Topological Reduction of Gas Transport Networks

Anton Baldin[(1)], Tanja Clees[(1,2)], Barbara Fuchs[(1)], Bernhard Klaassen[(1)], Igor Nikitin[(1)], Lialia Nikitina[(1)], Inna Torgovitskaia[(1)]

[(1)] Fraunhofer Institute for Algorithms and Scientific Computing, Sankt Augustin, Germany
[(2)] University of Applied Sciences Bonn-Rhein-Sieg, Sankt Augustin, Germany
Email: Name.Surname@scai.fraunhofer.de

***Abstract*—The paper presents the topological reduction method applied to gas transport networks, using contraction of series, parallel and tree-like subgraphs. The contraction operations are implemented for pipe elements, described by quadratic friction law. This allows significant reduction of the graphs and acceleration of solution procedure for stationary network problems. The algorithm has been tested on several realistic network examples. The possible extensions of the method to different friction laws and other elements are discussed.**

Keywords–modeling of complex systems; topological reduction; globally convergent solvers; applications; gas transport networks.

I. Introduction

The physical modeling of gas transport networks is comprehensively described in works [1]–[3]. The element equations for pipes vary from the simplest quadratic form to more complex formulae by Nikuradze, Hofer and Colebrook-White. In our paper [4], we have shown how to continue these formulae to the whole domain of model variables, in order to achieve a global convergence for non-linear solvers. Further, in paper [5] we have constructed a universal translation algorithm, capable of formulating network problems for non-linear solvers with arbitrary problem description language. In paper [6], we presented theoretical foundations of topological reduction methods for generic stationary network problems.

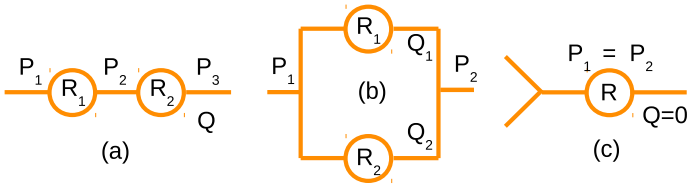

Figure 1. Main operations in GSPG reduction: series (a), parallel (b) connections to be reduced, contraction of a leaf (c).

In this paper, we continue the development of topological reduction methods in application to gas transport networks. Our motivation is to accelerate solution procedure for stationary gas network problems. The goal is to perform significant reduction of the graphs, preserving the accuracy of the modeling. The main idea is to reduce the series and parallel connections of elements in the network, with the operations, known in the theory of Series-Parallel Graphs (SPG) [7]. These operations can also be extended by contraction of a leaf, which after recurrent application contracts tree-like subgraphs, leading to Generalized Series-Parallel Graphs (GSPG) [8]. Such elementary operations are shown in Figure 1. In paper [6], we have estimated the efficiency of this method and shown on realistic gas transport networks that high reduction factors can be achieved. In our current work, we perform an actual implementation of the topological reduction for pipes, which form a considerable part of the networks.

In Section II, we present the details of a topological reduction procedure for pipes, modeled by quadratic friction law. In Section III, the results of numerical experiments with estimation of reduction factors and acceleration rates are given. In Section IV, we perform a comparison of our method with [9], which is also based on graph theory but using a different approach. In Section IV, we also discuss possible extensions of our method.

The described algorithms are implemented in the software MYNTS (Multi-phYsics NeTwork Simulator) [10], developed in our group.

II. Topological Reduction Algorithm for Pipe Networks

For the equations representing the pipes, one can use the simplest quadratic friction law from [1][9]:

$$P_{in}|P_{in}| - P_{out}|P_{out}| = RQ|Q|, \tag{1}$$

where $P_{in,out}$ are the input and output pressures and Q is the mass flow through the pipe. R is a resistance coefficient, depending on the pipe length L, diameter D, roughness parameter k, universal gas constant R_{gas}, temperature T, compression factor z and molar mass μ:

$$R = 16L/(\pi^2 D^5)/(2\log_{10}(D/k) + 1.138)^2 \times R_{gas} T z/\mu \cdot 10^{-10}. \tag{2}$$

All parameters are given in SI units (French, Système International), except of pressures, given in bar, hence the scale factor at the end of the formula. The structure of the term $Q|Q|$ ensures the symmetry of the equation when reversing the flow direction $Q \to -Q$. The similar structure of P-terms has a very special reason: it provides a monotonic continuation of the equation to the non-physical domain $P < 0$.

