Model order reduction for gas and energy networks

Christian Himpe1*, Sara Grundel1 and Peter Benner1

*Correspondence: himpe@mpi-magdeburg.mpg.de
1Computational Methods in Systems and Control Theory Group at the Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, D-39106 Magdeburg, Germany

Abstract

To counter the volatile nature of renewable energy sources, gas networks take a vital role. But, to ensure fulfillment of contracts under these circumstances, a vast number of possible scenarios, incorporating uncertain supply and demand, has to be simulated ahead of time. This many-query gas network simulation task can be accelerated by model reduction, yet, large-scale, nonlinear, parametric, hyperbolic partial differential(-algebraic) equation systems, modeling natural gas transport, are a challenging application for model order reduction algorithms.

For this industrial application, we bring together the scientific computing topics of: mathematical modeling of gas transport networks, numerical simulation of hyperbolic partial differential equation, and parametric model reduction for nonlinear systems. This research resulted in the morgen (Model Order Reduction for Gas and Energy Networks) software platform, which enables modular testing of various combinations of models, solvers, and model reduction methods. In this work we present the theoretical background on systemic modeling and structured, data-driven, system-theoretic model reduction for gas networks, as well as the implementation of morgen and associated numerical experiments testing model reduction adapted to gas network models.

MSC: 93-04; 76N15; 93B20

Keywords: Digital twin; Gas network; Model reduction; Empirical Gramians; Hyperbolic systems

1 Introduction

Rapid transient simulations of gas flow in pipeline networks are essential for safe operations of gas networks as well as reliable delivery of denominations. Yet, in a volatile supply and demand environment, due to increasing renewable energy sources, the time horizon for planning dispatch and load forecasting shortens while more sources of accountable uncertainties, such as effects of weather on energy consumption and production are introduced; to a lesser degree this is a long standing challenge [6]. An example is the interconnection of gas and power grids through gas-fired power plants [27, 91, 144]. Thus, more simulations for the uncertainty quantification of dynamic gas network behavior need to be completed in less time by the gas grid operators. However, available compute power
is (and was [138]) never sufficient. To this end we evaluate customized model reduction techniques for an established class of gas network models.

This work and the associated software platform are an effort to determine which model reduction methods are suitable for enabling digital twins [58, 87, 121] of gas networks. Depending on the mathematical model and quantities of interest, the twin may contain redundant or superfluous information with respect to the simulations. Therefore, model reduction compresses the twin to a matched surrogate model, which is sufficiently accurate in the chosen operating region.

The swift numerical simulation of gas network twins by reduced order modeling is highly relevant, not only due to the transition towards renewables at the time of writing, which is underlined by the research projects MathEnergy\(^1\) (Mathematical Key Technologies for Evolving Energy Grids) [29] that the authors are part of, and TRR154\(^2\) (Mathematical Modelling, Simulation and Optimization using the Example of Gas Networks) [85], but also because of the intriguing numerical problem of model reduction for hyperbolic, nonlinear, coupled, parametric, multiscale partial differential-algebraic equation systems.

If relevant intraday demand changes occur, established steady / stationary / static simulations may not be sufficient anymore [42]. The basic model for the simulation of unsteady / dynamic / transient flow processes in gas pipelines is based on the one-dimensional (isothermal) Euler equations, originally introduced in [55], and popularized in [104] as well as in [80, 81] around the same time. A practical extension in the context of gas networks is the repetitive modeling approach [37], which enables a modular construction. For extensive details on gas network modeling, see the works [19, 33, 45, 94, 105, 123, 128, 131], and for a concise summary of the overall approach we recommend [7]. Furthermore, a system-theoretic approach to gas networks is discussed in [35, 54], and results on boundary reachability (controllability) and observability for this class of models have been derived in [8, 9].

In terms of complexity reduction for gas network models, earlier works applied techniques such as combining parallel pipelines [129], singular perturbation [127] and symbolic simplifications [95]. Younger works introduced projection-based model reduction methods from fluid dynamics, proper orthogonal decomposition (POD) [50, 51, 53], and system-theoretic methods [88] (matrix interpolation), [1, 89] (balanced truncation), or Padé-type approximations [39, 86] (moment matching).

