Topological Reduction of Stationary Network Problems: Example of Gas Transport

Anton Baldin[(1)], Tanja Clees[(1,2)], Bernhard Klaassen[(1)],
Igor Nikitin[(1)], Lialia Nikitina[(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 general method of topological reduction for the network problems is presented on example of gas transport networks. The method is based on a contraction of series, parallel and tree-like subgraphs for the element equations of quadratic, power law and general monotone dependencies. The method allows to reduce significantly the complexity of the graph and to accelerate the solution procedure for stationary network problems. The method has been tested on a large set of realistic network scenarios. Possible extensions of the method have been described, including triangulated element equations, continuation of the equations at infinity, providing uniqueness of solution, a choice of Newtonian stabilizer for nearly degenerated systems. The method is applicable for various sectors in the field of energetics, including gas networks, water networks, electric networks, as well as for coupling of different sectors.**

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

I. Introduction

This work is an extension of our conference paper [1], where the method of topological reduction for stationary gas transport network problems has been introduced. We have performed additional testing of the method on a large number of networks, evaluated various statistical characteristics, improved main algorithms for representing the equations and for their solution, considered the extension of the method for the networks of general type, as well as their coupling.

The physical modeling of gas transport networks is comprehensively described in works [2]–[4]. The element equations for pipes vary from the simplest quadratic form to more complex formulae by Nikuradze, Hofer and Colebrook-White. In our papers [5][6], 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 [7] we have constructed a universal translation algorithm, capable of formulating network problems for non-linear solvers with arbitrary problem description language. In paper [8], we presented theoretical foundations of topological reduction methods for generic stationary network problems.

In this paper, we continue the development of general topological reduction methods, applied to gas transport networks as an example. Our motivation is to accelerate solution procedure for stationary 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) [9]. 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) [10]. Such elementary operations are shown in Figure 1. In paper [8], 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 gas transport networks.

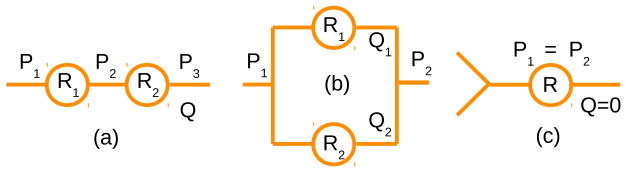

Figure 1. Main operations in GSPG reduction: series (a), parallel (b) connections to be reduced, contraction of a leaf (c).

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 [11], which is also based on graph theory but using a different approach. In Section IV, we also discuss possible extensions of our method. In Section V, we present further improvement of the main algorithms.

The described algorithms are implemented in the software MYNTS (Multi-phYsics NeTwork Simulator) [12], 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 [2][11]:

$$P_{in}|P_{in}| - P_{out}|P_{out}| = RQ|Q|, \qquad (1)$$

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:

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

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 [6] 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 (and the refinements, done here in Section V), 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 [6], 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 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):

$$\begin{aligned} P_1|P_1| - P_2|P_2| &= R_1 Q|Q|, \\ P_2|P_2| - P_3|P_3| &= R_2 Q|Q|. \end{aligned} \tag{3}$$

From here, we add the 2 formulas to get:

$$P_1|P_1| - P_3|P_3| = R^s_{12} Q|Q|, \;\; R^s_{12} = R_1 + R_2. \tag{4}$$

The inverse reconstruction of the eliminated variable P_2 is:

$$P_2|P_2| = P_1|P_1| - R_1 Q|Q|. \tag{5}$$

The parallel connection is (see Figure 1b):

$$\begin{aligned} P_1|P_1| - P_2|P_2| &= R_1 Q_1|Q_1| = R_2 Q_2|Q_2|, \\ Q &= Q_1 + Q_2. \end{aligned} \tag{6}$$

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

$$\begin{aligned} P_1|P_1| - P_2|P_2| &= R^p_{12} Q|Q|, \\ R^p_{12} &= \left(R_1^{-1/2} + R_2^{-1/2} \right)^{-2}. \end{aligned} \tag{7}$$

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

$$\begin{aligned} Q_1 &= Q/((R_1/R_2)^{1/2} + 1), \\ Q_2 &= Q/((R_2/R_1)^{1/2} + 1). \end{aligned} \tag{8}$$

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:

$$P_2|P_2| = P_1|P_1| - R Q_{set}|Q_{set}|. \tag{9}$$

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 [8] 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.

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).