It was shown in [4] that, as a result of such continuation, the solver maintains stability also in the non-physical domain, where it can occasionally wander during the iterations. In addition, with such an extension, the system describing the stationary state of the network has a unique solution, even if the problem was set infeasibly. The simplest example of such an infeasible setting is to take a real network, such as shown in Figure 2, require a large throughput from suppliers to consumers, but at the same time switch off all the compressors. This problem, obviously, will not have a solution. On the other hand, if one uses the techniques from [4], the solution will exist and will be unique even in this case, but it will be located in the nonphysical domain $P < 0$. Thus, in this approach, one

 ISBN: 978-1-61208-732-0

has a necessary and sufficient *feasibility indicator*, lacking for other solvers, for which the infeasible statement of the problem is indistinguishable from the occasional divergence.

Let us consider the above described GSPG elementary operations for pipe networks.

The series connection is (see Figure 1a):

$$P_1|P_1| - P_2|P_2| = R_1 Q|Q|, \quad (3)$$
$$P_2|P_2| - P_3|P_3| = R_2 Q|Q|.$$

From here, we add the 2 formulas to get:

$$P_1|P_1| - P_3|P_3| = R_{12}^s Q|Q|, \; R_{12}^s = R_1 + R_2. \quad (4)$$

The inverse reconstruction of the eliminated variable P_2 is:

$$P_2|P_2| = P_1|P_1| - R_1 Q|Q|. \quad (5)$$

The parallel connection is (see Figure 1b):

$$P_1|P_1| - P_2|P_2| = R_1 Q_1|Q_1| = R_2 Q_2|Q_2|, \quad (6)$$
$$Q = Q_1 + Q_2.$$

From here, we solve this system for $Q_{1,2}$ to get:

$$P_1|P_1| - P_2|P_2| = R_{12}^p Q|Q|, \quad (7)$$
$$R_{12}^p = \left(R_1^{-1/2} + R_2^{-1/2}\right)^{-2}.$$

The inverse reconstruction of the eliminated variables $Q_{1,2}$ is:

$$Q_1 = Q/((R_1/R_2)^{1/2} + 1), \quad (8)$$
$$Q_2 = Q/((R_2/R_1)^{1/2} + 1).$$

Contracting the leaf, see Figure 1c, in the simplest case of zero flow results in the removal of P_2, Q variables. The inverse reconstruction consists of the setting $Q = 0$ and copying $P_2 = P_1$.

It should be noted that there are two types of source/sink nodes in gas networks. Q_{set} is the node in which the flow is set. P_{set} is the node where the flow is not fixed, but the pressure is set. For parallel connections, nodes of this type at the ends do not pose a problem. For series connections, the presence of such specifiers in the intermediate node leads to deviations from Kirchhoff's law and represents an obstacle to the reduction. Next, we discuss a special algorithm that allows to move the Q_{set} specifiers over the network. In combination with it, the reduction can be continued.

For contraction of the leaf, the P_{set} specifier represents an obstacle, because when shifting to the neighboring node, the P_{set} specifier gets an unfixed pressure value that depends on the flow. To contract a leaf with the Q_{set} specifier, two options are possible. First, block contracting leafs with a nonzero Q_{set}. As a result, the reduction will be incomplete, but the end Q_{set} nodes will be intact, which is convenient for formulating scenarios with different values of Q_{set} and for controlling the feasibility condition $P > 0$ at endpoints. Second, allow such leafs to be moved, with Q_{set} moving to the other side and summing it up with another Q_{set} that may be located there. For the inverse reconstruction, the value of Q_{set} must be saved, after that the inverse operations can be performed. The pressure at the free end is not determined by simple copying, but is found from the equation of the element:

$$P_2|P_2| = P_1|P_1| - RQ_{set}|Q_{set}|. \quad (9)$$

III. THE RESULTS

We have implemented GSPG reduction algorithm with fixed Qsets and tested it on three realistic networks. The simplest network N1 is shown in Figure 2. It includes 4 compressors (2 stations with 2 compressors each), 2 Psets (shown by rhombi n56, n99) and 3 Qsets (triangles n76, n80, n91). Originally (level0), the network contains N=100 nodes and E=111 edges, including P=34 pipes. Then (level1), a topological cleaning algorithm from [6] is used, removing (if any) parts of the graph, disconnected from pressure suppliers, as well as contracting superconducting edges, such as shortcuts, open valves and short pipes ($D = L = 1$ m). This operation is absolutely necessary for the stability of the solver, since disconnected parts possess undefined pressure and loops of superconducting edges have undefined circulating flow. This level of reduction looks similar to level0, just some valves, shortcuts and internals of stations are removed. The total count on this level is N=39, E=40, P=34.