In this work, we conceptually combine these previous approaches, by using system-theoretic but data-driven methods that are structure-preserving. The utilized data-driven assembly of the system-theoretic operators, central to the employed methods, is also a partial answer to the challenges posed in [57, Remark 5.10]; while structure preservation means in this context, retaining (particularly not mixing) the discretized physical variables in the reduced order model. Furthermore, we note that from this work’s point of view, [130, 144, 145] are concerned rather with (valuable) model simplifications than model reduction.

To avoid the analytically most complex aspects of the gas network model – the nonlinearities – one could linearize the model around an operating point. Yet, the different nonlinearities (i.e. friction, compressibility and compressors) are unlikely to have a compatible operating point for a wide range of scenarios. Furthermore, linearized and simplified

\(^1\)https://mathenergy.de
\(^2\)https://trr154.fau.de
models of gas flow have limitations with simulations of real scenarios [61], [134, Ch. 7]; hence, we use a nonlinear model. Since there is no general theory for model reduction of nonlinear systems, and a high degree of modularity in the gas network modeling process, model reduction algorithms have to be compared heuristically to determine their applicability. As a result of this reasoning and a demand for gas network simulation software tools [32, 69], a platform named morgen (model order reduction for gas and energy networks) was designed with the goal to compare different models, solvers and reductors. The morgen platform is an open-source project, and designed in a configurable, modular, and extensible manner, so that modeling, discretization or model reduction specialists can compose and compare their methods fairly.

In summary, this work contributes a full, but also fully modular, modeling, model reduction and simulation open-source software stack for gas networks, and potentially other energy network systems (i.e. district heating networks, water networks), which brings together research results from various disciplines.

Overall, this work is organized as follows: In Sect. 2 the gas network model, simplifications, non-pipe elements, a relation to port-Hamiltonian theory, and obtaining a steady-state initial condition are described. Section 3 and Sect. 4 outline the general model reduction idea and propose five reduction method classes. The design and features of the morgen platform are summarized in Sect. 5, followed by three sets of numerical experiments in Sect. 6. We conclude by an outlook (Sect. 7) and an evaluation of our findings in Sect. 8. A list of recurring symbols is found in Table 1.

### 2 The transient gas network model

The goal of this section is to describe the partial differential-algebraic equation model of a gas network as an input-output system that maps boundary values to quantities of interest. First, the model for a single pipeline is summarized, which is based on the isothermal Euler equations of gas dynamics [104, 106]. Then, it is generalized to a network of pipes, and simplified compressors are added. Additionally, a connection to energy-based modeling is made.

Even though further non-pipe elements are common in gas networks, such as resistors, coolers, heaters, valves and control valves [45, 96], we prioritized compressors to focus on the model reduction aspect on a macro scale. Moreover, the practical numerical problems of scale homogenization, spatial discretization, index reduction and steady-state approximation are discussed in this section.
2.1 The gas pipeline model

The principal building blocks of gas transport networks are pipelines or ducts. Since the length of pipes exceeds their diameter by far \((L > 500d, [81])\), a spatial one-dimensional model suffices. We model gas flow in a (cylindrical) pipe of length \(L\) connecting two junctions by the isothermal Euler equations:

\[
\frac{1}{\gamma_0 z_0} \frac{\partial_t p}{\partial t} = -\frac{1}{S} \frac{\partial_x q}{\partial x},
\]

\[
\frac{\partial_t q}{\partial x} = -S \frac{\partial_x p}{\partial x} - \left( \frac{S g \partial_x h}{\gamma_0 z_0} p + \frac{\gamma_0 z_0 \lambda_0}{2dS} |q| q \right),
\]

which determine the evolution of the coupled pressure \(p(x,t)\) and mass-flux \(q(x,t)\) variables. The physical dimension of the pipe enters as its diameter \(d\) and the derived cross-section area \(S = \pi \frac{d^2}{4}\), which is assumed constant, ignoring the influence of temperature and pressure on the pipe walls. These coupled partial differential equations (PDE) can also be characterized as a nonlinear, two-dimensional, first-order hyperbolic system of conservation laws: the pressure \(p\) preserves continuity, while the mass-flux \(q\) conserves momentum.