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

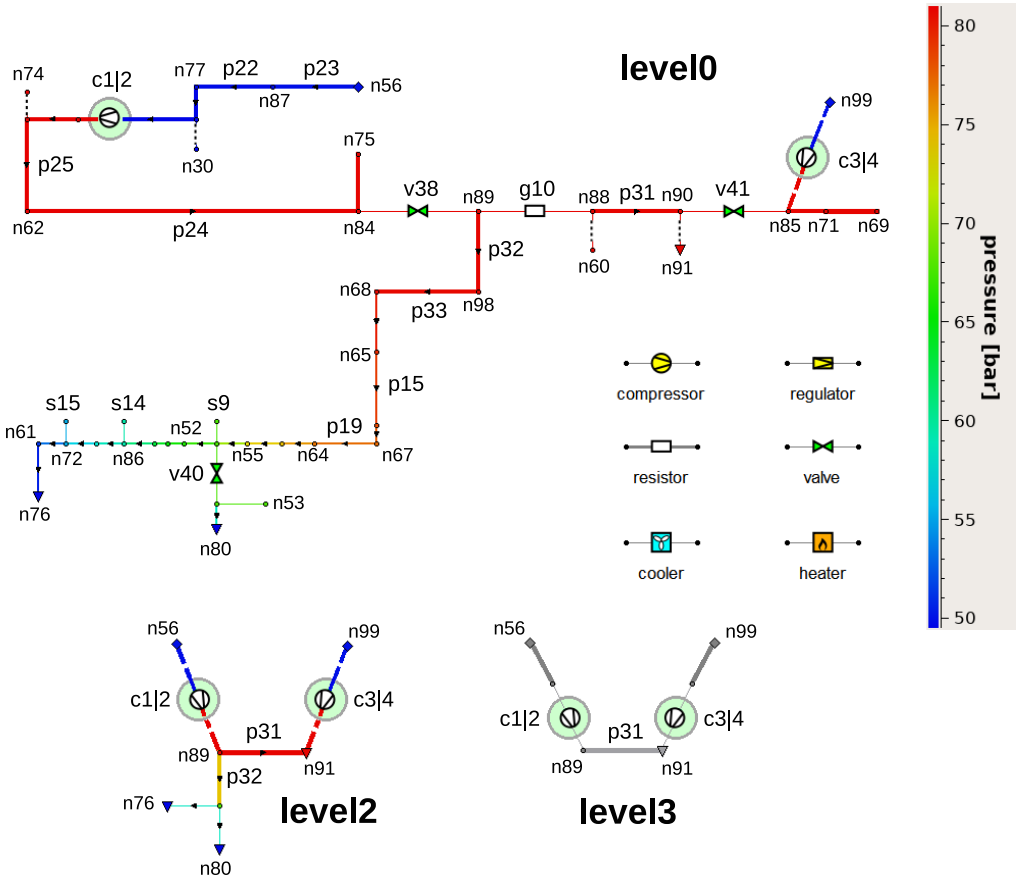

Figure 2. Realistic gas transport network N1 at different reduction levels: level0 = original network; level2 = GSPG reduction with fixed Qsets; level3 = GSPG reduction with moving Qsets. (Not shown: level1 = removing disconnected parts and superconductive elements, looking similar to level0.)

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

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

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.

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 [7]. 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) [13], Mathematica [14], MATLAB (MATrix LABoratory) [15] and a Newton solver, developed in our group. The best results for our type of problems have been obtained with IPOPT and Newton,