TABLE I. PARAMETERS OF TEST NETWORKS

network	compressors	regulators	Psets	Qsets
N1	4	0	2	3
N2	7	18	4	64
N3	25	54	6	290

TABLE II. NODES:EDGES:PIPES COUNT FOR DIFFERENT REDUCTION LEVELS

network	level0	level1	level2	level3
N1	100:111:34	39:40:34	13:14:8	8:9:3
N2	973:1047:500	528:541:479	198:208:146	126:134:72
N3	4721:5362:1749	1723:1814:1666	705:755:607	296:332:184

TABLE III. TIMING FOR TWO REDUCTION LEVELS*

network	level1		level2	
	filter	solve	filter	solve
N1	0.006	0.044	0.009	0.02
N2	0.063	0.5	0.09	0.196
N3	0.243	2.103	0.371	0.944

* in seconds, for 3 GHz Intel i7 CPU 8 GB RAM workstation; 'filter' includes removing disconnected parts and superconductive elements (for level1,2) and GSPG reduction (for level2); 'solve' includes translation procedure, actual solving and extracting the result; the actual solving is performed with IPOPT.

After that (level2), GSPG reduction with fixed Qsets is applied, leaving N=13, E=14, P=8 elements. This corresponds to the reduction factor 2.9. Then, we have implemented all necessary GSPG operations described by the formulae above. For the solution procedure, after the reduction, we obtain the acceleration factor 2.2. The solution on level2 is identical with level1 up to the solver tolerance (set to tol=10^{-5} in our numerical experiments).

 ISBN: 978-1-61208-732-0

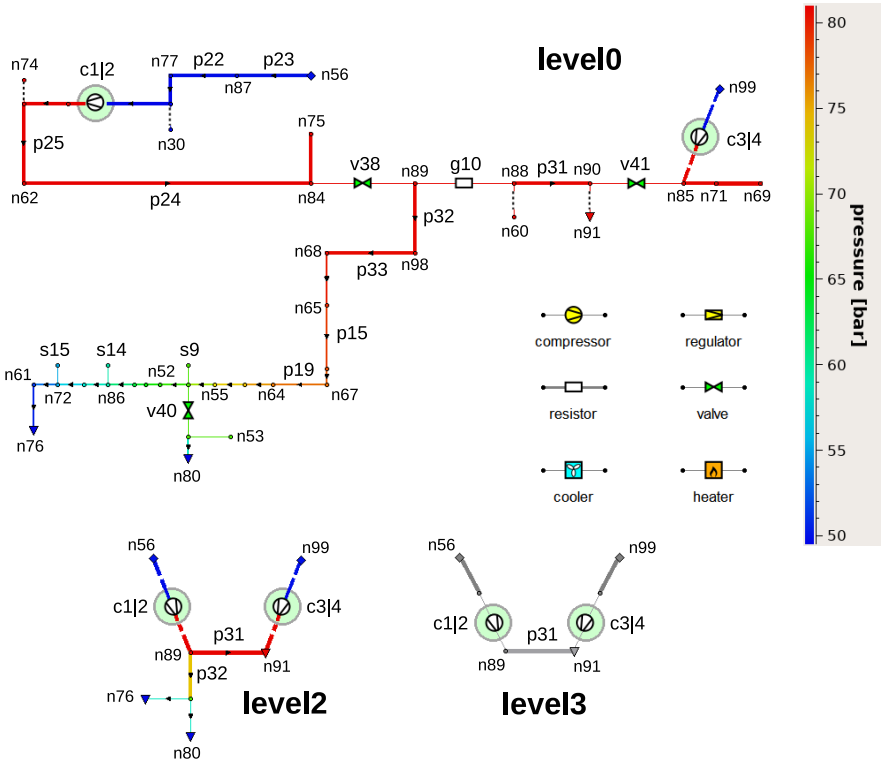

Figure 2. Realistic gas transport network N1 at different reduction levels: level0 = original network; level2 = GSPG reduction with fixed Qsets; level3 = GSPG reduction with moving Qsets. (Not shown: level1 = removing disconnected parts and superconductive elements, looking similar to level0.)

