgments** The author is grateful to Open Grid Europe GmbH (OGE, Essen/Germany) for supporting this work. Furthermore, the author wants to thank all collaborators in the FORNE project and all developers of Lamatto++.

## References

1. Fügenschuh, A., Hiller, B., Humpola, J., Koch, T., Lehman, T., Schwarz, R., Schweiger, J., Szabó, J.: Gas network topology optimization for upcoming market requirements. In: IEEE Proceedings of the 8th International Conference on the European Energy Market (EEM), pp. 346–351 (2011)
2. Gaslib: A library of gas network instances (2013). <http://gaslib.zib.de>
3. Geißler, B., Martin, A., Morsi, A.: Lamatto++ (2014). <http://www.mso.math.fau.de/edom/projects/lamatto.html>
4. Koch, T., Hiller, B., Pfetsch, M.E., Schewe, L. (eds.): From Simulation to Optimization: Evaluating Gas Network Capacities. MOS-SIAM Series on Optimization. SIAM—Society for Industrial and Applied Mathematics (2015)
5. Pfetsch, M.E., Fügenschuh, A., Geißler, B., Geißler, N., Gollmer, R., Hiller, B., Humpola, J., Koch, T., Lehmann, T., Martin, A., Morsi, A., Rövekamp, J., Schewe, L., Schmidt, M., Schultz, R., Schwarz, R., Schweiger, J., Stangl, C., Steinbach, M.C., Vigerske, S., Willert, B.M.: Validation of nominations in gas network optimization: models, methods, and solutions. *Optim. Methods Softw.* (2014)
6. Singh, K.J., Philpott, A.B., Wood, R.K.: Dantzig-wolfe decomposition for solving multistage stochastic capacity-planning problems. *Oper. Res.* **57**(5), 1271–1286 (2009)
7. Wallace, S.W.: Stochastic programming and the option of doing it differently. *Ann. Oper. Res.* **177**(1), 38 (2010)