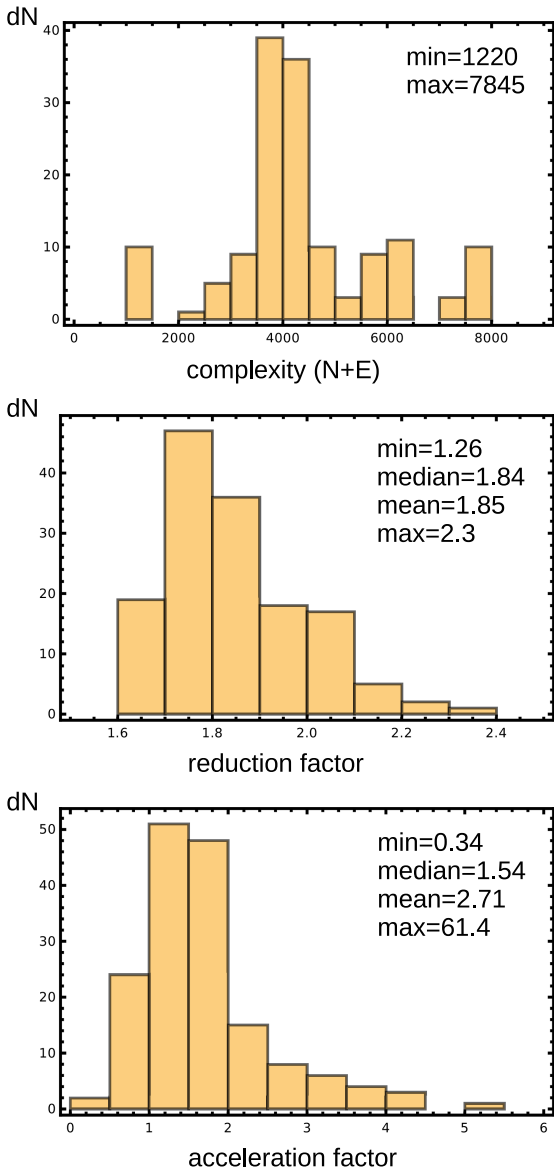

Figure 3. Topological reduction characteristics for additional 146 gas transport networks.

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 [7]. Although global convergence from an arbitrary starting point for stationary network problems is guaranteed theoretically [6], 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* [8]. The timing in Table III presents the sum over 3 phases.

More testing: in this paper, we performed the measurement of topological reduction characteristics for additional 146 gas transport networks of different complexity. The results are shown on Figure 3. The upper histogram shows the distribution of complexity value $N + E$ at the reduction level 1. This complexity value is more appropriate than the nominal one at level 0, since it counts only the active parts of the networks and only non-trivial (resistive) elements. The complexity values between approximately 1K and 8K elements are populated, with the maximum at 4K. The reduction factor between level 1 and level 2 is shown in the middle image. The reduction varies between 1.26 and 2.3, with median value 1.84. The shape of the histogram resembles Poisson distribution.

The acceleration factor is shown on the bottom image. It varies between 0.34 (deceleration) and 61.4 (strong acceleration), with median 1.54 and mean 2.71. The distribution is also looking like Poisson one, with outliers. The reason for these outliers is related with a randomness of the solver path towards the solution. Even with the regularization, the stationary problem formulation $f(x) = 0$ contains a complicated landscape of the function f, including sharp hills and narrow valleys, which lead to a slowdown of the solution procedure. The location of such features is random, so that topologically reduced problem can stuck in such regions, while the non-reduced problem was solved smoothly, leading to deceleration. Vice versa, the topological reduction can help to avoid these problematic regions, bringing very strong acceleration factor. Both cases are visible on the histograms as outliers.

Our statistics shows that deceleration events happen at a small probability and acceleration of solution prevails. On the other hand, strongly accelerated outliers skew the distribution, so that mean differs from the median significantly. While the median presents a statistical center of the distribution, mean value can be important in practically relevant setup, when a large number of the network cases is solved as a whole. Such applications appear, for example, in ensemble simulations, when stability of solution, sensitivity to variation of parameters or other statistically relevant characteristics are evaluated. The total time of ensemble simulation sums the individual time of each solution, making the mean time an important characteristic. Note that not the mean of the ratios between the original t_1 and reduced topology t_2 timings should be taken, but the ratio of corresponding means. In our testing cases, however, these characteristics are similar: $\langle t_1/t_2 \rangle = 2.71$, while $\langle t_1 \rangle / \langle t_2 \rangle = 2.69$.

IV. Possible extensions

At first, we perform a comparison with paper [11], where a different approach for topological reduction was taken. Then, we describe possible generalizations of our topological reduction algorithm.

a) Comparison with paper [11]: 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.

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.

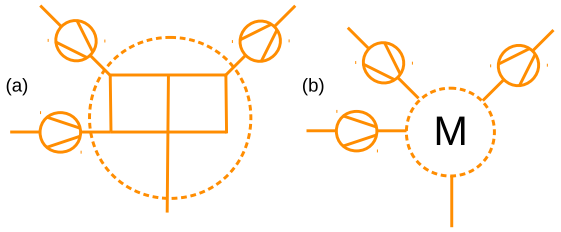

Figure 4. Shrinking a subgraph (a) creates a generalized network element with a fixed number of pins (b), a multipin (M).

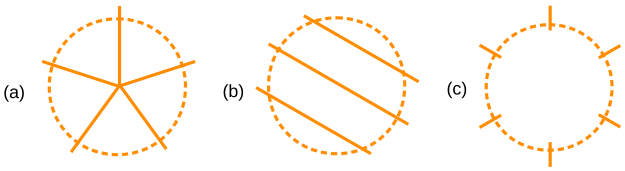

Figure 5. Particular examples: (a) 5-pin star; (b) 6-pin with 3 parallel connections; (c) empty 6-pin. In all cases, the number of equations describing the multipin is equal to the number of pins.