For GSPG reduction with moving Qsets (level3), we have implemented the formal reduction algorithm, sufficient for the estimation of the reduction factor. On this level, we have N=8, E=9, P=3 elements, comprising the reduction factor 1.6 relative to the previous level. The numerical counterpart of the algorithm has not been implemented yet, that is why the reduced network for level3 on Figure 2 does not have pressure data. In the next section, we will discuss the details of Qset movement algorithm necessary for this level.

The same tests have been performed on more complex networks N2 and N3, provided by our industrial partners for benchmarking. The parameters of the networks and the results of the reduction are presented in Tables I-III. The obtained level1/level2 reduction factors vary in the range 2.4-2.9, while acceleration factors solve1/solve2 are 2.2-2.6. The 'filter' step in Table III includes the necessary preprocessing and reduction of the networks. The 'solve' step includes translation of the network to the form suitable for the solver and the solution procedure itself, which share the timing in 1:1 proportion. Currently, our system uses the universal translation algorithm from [5]. It allows to plug in generic non-linear solvers with an arbitrary problem description language, requiring only to adjust a translation matrix in the algorithm. In particular, we have experimented with IPOPT (Interior Point OPTimizer) [11], Mathematica [12], MATLAB (MATrix LABoratory) [13] and a Newton solver, developed in our group. The best results for our type of problems have been obtained with IPOPT and Newton, while these two solvers among themselves have comparable performance. The details of the implementation of the Newton solver will be published elsewhere.

The solution procedure involves a multiphase workflow, described in [5]. Although global convergence from an arbitrary starting point for stationary network problems is guaranteed theoretically [4], the multiphase procedure is still empirically faster. This procedure gradually increases the complexity of the modeling and uses the result of the previous phase as a starting point for the next one. In our numerical experiments, a 3-phase procedure is used, relevant to the modeling of compressors and regulators in the network. In the first phase, compressors and regulators have enforced goals, e.g., $P_{out} = Const$. Then, they are set to a simplified universal *free model* and, finally, to the individually calibrated *advanced model* [6]. The timing in Table III presents the sum over 3 phases.

IV. Discussion

At first, we perform a comparison with paper [9], where a different approach for topological reduction was taken. Then, we describe possible generalizations of our topological reduction algorithm.

a) Comparison with paper [9]: in this paper, the stationary problem in gas transport networks was studied, where subgraphs consisting of pipes only were considered. The pipes were modeled by the expressions of type (1) and the 2nd Kirchhoff law was consistently applied, by summing this expression over independent cycles in the subgraph. As a result, P-variables drop off from such sums and a system of smaller size depending only on Q-variables remains, for which the existence and uniqueness of the solution is proven.

Although the approach looks promising, for its practical implementation, some problems exist.

 ISBN: 978-1-61208-732-0

This approach does allow to reduce the dimension of the system by extracting from it a subsystem that depends only on Q-variables. The dimension of the subsystem is equal to the number of independent cycles in the subgraph. The subsystem has a unique solution for which, however, it is generally impossible to obtain an analytic expression. Thus, it should be solved numerically, for example, by Newton's method. The remaining variables in the subgraph are obtained by an unambiguous analytical reconstruction procedure. The problem appears when this subgraph is considered in the context of a complete graph containing other elements than pipes, for example, compressors. The solution of the complete problem is usually also found by the Newton's method. For the subgraph, this means that the solution must be found many times, with variable boundary conditions. In this case, a combination of two Newton's methods, external and internal, will require from the subgraph not only a solution, but also its derivatives with respect to the boundary conditions. Such a combination is in any case not an efficient way to solve the system.

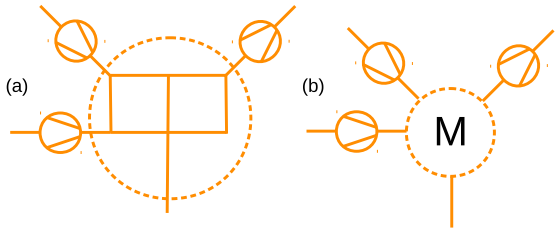

Figure 3. Shrinking a subgraph (a) creates a generalized network element with a fixed number of pins (b), a multipin (M).