Following [64, 79, 108] and [19, Sect. 2.1], the inertia term has been neglected due to a low Mach number \(m \ll 1\), which leads to the ISO2 model in the TRR154 classification [33], also known as friction-dominated model [24, Sect. 3.2.1]. Furthermore, we assume a turbulent flow with a Reynolds number exceeding \(\text{Re} \gg 10^5\) [40, 55], neither line breaks or valve closings happen intraday (to preclude associated shocks [34]), and low-frequency boundary values [10, 108], which in this work are (but generally not limited to) the supply pressure and demand mass-flux, due to frequent use in literature, and use-cases like guaranteed demand pressures [22, 63].

In (1), the linear reaction term describes the effect of gravity (with standard gravity \(g \equiv 9.80665 \text{m/s}^2\)) due to the pipe height \(h(x)\), while the nonlinear reaction term models loss of momentum due to friction at the pipe walls, specified by the (Darcy-Weisbach) friction factor \(\lambda_0 := \lambda(d, k, \text{Re}_0)\), given a pipe roughness \(k\), and an estimated mean Reynolds number \(\text{Re}_0\), see [19, Sect. 2.2].\(^4\) This friction term is principal to the accuracy of the gas pipeline model [33, 61, 93, 107].

In this model variant, a (globally) constant mean compressibility factor \(z_0 := z(p_0, T_0) \in \mathbb{R}\) is assumed [35, 62, 106, 111], as well as a constant gas state \(\gamma_0 := R_s T_0\), whereas the temperature \(T_0\) and the specific gas constant \(R_s\) are treated as parameters (see Sect. 2.6). To this end, the steady-state pressure \(\bar{p} := p_0\) is used to compute \(z_0\), via heuristic formulas based on the Virial expansion [25], [19, Sect. 2.3].\(^5\)

2.2 Homogenizing scales

The SI-based units for pressure and mass-flux are [Pa] and [kg/s], respectively. This introduces a difference in scales of five orders of magnitude between the variables \(p\) and \(q\), and hence induces numerical problems. To counter this multiscale structure, we simply rescale the pressure from [Pa] to [bar] which conveniently comprises a factor of \(10^5\). Nonetheless,

\(^4\) Additionally to [19], the IG T formula [33], [96, Sect. 15.2.3] is implemented in morgen.

\(^5\) Additionally to [19], the DVGW-G-2000 equation [94, Ch. 9] is implemented in morgen.
the model still consists of two interacting physical variables, hence the model still has to be treated as a coupled system, however, without numerical multiscale issues.

### 2.3 The gas network model

Given the model for a single pipe from the previous section, a (gas) network of pipes can be encoded by a finite directed graph, which is a tuple \( G = (\mathcal{N}, \mathcal{E}) \) of finite sets symbolizing nodes \( \mathcal{N} \), and oriented edges \( \mathcal{E} \). The edges correspond to pipes, while the nodes represent the junctions connecting pipes. The connectivity of the network is the relationship between edges and junctions, given by the incidence matrix \( \mathcal{A} \in \{-1,0,1\}^{\left|\mathcal{N}\right| \times \left|\mathcal{E}\right|} \), a map from edges to nodes, such that:

\[
\mathcal{A}_{ij} = \begin{cases} 
-1 & \text{\( \mathcal{E}_j \) connects from \( \mathcal{N}_i \)}, \\
0 & \text{\( \mathcal{E}_j \) connects not \( \mathcal{N}_i \)}, \\
1 & \text{\( \mathcal{E}_j \) connects to \( \mathcal{N}_i \)}. 
\end{cases}
\]

Note, that the orientation of the edges is not enforcing the dynamic flow direction of the gas, but is necessary to determine the complexity and boundary of the overall networked model \[19, 53\].