Another problem is that, according to [11], 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 4. 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 [11] 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 [8]). 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 5a, 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 5b 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 5c, 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 [11]. 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).

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 [6]. 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 [16]. 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 [6], 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 [3][4]. 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 also be 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 6. 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:

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

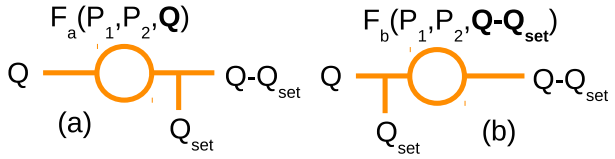

Figure 6. Q_{set} movement algorithm.

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, which 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 [8], 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. Improved algorithms

g) Triangulation: yet another possibility [8] is the representation of element equation as triangulated surface in the space of main variables (P_1, P_2, Q), see Figure 7. Such a surface defines a continuous piecewise linear function, e.g., $P_2(P_1, Q)$, whose monotony conditions are related with the direction of normal at each triangle. Namely, the normal n should be directed in octant, corresponding to $(+, -, -)$ signature. This property can be exhaustively checked for all triangles. Practically, the marginal signatures, e.g., $(+, -, 0)$,

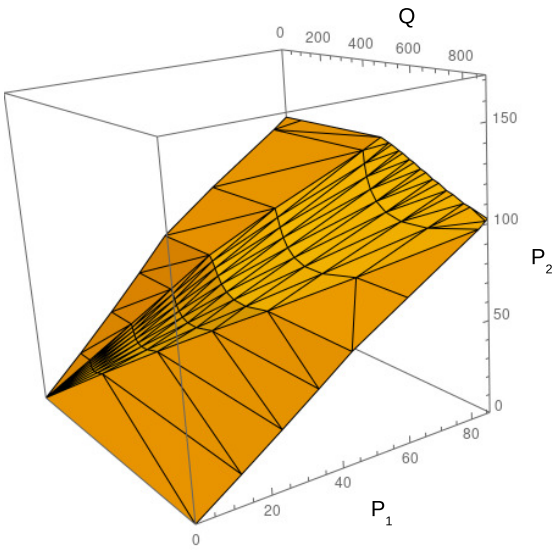

Figure 7. An example of triangulated representation $P_2(P_1, Q)$.

as for some triangles in the figure, are also allowed, if one adds a linear function $\epsilon(P_1 - P_2 - Q)$, with small positive regularization parameter ϵ, to the element equation.

When specified in a box on (P_1, Q) plane, the element equation can be continued outside of the box by the formula [6]:

$$\begin{aligned}
&\tilde{P}_2(P_1, Q) = P_2(\hat{P}_1, \hat{Q}) \quad (20)\\
&+k_P(\min(P_1 - P_{min}, 0) + \max(P_1 - P_{max}, 0))\\
&+k_Q(\min(Q - Q_{min}, 0) + \max(Q - Q_{max}, 0)),\\
&\hat{P}_1 = \min(\max(P_1, P_{min}), P_{max}),\\
&\hat{Q} = \min(\max(Q, Q_{min}), Q_{max}),
\end{aligned}$$

with constants $k_P > 0$, $k_Q < 0$. It is a particular case of more general formula:

$$\begin{aligned}
&\tilde{F}(x_1, ..., x_n) = F(\hat{x}_1, ..., \hat{x}_n) \quad (21)\\
&\quad + \sum_{i=1}^{n} k_i(\min(x_i - a_i, 0) + \max(x_i - b_i, 0)),\\
&\hat{x}_i = \min(\max(x_i, a_i), b_i)
\end{aligned}$$

for continuation of the function defined in a box in R^n, monotone with respect to each argument, to the outside, preserving the signature $\mathrm{sgn}(k_1, ..., k_n) = \mathrm{sgn}(\nabla F)$.

In this way, we obtain a piecewise linear representation of the function, fulfilling all necessary monotony conditions in the whole R^2 domain. As a result, the system satisfies the conditions [17] for the existence of solution of a system $f(x) = c$ for piecewise-linear function f and arbitrary c. This solution can be found using Katzenelson algorithm [18][17][5], converging to the solution in a finite number of steps. As an alternative, one can use a standard Armijo rule [19], applicable for smooth functions, which for the considered piecewise linear functions also converges to the solution in a finite number of steps. The reason for this is that the both algorithms require a finite number of steps to find a triangle where the solution is located, then, due to the linearity of the functions, converge to the solution in one iteration.