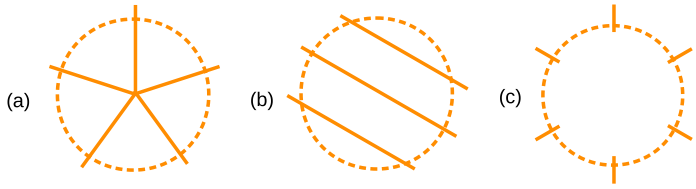

Figure 4. Particular examples: (a) 5-pin star; (b) 6-pin with 3 parallel connections; (c) empty 6-pin. In all cases, the number of equations describing the multipin is equal to the number of pins.

Another problem is that, according to [9], a pure pipe subgraph, contained in a general graph, can be shrunk to a single point. We cannot agree with this statement, since a subgraph can have many boundary points in which nodal P-variables are different, see Figure 3. The subgraph is not shrunk to a point, but to a generalized element containing N_b boundary points, or *pins* like in a microchip. We refer to such a generalized element further as *multipin*. As we show below, this element introduces not one, but N_b equations.

Without loss of generality, we can consider a connected subgraph for which the pins have definite flows serving as source/sink boundary conditions for the subgraph, as well as N_b nodal P-variables. One condition necessary for the stationary problem is the annulation of the sums of boundary flows. Here, for definiteness, we place all external sources/sinks in the subgraph, including Q_{set} and P_{set} nodes, on separate pins. Further, considering one of the boundary nodes as a point with a given pressure, the procedure from [9] uniquely reconstructs all other N_b-1 boundary pressures in terms of the first pressure and the boundary flows. The conditions for the equality of the reconstructed pressures to the given boundary pressures are the equations presenting the multipin for the external graph, totaling N_b equations.

In principle, it seems possible to precompute these N_b functions on a grid in the space of parameters and use fast interpolation algorithms to represent the multipin. The problem is the rapid increase of the grid data volume with the increasing dimension of N_b. In our approach, we have restricted our calculations to 2-pins, $N_b = 2$, which generally allows 2D tabulation (the pixel buffer from [6]). In this paper, we concentrate on the quadratic pipe model (1), which allows to encapsulate all the characteristics in one R-parameter and to perform all calculations analytically, without tabulated functions. Below, we consider also an intermediate case, where 1D-tabulation by splines is used. Thus, we avoid *curse-of-dimensionality* problems existing for general multipins and are still capable to reduce the dimension of the problem considerably.

In the remainder of this subsection, we consider in more detail an interesting question, why the multipin, regardless of its structure, is described by the same number of equations. Indeed, the number of equations external to the excluded subgraph is the same and does not depend on the topology of the subgraph. After eliminating the subgraph, the system must remain closed, meaning that the subgraph introduces the same number of equations. To calculate this number, it is enough to consider a specific configuration.

In Figure 4a, the star-like multipin is considered. One equation is the zero sum of the flows into the multipin. The Kirchhoff law in the center is equivalent to this equation. There is one P-variable in the middle, but there are also N_b conditions relating it to the boundary P_b and Q_b. In total, N_b-pin is equivalent to N_b equations on the boundary P and Q. Figure 4b shows the case when N_b is even and N_b-pin represents $N_b/2$ conditions for equality of incoming and outgoing flows, as well as $N_b/2$ of element equations. In total, we obtain N_b equations. In fact, even the connectivity of the graph is not important here. In Figure 4c, the case of an empty subgraph is considered, when all pins hang freely. Then, $Q_b = 0$ in all of them, comprising N_b equations.

b) Possible generalizations of friction laws: in the equations of the element, a general power dependence can be used, as was done in [9]. The consideration is quite similar. The element equation, series and parallel connections are described by:

$$P_{in}|P_{in}| - P_{out}|P_{out}| = RQ|Q|^{\alpha-1},\ \alpha \geq 1, \qquad (10)$$
$$R^s_{12} = R_1 + R_2,$$
$$R^p_{12} = \left(R_1^{-1/\alpha} + R_2^{-1/\alpha}\right)^{-\alpha}.$$

The quadratic law (1) corresponds to $\alpha = 2$.

Contraction of the leaf and reverse reconstruction are done in the same way.

Consider a more general case:

$$F(P_{in}) - F(P_{out}) = G(Q), \qquad (11)$$

where F, G are monotonously increasing functions, every element has an own G, while F is the same for all elements (strictly speaking, it is enough if F is the same in a connected component of the graph).