We introduce the notation \( |\mathcal{A}| \) for the component-wise absolute value of a matrix. Using this absolute value, the following partial incidence matrices associating edges entering and leaving nodes respectively are defined similar to a Heaviside function:

\[
\mathcal{A}_R := \frac{1}{2}(\mathcal{A} + |\mathcal{A}|), \quad \mathcal{A}_L := \frac{1}{2}(\mathcal{A} - |\mathcal{A}|).
\]

Next, based on this connectivity, certain conservation properties are enforced to maintain a network balance, and thus ensure physical relevance of the gas network model. Specifically, the Kirchhoff laws are applied to the network in vectorized (or rather matricized) form \[18, 132\]:

1. The sum of in- and outflows (mass-flux) at every node (junction) is zero: This means that no gas gets lost in transport from one pipe to the next, with the exception of boundary nodes. Hence, a vector of flows \( q \in \mathbb{R}^{\left|\mathcal{E}\right|} \) applied to the incidence matrix equals the (out-)flow at the boundary (discharge) nodes \( d_q : \mathbb{R} \rightarrow \mathbb{R}^{\left|\mathcal{N}_D\right|} \), which are mapped into the network via \( \mathcal{B}_d \in \{0,1\}^{\left|\mathcal{N}\right| \times \left|\mathcal{N}_D\right|} \):

\[
\mathcal{A}\mathbf{q}(t) = \mathcal{B}_d \mathbf{d}_q(t),
\]

with \( \mathcal{N}_D \subset \mathcal{N} \) denoting the subset of boundary nodes, which only connect from one node respectively, but not to any node.

2. The sum of directed pressure drops in every fundamental loop is zero: Fortunately, an equivalent representation \[143, \text{Ch. 7.3}\] can be used, which resolves implicitly. It remains to ensure that the nodal pressures at the in-flow boundary (supply) nodes are associated to the boundary function \( s_p : \mathbb{R} \rightarrow \mathbb{R}^{\left|\mathcal{N}_S\right|} \), which are mapped to the network via \( \mathcal{B}_s \in \{0,1\}^{\left|\mathcal{N}_S\right| \times \left|\mathcal{E}\right|} \):

\[
\mathcal{A}_i^T p(t) + \mathcal{B}_s^T s_p(t) = \mathcal{A}_0^T p(t) - \mathcal{A}_L^T p(t),
\]
with \( N_S \subset N \) denoting the subset of boundary nodes, which only connect to one node respectively, but not from any node, and the reduced incidence matrix

\[
A_0 \in \{-1, 0, 1\}^{\vert N_0 \vert \times \vert E \vert}, \quad \vert N_0 \vert = \vert N \vert - \vert N_S \vert,
\]

with all rows associated to supply nodes removed.

Given a connected and directed graph representing a gas network topology, with the dynamic flow in the pipe edges modeled by the PDE (1), then yields a partial differential-algebraic equation (PDAE) due to the above constraints.

### 2.4 Discretization and index reduction

Next, we delineate the discretization of the spatial differential operators and reduction of the (P)DAE index in the networked system, yielding a system of ordinary differential equations (ODE). Eventually, the remaining discretization of the temporal differential operators is addressed.

We explicitly do not use the decoupling approaches from [12] or [11], as the former employs linearization and hence does not fit this setting, while compared to the latter, our equivalent analytic index reduction is more convenient here.

The partial differential-algebraic equation is discretized using the method of lines: First in space, then in time, yielding a (nonlinear) dynamic system. For the spatial discretization a first-order upwind finite difference scheme is utilized [10, 133]. We select (only) two points for each of the \( k \) pipes with length \( L_k \), namely the start (\( \cdot^S \)) and end point (\( \cdot^L \)):

\[
\begin{align*}
\partial_x p_k &\approx \frac{p_k^R - p_k^L}{L_k}, \quad k \in E, \\
\partial_x q_k &\approx \frac{q_k^R - q_k^L}{L_k}, \quad k \in E.
\end{align*}
\]

The matter of short, long and varying lengths \( L_k \) is addressed in Sect. 2.4.3.