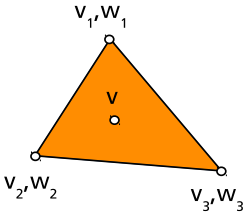

Figure 8. Illustration to barycentric coordinates.

Considering this representation in the context of topological reduction, we see that the necessary lookup operation $P_2^{(2)}(P_2^{(1)}(P_1,Q),Q)$ for serial connection is straightforward. For parallel connection, consider the formula (20), representing the element equation as continuous piecewise linear function on the whole plane R^2. It can also be reprojected to the other axes, e.g., $Q(P_1,P_2)$. Then we make a new triangulation with the required resolution on a square (P_1,P_2), and evaluate Q in the vertices of the triangles. Repeating the same operations with the parallel element, sum Q_1+Q_2 in the vertices of the shared triangulation and continue outside of the square by the $Q(P_1,P_2)$-analog of the formula (20). It can be then reprojected back to $P_2(P_1,Q)$-representation if needed. The operation count for the reduction, valid for both serial and parallel connections, is $O(N_{tri})$ multiplied to the cost of one evaluation of the triangulated function representation. This cost varies between $O(N_{tri})$ if direct search of triangle is implemented in the function evaluation and $O(1)$ for indexed search.

h) Low level implementation: construction of triangulated surface starts with a subdivision, done either with experimentally measured points or with continuous models, sampled with the necessary precision. This is done once in a preprocessing stage and the list of triangles is stored. Further, during the solution stage, for a given point $(v^x,v^y)=(x,y)=(P_1,Q)$ the triangle is found, containing this point. The search is done either directly, or with creating an auxiliary indexing structure during the preprocessing. This structure can be, e.g., a rectangular grid, assigning a sublist of triangles to a particular cell. After the triangle is found, its representation in barycentric coordinates is used, see Figure 8. Namely, the weights w_i are introduced at the vertices v_i of the triangle, $i=1..3$, such that

$$\sum_i w_i=1,\ v^k=\sum_i w_i v_i^k,\ k=x,y. \tag{22}$$

The same weights are used for linear interpolation of the function $v^z=z=P_2$ we are looking for:

$$z=\sum_i w_i z_i. \tag{23}$$

Solving (22) for w_i and substituting it to (23), obtain:

$$\begin{aligned}
&w_i=l_i^0+l_i^x x+l_i^y y,\ i=1..3, \qquad (24)\\
&z=l_4^0+l_4^x x+l_4^y y,\\
&l_i^k=n_i^k/d,\ k=0,x,y,\ i=1..4,\\
&n=\{\{x_3y_2-x_2y_3,-y_2+y_3,x_2-x_3\},\\
&\{-x_3y_1+x_1y_3,y_1-y_3,-x_1+x_3\},\\
&\{x_2y_1-x_1y_2,-y_1+y_2,x_1-x_2\},\\
&\{x_3y_2z_1-x_2y_3z_1-x_3y_1z_2\\
&+x_1y_3z_2+x_2y_1z_3-x_1y_2z_3,\\
&y_3(z_1-z_2)+y_1(z_2-z_3)+y_2(-z_1+z_3),\\
&x_3(-z_1+z_2)+x_2(z_1-z_3)+x_1(-z_2+z_3)\}\},\\
&d=x_3(-y_1+y_2)+x_2(y_1-y_3)+x_1(-y_2+y_3).
\end{aligned}$$

The necessary and sufficient condition for the point to belong to the triangle is that all $w_i\geq 0$. Note also that the function z in the triangle is linear with respect to (x,y), and its gradient is given by