 ISBN: 978-1-61208-732-0

For series connections, the equations can be combined as before:

$$F(P_1) - F(P_3) = G^s_{12}(Q), \qquad (12)$$
$$G^s_{12}(Q) = G_1(Q) + G_2(Q).$$

If the original functions were monotonic, then their sum will also be. The inverse reconstruction is:

$$P_2 = F_{inv}(F(P_1) - G_1(Q)), \qquad (13)$$

where by subscript *inv* we denote the inverse 1D-function, so as not to be confused with the algebraic inversion: $x^{-1} = 1/x$.

For parallel connections, the equations can be combined analogously:

$$F(P_1) - F(P_2) = G^p_{12}(Q), \qquad (14)$$
$$G^p_{12} = (G_{1,inv} + G_{2,inv})_{inv}.$$

Proof:

$$F(P_1) - F(P_2) = x = G_1(Q_1) = G_2(Q_2),$$
$$Q_1 = G_{1,inv}(x),\ Q_2 = G_{2,inv}(x),\ Q =$$
$$Q_1 + Q_2 = G_{1,inv}(x) + G_{2,inv}(x) = G^p_{12,inv}(x),$$
$$x = G^p_{12}(Q) = (G_{1,inv} + G_{2,inv})_{inv}(Q). \blacksquare$$

It can be seen that the resulting G-function is also monotonic. The structure of the formulas for quadratic and α-power resistance is also clear: the inverse of the power function is also a power function. Thus, the inverse reconstruction is:

$$Q_1 = G_{1,inv}(G^p_{12}(Q)),\ Q_2 = G_{2,inv}(G^p_{12}(Q)). \qquad (15)$$

To store 1D functions $y(x)$, one can use lists of tabulated values (x_n, y_n) and interpolate between them using cubic splines. Outside the working area $|P| \le 150$ bar, $|Q| \le 1000$ Nm^3/h, the data can be extended by linearly growing functions, similar to [4]. Such a representation is convenient for inverting the functions, for which it suffices to swap $(x_n, y_n) \to (y_n, x_n)$ and reconstruct the splines [14]. The accuracy of this procedure is controlled by the smoothness of the function and the density of subdivision. The computational complexity is proportional to the number of tabulated values, $O(N)$.

In the problems we are considering, the functions are odd: $y(-x) = -y(x)$. This means that it is enough for them to construct splines in the region $x \ge 0$ and use the symmetry for complete reconstruction. In addition, the functions have a vanishing derivative at zero, for example, $y = x|x| = x^2 \operatorname{sgn} x$, which leads to a non-smooth root dependence for inverse functions $x = \sqrt{|y|} \operatorname{sgn} y$. This leads to problems for representing such functions by cubic splines. In fact, as noted in [4], vanishing of the derivative also leads to instability of the solver. The case $Q = 0$ can occur in large regions of the network in the absence of a flow in them. This leads to zeroing of the derivative of the function $Q|Q|$ and entails the degeneration of the Jacobi matrix of the complete system. To overcome this problem, the laminar term $Q|Q| + \epsilon Q$ must be added to this function; similar regularizing terms must also be added to the P-functions. After this, the problem with the zero derivative disappears and does not hinder the spline inversion.

c) Precise friction laws: better precision can be achieved by Nikuradze and Hofer formulae [2][3]. These differential formulae can be analytically integrated under assumption of slow variation of temperature and compression factor over the pipe. If needed, the long pipes can be subdivided into smaller segments to achieve the necessary precision of the modeling. This piecewise integration approach is similar to the *finite element method* in modeling of flexible materials, flow dynamics, etc. The resulting formulae have the same quadratic form (1), with the resistance $R(Q, P_1, P_2)$ weakly (logarithmically) dependent on the flow and the pressures. Direct comparison between the quadratic and Hofer pipe laws on our test networks shows the difference on the level of 7-10%. The practical use of calculations with the approximate quadratic formula is a rapidly computable starting point for the subsequent refinement iterations with the precise formula. The gravitational term, available in the precise formula and taking into account the profile of the terrain, can be also embedded in the quadratic formula:

$$P_1|P_1|(1+\gamma) - P_2|P_2|(1-\gamma) = ... \qquad (16)$$
$$\gamma = \mu g(H_1 - H_2)/(R_{gas} T z),$$

where the dots denote the flow-dependent right part, in any form that we have considered. The dimensionless hydrostatic factor γ is determined by the gravitational acceleration g, the height difference $H_1 - H_2$ and the usual gas parameters. In real problems, the parameter γ is small, $|\gamma| \ll 1$, so the factors $(1 \pm \gamma)$ do not change the signature of the terms in the equation.