For each pipe, this leads to the following equations:

\[
\begin{align*}
\frac{1}{\gamma_0 z_0} \partial_t p^*_k &= -\frac{1}{S_h} \frac{q_k^R - q_k^L}{L_k}, \\
\partial_t q^*_k &= -S_k \frac{p_k^R - p_k^L}{L_k} - \left( \frac{S_k g(h_k^R - h_k^L)}{\gamma_0 z_0 L_k} p_k^* + \frac{\gamma_0 z_0 \lambda_{0,k}}{2 d_{s,k}} q_k^* \right).
\end{align*}
\]

Now, different choices for \( \cdot^* \) are surmisable. Subsequently, two specific combinations of \( p^* \) and \( q^* \) will be discussed: The midpoint discretization [18, 50, 51, 53, 145], and the left-right discretization [48, 52, 115] resulting in (implicit) ODEs. For an error analysis of these two discretization variants, see [128]. In the following, we describe a unified approach of deriving these index-reducible discretizations.

For notational ease in the coming subsections, a vectorized form for the above (networked) system including its constraints is given by:

\[
\begin{align*}
\dot{d}_0 \partial_t p^* &= D_p (q^* - q^t), \\
\partial_t q^* &= D_q (A_0^T p + B_0^T q) - \left( D_q D_{0} d_0 p^* + D_{1} \frac{1}{d_0} \frac{q^*}{q^t} \right), \\
A_0 q^* &= B_d d_q,
\end{align*}
\]
using, thus resolving, the constraint $A_T^0 p + B_T^s s_p = p^R - p^L$, as well as $d_0 := \frac{1}{\gamma_{0z}} \in \mathbb{R}$ and the diagonal matrices:

$$D_{p,kk} := -\frac{1}{S_k L_k}, \quad D_{q,kk} := -\frac{S_k}{L_k}, \quad D_{g,kk} := g(h_k^R - h_k^L), \quad D_{f,kk} := \frac{\lambda_0}{2d_k S_k}.$$

Note, that $(d_0 \cdot p^*)$ corresponds to the global average density, $(S_k^{-1} \cdot q_k^*)$ to the local flow rate, and depending on the choices for $p^*$ and $q^*$, the model's analytic and numerical character will differ.

### 2.4.1 Midpoint discretization

In case of the midpoint discretization, we set $p^*_k$ and $q^*_k$ to the mean of its associated edge's endpoints:

$$p^*_k = \frac{p_k^R + p_k^L}{2} := p_k^+, \quad q^*_k = \frac{q_k^R + q_k^L}{2} := q_k^-.$$

Furthermore, we define $q^- := \frac{1}{2}(q_k^R - q_k^L)$, and note, that in vectorized form, $p^* = \frac{1}{2}(|A_0^1| p + |B_0^1| s_p)$.

Together with the algebraic constraints from Sect. 2.3, a DAE system in the variables $p$, $q^*$, and $q^-$ arises:

\begin{align*}
  d_0 \frac{1}{2}(|A_0^1| p + |B_0^1| s_p) &= D_p 2q^-, \quad (3a) \\
  \dot{q}^- &= D_q (A_0^1 p + B_0^1 s_p) - \left( D_q D_2 d_0 \frac{1}{2}(|A_0^1| p + |B_0^1| s_p) + D_f \frac{|q^*| |q^-|}{d_0 \frac{1}{2}(|A_0^1| p + |B_0^1| s_p)} \right), \quad (3b) \\
  0 &= A_0 q^- + |A_0| q^* - E_ddq. \quad (3c)
\end{align*}

Since we aim to obtain an ODE, we need to transform this DAE system. The complexity of deriving this transformation is quantified by the DAE's index. From the various DAE index concepts, we use the tractability index $\tau$ [92], for which the midpoint discretization guarantees $\tau \leq 2$ [53].

This DAE can be decoupled into an ODE by rewriting it in the variables $p$ and $q^*$. To this end,

1. the pressure boundary condition implicitly resolves (2).
2. By multiplying the differential equation (3a) by $(|A_0| D_p^{-1})$ from the left, the remaining algebraic constraint (3c) is removed by replacing $A_0 q^-$ by $(-A_0 q^* + E_ddq)$ in (3a).