$$(\partial z/\partial x,\partial z/\partial y)=(l_4^x,l_4^y). \tag{25}$$

i) Existence and uniqueness of solution: mathematically, the difference between the continuous piecewise linear and smooth modeling in our class of problems is that the mapping used in the equation $f(x)=c$ is either invertible continuous (homeomorphism) or invertible differentiable, for both forward and backward mappings (diffeomorphism). Any smooth mapping can be approximated by the piecewise linear one. For the considered class of element equations, any piecewise linear representation can be approximated by a smooth surface with the same signature of the normal. For this purpose, one should simply smooth the corners for the polyhedrons in 3 dimensions, representing our element equations. In paper [6] we have proven that if the element equations in the form $F(P_1,P_2,Q)=0$ everywhere possess a signature of the gradient $\nabla F=(+,-,-)$ and all connected components of the network have P_{set}-entries, then the Jacobi matrix $J=\partial f/\partial x$ of the system $f(x)=c$, composed of element and Kirchhoff equations, is globally non-degenerate. According to [19], the boundness $||J^{-1}||<C$ provides convergence of Newton method with Armijo line search rule, being started from an arbitrary point, therefore, the solution of the system $f(x)=c$ always exists. Also, non-degeneracy of the Jacobian means that the mapping $f(x)$ does not have folds or other singularities, as a result, the system $f(x)=c$ at all c possesses the same number of solutions, preimages of c under mapping f. This does not mean yet that there is a unique solution/preimage. Examples of non-trivial topology can be constructed, possessing everywhere non-degenerate Jacobian and having two or more preimages, e.g., mapping of torus to itself with $n>1$ winding numbers. Such examples can also be constructed for continuous piecewise linear mappings. To obtain a unique solution in the considered R^n case, we need to specify the behavior of the mapping at infinity. Below, we will show that in the considered problem, the element equation can be continued to infinity as a linear function with the necessary signature. As a result, the mapping $f(x)$ at infinity will be linear with non-degenerate Jacobian. Such a mapping with necessity has a single preimage, then due to the absence of singularities, there will be a single preimage of the mapping $f(x)$ everywhere. Thus, we prove that the mapping $f(x)$ is true diffeomorphism and the system $f(x)=c$ for the considered class of problems possesses *a unique solution*.

Lemma: the element equation $F(P_1,P_2,Q)=0$ with smooth F of signature $\nabla F=(+,-,-)$ in a box can be continued to infinity as a linear function, keeping the same signature everywhere.

Proof: at first, consider the continuation (21) of the function F, a linear function $F_0=\sum_i k_i x_i$ and linear interpolation between them: $F_1=(1-w)F+wF_0$, where $w(x)\in[0,1]$

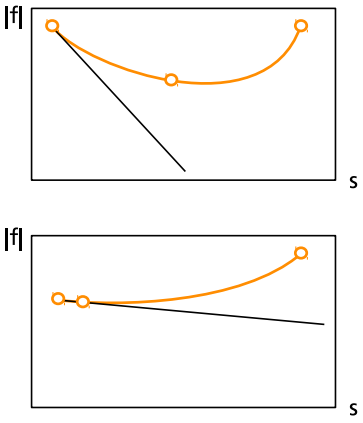

Figure 9. Illustration to Armijo rule. At the top: normal situation, at the bottom: stagnation of the algorithm.

is the weight function. Consider two spheres $|x| = R_{1,2}$, the first one contains the box of original F definition, the second one is of a greater radius $R_2 > R_1$. The function $w(x)$ is varied smoothly from 0 inside R_1-sphere to 1 outside R_2-sphere. Computing the gradient:

$$\nabla F_1 = (1-w)\nabla\tilde{F} + w\nabla F_0 + \nabla w(F_0 - \tilde{F}). \qquad (26)$$

The sum of the first two terms has correct signature automatically. For the third term, $|F_0 - \tilde{F}| < C$ is bounded, due to the equal slopes k_i in the linear function F_0 and the continuation $\tilde{F}$. By choosing $R_2 \gg R_1$, the function $w(x)$ can be selected, so that $|\nabla w|$ will be arbitrarily small. As a result, the third term can be made arbitrarily small, so that it will not change the signature of the gradient, defined by the sum of the first two terms. ■

Note that if the function F everywhere satisfies monotony conditions, it becomes unimportant, where $[a_i, b_i]$ limits of the box are set and how R_1 and R_2 spheres are selected. For every choice, the uniqueness of solution becomes proven. Therefore, the continuation beyond the box and interpolation between R_1 and R_2 spheres can be considered as auxiliary elements of the construction, provided for the purpose of proof only.

In more general terms, in differential topology, a degree of mapping $f(x) = c$ is defined as a sum $\deg f = \sum \operatorname{sgn} \det J$, taken over all preimages of a point c. This characteristics does not depend on the choice of a point c (assuming that the point is regular, $\det J \neq 0$). Actually, we have proven that at infinity $\deg f$ is equal $+1$ or -1, dependently on the sign of Jacobian at the linear continuation of the mapping. Then, it is the same everywhere, and since the Jacobian always has the same sign, the number of preimages is always 1, the system has a unique solution.