d) Inverse reconstruction: for practical purposes, it is enough to solve the problem on the reduced graph, the topological skeleton. The users are mainly interested in the values of flows and pressures at the end points of pipe subgraphs, where they are connected to active elements such as compressors and regulators or directed to the end consumers. One also needs to control the feasibility indicator $P > 0$. As we now show, it is enough to control this indicator at the endpoints.

Consider GSPG operations in the presence of nodes with negative pressure. For parallel connection, in the presence of negative pressure in the end node, it remains there after the reduction. For series connection, if there is negative pressure at the intermediate node, it will also be negative at the end node downstream. Indeed, considering the most general case with gravity corrections,

$$P_3|P_3|(1-\gamma) = P_2|P_2|(1+\gamma) - R_2 Q|Q|, \qquad (17)$$

since the factors $(1 \pm \gamma), R_2$ are positive, for $P_2 < 0$, $Q \ge 0$, we get $P_3 < 0$. Only contraction of a leaf with $Q_{set} > 0$ can be a problem, since this procedure can hide a negative pressure node downstream. As we have already explained, there is an option to block contracting leafs with nonzero Q_{set}. In this case, it suffices to check $P > 0$ at the end nodes of the pipe graph.

On the other hand, the data recovery in reduced elements is a straightforward analytical procedure. For this, a complete reduction history with all intermediate parameters and/or tabulated functions must be recorded. Then, the above-described inverse operations can be applied. On the graph obtained, it is possible to monitor the fulfillment of the condition $P > 0$ or the enhanced condition $P > 1$ bar or any other inequality on pressures and flows.

e) Level 3, Q_{set} movement algorithm: consider the two graphs depicted in Figure 5. Assuming that the central element is described by the general equation $F(P_1, P_2, Q)$, we require the equivalence of solutions, connecting these equations with the shift transformation of the argument:

$$F_b(P_1, P_2, Q) := F_a(P_1, P_2, Q + Q_{set}), \qquad (18)$$
$$F_a(P_1, P_2, Q) = 0 \;\Rightarrow\; F_b(P_1, P_2, Q - Q_{set}) = 0.$$

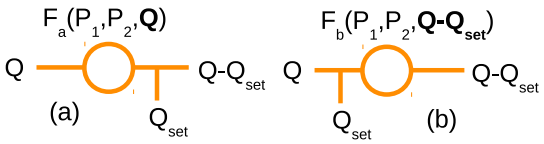

Figure 5. Q_{set} movement algorithm.

As a result, it is possible to move the Q_{set} specifier along a graph to an arbitrary place. For example, all Q_{set} specifiers can be moved to the P_{set} node, which should be present in each connected component of the graph. In this case, the undefined flow in this node will be shifted by the total Q_{set} in the subgraph. Alternatively, one can move all Q_{set} specifiers into one main consumer, who will represent all consumers in the subgraph. Note that such transformations change the distribution of flows in the graph, representing only a virtual distribution, that is visually unsimilar, but mathematically equivalent to the original one. To represent the result, of course, all the displaced Q_{set} specifiers must return to their places using inverse transformations. Note also that the argument shifts change the position of zero and violate the oddness of the functions. This requires to modify the tabulation algorithms; the easiest way is to consider all dependencies as monotonic functions of general form.

f) Not only pipes, combining 2D characteristic maps: after all pipe subgraphs are reduced to 2-pins, the functions can be transformed to a more general representation, in one of the equivalent forms:

$$Q = F(P_1, P_2), \;\; P_1 = F(P_2, Q), \;\; P_2 = F(P_1, Q). \quad (19)$$

All other elements, such as compressors and regulators, can be represented in the same way. Such a representation can use the 2D-tabulation (pixmaps) algorithms described in [6], as well as piecewise linear monotone extensions outside of the working region. As a result, GSPG reduction can be continued at the level of 2D functions. Thus, our proposed strategy is to keep the low-dimensional representations as long as possible, such as quadratic equations or 1D-splines for pipes, and then, after the network is strongly reduced, proceed to pixmaps.