Since $D_p$ is a diagonal matrix, this is also numerically feasible.
We also pre-multiply (3b) with the inverse of the diagonal matrix \( D_q \). Altogether, we obtain:

\[
\begin{align*}
|A_0|\left( \frac{1}{2} D_P^{-1} d_0 \right) |A_0| |\dot{p}| = -A_0 q^\dagger + B_d d_q - |A_0|\left( \frac{1}{2} D_P^{-1} d_0 \right) |B_q^T| \dot{s}_p, \\
D_q^{-1} \dot{q}^\dagger = A_{10}^T p + B_{20}^T s_p - \left( D_d d_0 \frac{1}{2} \left( |A_0| |p| + |B_q^T| \dot{s}_p \right) \right).
\end{align*}
\]  

(4a)  
(4b)

This system of a pressure and mass-flux variable now consists of only differential equations. Notably, the first equation of the ODE system contains a temporal derivative of the input function \( s_p \), which practically would need to be approximated numerically, for example by finite differences. However, we will assume that all inputs are sums of step functions, so that effectively \( \dot{s}_p \equiv 0 \), which is reasonable as we assume exclusively low-frequency boundary values.

2.4.2 Endpoint discretization

For the endpoint discretization, also called left-right discretization, we set \( p_k^* \) and \( q_k^* \) to the left and right endpoints, respectively:

\[
\begin{align*}
p_k^* &= p^R_k, \\
q_k^* &= q^L_k.
\end{align*}
\]

Since \( (B_q^T + |B_q^T|) s_p = 0 \), we can write \( p^R = A_{10,R}^T p \). With the algebraic constraints from Sect. 2.3, a DAE system in the variables \( p, q^R, \) and \( q^L \) results:

\[
\begin{align*}
d_0 A_{0,R} p &= D_p (q^R - q^L), \\
q^L &= D_q \left( A_{10}^T p + B_{20}^T s_p \right) - \left( D_d d_0 \frac{1}{2} \left( |A_0| |p| + |B_q^T| \dot{s}_p \right) \right), \\
0 &= A_{0,R} q^R + A_{0,L} q^L - B_d d_q.
\end{align*}
\]  

(5a)  
(5b)  
(5c)

As for the midpoint discretization, we want to derive a system of ODEs. For the endpoint discretization, it is shown in [52, 115], that the tractability index is \( \tau = 1 \), if all edges connecting supply nodes are directed from the supply, and each component of the graph is connected to at least one supply. This implies, no two supplies are to be directly connected. Similar to Sect. 2.4.1, this DAE can be decoupled into an ODE by rewriting it in the variables \( p \) and \( q^L \). Applying equivalent steps to (5a)–(5c) as for the midpoint decoupling (3a)–(3c) yields:

\[
\begin{align*}
(A_{0,R} D_P^{-1} d_0 A_{10,R}^T) \dot{p} &= -A_0 q^L + B_d d_q, \\
D_q^{-1} \dot{q}^L &= A_{10}^T p + B_{20}^T s_p - \left( D_d d_0 A_{0,R}^T p + D_q^{-1} D_q D_d d_0 \frac{1}{2} \left( |A_0| |p| + |B_q^T| \dot{s}_p \right) \right),
\end{align*}
\]  

(6a)  
(6b)

using \( A_{0,R} + A_{0,L} = A_0 \) in (6a).

An advantage of this endpoint discretization, in addition to the absence of a derivative of an input function, is the input-free friction term in (6b).
2.4.3 Temporal discretization

After spatial discretization and index reduction of the gas network model, a system of stiff nonlinear ODEs (in time) remains. The remaining temporal differential operator(s) are discretized using time-stepping schemes.

Due to the hyperbolicity of the pipeline model, the temporal resolution $\Delta t$ used for the discrete integration interdepends on the spatial resolution $\Delta x$ employed for the discretization of the spatial differential operators. Formally, this is expressed by the Courant-Friedrichs-Levy (CFL) condition $[13, 59, 140]$, which states that the propagation of information cannot be faster than its conveyor:

$$\lambda_{CFL} := \frac{v_{\text{max}} \Delta t}{\Delta x} < 1,$$

with the (dimensionless) CFL constant $\lambda_{CFL}$, symbolizing the ratio of temporal and spatial discretization step-width scaled by the peak gas velocity $v_{\text{max}}$. Since the flow is subsonic, $v_{\text{max}}$ could be estimated from the (linearized) characteristics $[59]$, or via the boundary values. However, we fix this maximum gas speed to $v_{\text{max}} = 20 \text{ m/s}$ (practically this is configurable in morgen).