j) A choice of Newtonian stabilizer: as already mentioned, a combination of Newton method with Armijo line search rule [19] has guaranteed convergence in our problem class. A similar proof is also available for Katzenelson algorithm [17]. Armijo rule serves as a stabilizer to Newton method, which does not allow to perform too large steps, accepting only the steps which reduce the norm of the system residual. Katzenelson algorithm does the same for piecewise linear systems, testing whether the solution is located in a current piece, and if not, going along the Newton direction to the adjacent piece. Currently, we consider the mixed system of linear, piecewise linear and non-linear equations and prefer more generic Armijo rule. In certain cases, this leads to the following problem. Some of our elements (compressors, regulators in gas transport networks) possess degenerate equations, with zero signature instead of the prescribed ± 1. Such equations are regularized, generally, adding a small linear function with a definite signature. However, numerically these systems are almost degenerate. In particular, we observe a stagnation of iterations when the problematic region is approached. The reason is that Armijo rule attempts to reduce the function, which is almost constant in such regions, performing *a plateau optimization*, see Figure 9. Consider normalized Newton direction $n = dx/|dx|$, $dx = -J^{-1}f$. A variation of the norm of residual, in linear approximation, in that direction is $d|f| = f^T Jns/|f|$, where s is the distance along n. As a result, $d|f| = -|f|s/|dx|$. Approaching the degenerate region, in general position, $|dx| \to \infty$, thus, $d|f| \to -0$, in linear approximation the norm of residual along Newton direction is almost constant. This leads to stagnation of iterations. Indeed, $\Delta|f| = c_1 s + c_2 s^2 + ...$, a minimum of this function of s is located at $s^* = -c_1/(2c_2)$, at $c_1 < 0$ and $c_2 > 0$. Since $c_1 \to -0$, any non-zero quadratic term will lead to the vanishing step $s^* \to 0$. Practically, being stopped at the given number of iterations, the stagnation of the algorithm is equivalent to its divergence.

To overcome this issue, we propose a relaxed version of Armijo algorithm, which recognizes degenerate situation and allows for larger steps in the problematic region.

Algorithm (relaxed Armijo rule):

```
repeat until convergence:
  do Newtonian step dx
  set λ = 1
  trial point: x_t = x + λdx
  do λ = λ/2                                  (* bisection *)
    if |f(x)|/|dx| < ξ_0
    then cond = (|f(x_t)| < |f(x)| + df)
    else cond = (|f(x_t)| < (1 − αλ)|f(x)|)
  until cond                                  (* exit condition *)
  x = x_t                                     (* trial point accepted *)
```

The algorithm detects plateau situation by estimating of the lowest SVD eigenvalue (Singular Value Decomposition):

$$Jdx = -f, \; n = dx/|dx|, \qquad (27)$$
$$\xi^2_{min} \leq \xi^2 = n^T(J^T J)n = |f|^2/|dx|^2.$$

It has two parameters: ξ_0 is the upper threshold for the estimated lowest eigenvalue, df – the allowed increase of the residual in plateau situation. The exit condition is therefore modified: for $\xi < \xi_0$ a slight increase of the norm of the residual is allowed, not more then by df. Otherwise, a standard sufficient decrease rule is applied.

Our tests show that this modification of the algorithm drastically improves the convergence rate, for 60 tested networks the old algorithm brings 12 scenarios to divergence, while the

new relaxed version diverges only in 1 case, where it causes the Newton iteration to cycle. Switching the relaxation off for this particular scenario helps to achieve 100% convergence.

In the considered almost degenerate cases none of the algorithms, standard or relaxed Armijo rule, provide the guaranteed convergence. Empirically, the relaxed version behaves better. A construction of stable algorithm for processing almost degenerate cases remains a challenge for further developments.

k) Coupling to other sectors: note that topological reduction is possible not only for gas transport networks, but also for other network problems with element equations with the power law Q^α or other from the considered forms above. A practically important problem is a coupling of transport networks from different energetic sectors, e.g., gas, water, electricity, etc. In our previous work [7] it was shown how the coupling of sectors can be done on the modeling level. The question of stability and global convergence of solution for multi-sectoral case has been insufficiently studied. However, we can already identify two cases where the solution procedure is globally convergent.

Case A: consider two sectors, both satisfying the conditions of global convergence from [6]. If two sectors are not coupled, their Jacobi matrix possesses block-diagonal structure. The coupling produces terms located in non-diagonal blocks. Jacobian of the whole system depends continuously on these terms and is non-degenerate when these terms are switched off. Therefore, when the coupling between the sectors is sufficiently weak, then the Jacobian remains non-degenerate. In practice, the usage of this property can encounter one obstacle. If the element equation has marginal signature of derivatives, the Jacobian is almost degenerate and the weak coupling can lead it to the complete degeneracy.