V. Conclusion

In this paper, the topological reduction method for gas transport networks has been presented. The method uses a contraction of series, parallel and tree-like subgraphs, containing the pipes, described by quadratic friction law. This way, we achieve the goal of significant lossless reduction of the graphs and we accelerate solution procedure correspondingly. Several realistic network examples of different complexity have been used for the benchmarking of the method. Comparing with the original network (level0), the elimination of superconductive elements and disconnected parts (level1) brings the reduction factor into the range 1.9-2.9, further GSPG reduction with fixed Qsets (level2) multiplies it by the factor 2.4-2.9, then GSPG reduction with moving Qsets (level3) gives a projected multiplicative factor 1.6-2.3. We have done performance comparison between the numerically implemented levels 1, 2. While level1 is absolutely necessary for the convergence, level2 brings the acceleration factor 2.2-2.6 for the solution procedure, with a little overhead for GSPG pre-filtering.

The possible extensions of the method include a power law and generic monotone formula for pipes, iterative schemes for Nikuradze and Hofer formulae, rapid inverse reconstruction of data in reduced subgraphs, Qset movement algorithm for deeper reduction and the extension of the reduction methods to other elements using 2D tabulation (pixmaps). The implementation of these extensions is on the way. The question of the optimality of the proposed reductions will also be studied.

VI. Acknowledgment

The work has been supported by the German Federal Ministry for Economic Affairs and Energy, project BMWI-0324019A, MathEnergy: Mathematical Key Technologies for Evolving Energy Grids and by the German Bundesland North Rhine-Westphalia using fundings from the European Regional Development Fund, grant Nr. EFRE-0800063, project ES-FLEX-INFRA. The authors thank Kira Konich and Kevin Reinartz for proofreading the paper.

References

[1] J. Mischner, H.G. Fasold, and K. Kadner, System-planning basics of gas supply, Oldenbourg Industrieverlag GmbH, 2011 (in German).

[2] M. Schmidt, M. C. Steinbach, and B. M. Willert, “High detail stationary optimization models for gas networks”, Optimization and Engineering, vol. 16, 2015, pp. 131-164.

[3] T. Clees, “Parameter studies for energy networks with examples from gas transport”, Springer Proceedings in Mathematics & Statistics, vol. 153, 2016, pp. 29-54.

[4] T. Clees, I. Nikitin, and L. Nikitina, “Making Network Solvers Globally Convergent”, Advances in Intelligent Systems and Computing, vol. 676, 2017, pp. 140-153.

[5] A. Baldin et al., “Universal Translation Algorithm for Formulation of Transport Network Problems”, in Proc. SIMULTECH 2018, vol. 1, pp. 315-322.

[6] T. Clees, I. Nikitin, L. Nikitina, and Ł. Segiet, “Modeling of Gas Compressors and Hierarchical Reduction for Globally Convergent Stationary Network Solvers”, Int. J. On Advances in Systems and Measurements, IARIA, vol. 11, 2018, pp. 61-71.

[7] D. Eppstein, “Parallel recognition of series-parallel graphs”, Information and Computation, vol. 98, 1992, pp. 41-55.

[8] N. M. Korneyenko, “Combinatorial algorithms on a class of graphs”, Discrete Applied Mathematics, vol. 54, 1994, pp. 215-217.

[9] R. Z. Rios-Mercado, S. Wu, L. R. Scott, and E. A. Boyd, “A Reduction Technique for Natural Gas Transmission Network Optimization Problems”, Annals of Operations Research, vol. 117, 2002, pp. 217-234.

[10] T. Clees et al., “MYNTS: Multi-phYsics NeTwork Simulator”, In Proc. SIMULTECH 2016, SCITEPRESS, pp. 179-186.

[11] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming”, Mathematical Programming, vol. 106, 2006, pp. 25-57.

[12] Mathematica, Reference Manual, http://reference.wolfram.com [retrieved: June, 2019].

[13] MATLAB, https://de.mathworks.com/products/matlab.html [retrieved: June, 2019].

[14] T. Clees, I. Nikitin, L. Nikitina, and S. Pott, “Quasi-Monte Carlo and RBF Metamodeling for Quantile Estimation in River Bed Morphodynamics”, Advances in Intelligent Systems and Computing, vol. 319, 2014, pp. 211-222.

 ISBN: 978-1-61208-732-0