Due to this relation of the space and time discretization and a pre-selected application-specific sampling frequency $\Delta t$ of the output trajectory (i.e., every 60 s), $\Delta x$ has to be adapted accordingly. The spatial discretization by finite differences in the previous sections ignores pipeline length, as each pipe is assigned only one (finite) difference. This means, pipes are potentially too long or too short with respect to a nominal length $\Delta x$, determined by the CFL condition $\Delta x = (1 - \varepsilon)v_{\text{max}} \Delta t$, $0 < \varepsilon \ll 1$. Thus, too long pipes are subdivided into virtual pipes of nominal length, while too short pipes, including a potential remainder of too long pipes, are “rounded” to a full nominal-length pipe, yet with a friction term scaled by the fraction of the short pipe’s length compared to the nominal length,

$$\tilde{D}_{f, kk} = \begin{cases} D_{f, kk} & L_k = \Delta x, \\ \frac{L_k}{\Delta x} D_{f, kk} & L_k < \Delta x. \end{cases}$$

This approach assumes that delays due to the forced virtual length of an actually short pipe are insignificant, hence this simple homogenization of pipe lengths may be improved by replacing short pipes with friction-less shortcuts and a static pressure-drop, as used in the quasi-static model $[50, 64]$ and similar to the subsequent compressor model in Sect. 2.5.

Overall, we refine each pipe into a sequence of pipes of a selected nominal length – a graph level refinement – which is determined using the CFL condition. We note here, that this methodology is aimed at ensuring a certain minimum length for each pipe, as the shortest pipe may dictate an unnecessarily finely resolved time discretization. In terms of a pipe’s maximum length, suggestions are for example: 5 km ($[50]$), or 10 km ($[144]$).

As the model is composed of a stiff, linear (hyperbolic) and a nonlinear component, an implicit solution of the linear part using a diagonally implicit Runge-Kutta (DIRK) method, and an explicit solution of the nonlinear part via a strong stability preserving

---

6 e.g. https://petrowiki.spe.org/Pipeline_design_consideration_and_standards#Gas_line_sizing
(SSP) method, by an IMEX (IMplicit-EXplicit) solver, as proposed in [13, Sect. 3.2.2], is targeted. The actual quadrature rules used to compute the transient solutions are detailed in Sect. 5.3.

2.5 Simplified compressors

Beyond pipes, gas networks comprise a variety of non-pipe elements, of which the most important are compressors. Compressors increase gas pressure to counteract cumulative effects of retarding forces (friction, gravity, inertia etc.), and are grouped into stations with many possible configurations. For our purposes, we just allow fixed configurations on a macro scale [90] per scenario, which leads to a compressor being modeled as a special kind of edge that boosts the pressure from its suction inlet to its discharge outlet.

Compressors are typically modeled “ideally”, based on their power consumption, for example as a special node type; in-depth discussions can be found in [63, 123]. Due to this consumption model, such ideal compressor units are useful for energy utilization optimization tasks [40], yet, for a simplified transient simulation aspect already too complicated. A more practical approach is taken in [91, 144, 145], where a compression ratio \( \alpha_i \geq 1 \) is used to scale the pressure in each node, or pipe [130], which means \( \alpha_i > 1 \) indicates compression / a compressor, otherwise (\( \alpha_i = 1 \)) a pipe.

Here, we utilize a likewise simple compressor model similar to [46, 139], for which we assume it is propelled by an external energy source, for example, given a compressor electrification, by excess renewable power [41], or that the off-take in gas is insignificant. But instead of using compression ratios, a constant (or parametric) target pressure is prescribed, modeling discharge pressure control [129].