Case B: consider, for definiteness, the coupling of gas transport and water thermal sectors, see Figure 10. The modeling of water thermal sector has been described in [7]. In the simplest case, it is represented by Kirchhoff law for thermal energy and element equations of the form:

$$\sum_e c_v Q_e T_e = 0,\ T_{n1} + \epsilon T_{n2} = T_e,\ Q_e > 0. \qquad (28)$$

The temperature in the edge T_e is defined by the temperature T_{n1} in the node upstream, also a small reverse diffusive term is introduced for regularization. In the presence of the given T_{set} temperature per every connected component of the graph, the problem is linear and non-degenerate. Moreover, the problem belongs to the already described class [7] with (linear) Kirchhoff nodal equations and (linear) element equation with (marginally) correct signature. For the coupling of sectors, an element is introduced representing a combustion chamber and a heat exchanger. Their equation

$$\Delta E = \eta H_m Q_g = c_v Q_w (T_{n2} - T_{n1}), \qquad (29)$$

in fact, transforms the energy of combustion of gas $H_m Q_g$, with a certain efficiency coefficient η, to the thermal energy of water $c_v Q_w \Delta T$. In this case, the definition of flows in one sector, gas or water thermal, can be renormalized, for the purpose of proof only, so that the energy from one sector will flow directly to another sector. In this way, a combined problem will be formulated, with Kirchhoff law for the energy flow and all elements possessing correct signature.

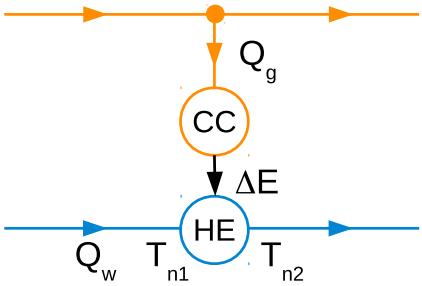

Figure 10. An example of coupling between gas and water thermal sectors. Gas (orange), water thermal (blue), combustion chamber (CC), heat exchanger (HE).

Therefore, for the combined problem the global convergence can be proven, without even using the fact that the water part of the system is linear. The generalization to the case of multiple contact points between sectors with different transfer coefficients is still to be studied. An interesting property of the considered formulation of multi-sectoral problems in the field of energetics is that all flows participating in Kirchhoff equations are energetic and universal, while the other type of flows, e.g., mass flow, molar flow, volume flow, etc., are related to the energetic ones by proportionality coefficients, depending on the particular sector.

VI. Conclusion

In this paper, the general method of topological reduction for the network problems has been presented, using gas transport networks as an example. The method uses a contraction of series, parallel and tree-like subgraphs, containing the edge elements, described by quadratic, power law or general monotone dependence. This way, we achieve the goal of significant lossless reduction of the graphs and we accelerate solution procedure correspondingly. A large set of realistic network examples of different complexity have been used for the benchmarking of the method. The statistical distribution for the reduction and the acceleration factors has been evaluated, together with corresponding minimum, maximum, median and mean values. 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 1.26-2.9, 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 varied from 0.34 (deceleration) to 61.4 (strong acceleration), with median 1.54 and mean 2.71, for the solution procedure, with a little overhead for GSPG pre-filtering.

The possible extensions of the method include 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). Additional extensions have been proposed, including triangulated element equations, continuation of the equations at infinity, providing uniqueness of solution, a choice of Newtonian stabilizer for nearly degenerated systems. The generalization

of the method for various sectors in the field of energetics has been proposed, including gas networks, water networks, electric networks, as well as for coupling of different sectors.

VII. Acknowledgment

We are grateful to the organizers and participants of INFOCOMP 2019 conference for fruitful discussions. 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.

References

[1] A. Baldin et al., "Topological Reduction of Gas Transport Networks", in Proc. of INFOCOMP 2019, IARIA, 2019, pp. 15-20.

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

[3] 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.

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

[5] T. Clees, N. Hornung, I. Nikitin, and L. Nikitina, "A Globally Convergent Method for Generalized Resistive Systems and its Application to Stationary Problems in Gas Transport Networks", In Proc. SIMULTECH 2016, SCITEPRESS, 2016, pp. 64-70.

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

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

[8] 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.

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

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

[11] 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.

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

[13] 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.

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

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

[16] 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.

[17] M. J. Chien and E. S. Kuh, "Solving piecewise-linear equations for resistive networks", International Journal of Circuit Theory and Applications, vol. 4, 1976, pp. 1-24.

[18] J. Katzenelson, "An algorithm for solving nonlinear resistor networks", Bell System Technical Journal, vol. 44, 1965, pp. 1605-1620.

[19] C. T. Kelley, Iterative Methods for Linear and Nonlinear Equations, SIAM, Philadelphia, 1995.