The following affine compressor model is a variant of the compressor presented in [126] and used in [7]. We model a simplified compressor by a level, short pipe which increases the pressure at its outflow to a specified value \( \hat{p}_c \) (and without friction, \( \lambda_{ij} \equiv 0 \)). Given the pipe from nodes \( i \) to \( j \) is treated as such a “compressor pipe”, with target pressure of \( \hat{p}_c \), the variables \( p_{ij} \) and \( q_{ij} \) are given by the differential equations:

\[
\begin{align*}
\dot{p}_{ij}(t) &= q_j - q_i, \\
\dot{q}_{ij}(t) &= \hat{p}_c - p_i.
\end{align*}
\]  

(7)

The target pressure \( \hat{p}_c \) could be a step function \( \hat{p}_c(t) \), and hence a control input [130], which could be accompanied by a discharge mass-flux output.

A compressor could also be interpreted as an actual pipe with “negative friction”, and we considered using such nonphysical pipes as compressor model, but a difficult transformation between friction and pressure increase would have to be calibrated for every model variant (including friction factor formulas) and updated with every change in any model.

2.6 Parametrization

For the considered pipeline model (1), two scalar parameters are of practical interest: The temperature of the gas \( T_0 \) (in [K]), which is assumed to be constant throughout the network, and the global specific gas constant \( R_S \) (in [J/(kg K)]).

Due to mainly underground on-shore pipelines [94, Ch. 45], and coolers in compressor stations [90], using an isothermal model is a reasonable simplification. However, temperature is relevant as a global parameter, since the use of an isothermal model “freezes” the
dynamic energy (temperature) component in the original Euler equations (in time), and while intraday ambient temperature variation can be neglected for simulations with a 24-hour time horizon, the temperature difference of a hot summer day and a cold winter day should be taken into account by a parameter representing an average temperature.

The specific gas constant, on the other hand, is determined by the gas composition, which may also vary. Again, local variations during an intraday simulation are neglected in this work, yet the average gas mixture of natural gas with, for example, hydrogen or biogas is relevant, so a parameter for the average specific gas constant is introduced. Together, the parameter-space \( \Theta_1 \) is given by:

\[
\theta := \begin{pmatrix} T_0 \\ R_S \end{pmatrix} \in \Theta_1 \subset \mathbb{R}^2,
\]

and note that \( \theta \) is used in the model (only) as \( d_0 = \frac{1}{\gamma_0} = \frac{1}{\gamma_0 \gamma_0} = \frac{1}{\gamma_0 \gamma_0} \). Yet, lumping into a single parameter would impede physical interpretation.

Applied to the respective components of the input-output model in Sect. 2.7, this leads to parameter-dependent quantities, which need to be regarded accordingly by the model reduction as discussed in Sect. 3.4.

2.7 Input-output model

After spatial discretization and index reduction, we end up with a square input-output system, a system with the same number of inputs and outputs, consisting of an ordinary differential equation, an output function and an initial value:

\[
\begin{pmatrix} E_p(\theta) \\ 0 \\ E_q \end{pmatrix} \begin{pmatrix} \dot{p} \\ \dot{q} \end{pmatrix} = \begin{pmatrix} 0 \\ A_{pq} \\ 0 \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} + \begin{pmatrix} 0 \\ B_{pd} \\ 0 \end{pmatrix} \begin{pmatrix} s_q \\ d_q \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \begin{pmatrix} F_c \\ f(p,q,s_p,\theta) \end{pmatrix},
\]

\[
\begin{pmatrix} s_q \\ d_p \end{pmatrix} = \begin{pmatrix} 0 \\ C_{dp} \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix},
\]

\[
\begin{pmatrix} p_0 \\ q_0 \end{pmatrix} = \begin{pmatrix} \hat{p}(s_p,\hat{d}_q) \\ \hat{q}(s_p,\hat{d}_q) \end{pmatrix},
\]

with parameter independent linear vector field components \( A \) and \( B \), parametric mass matrix \( E(\theta) \), and nonlinear friction and gravity retarding term \( F(p,q,s_p,\theta) \). The actual composition of the dynamical system components depends on the discretization and index reduction, cf. Sect. 2.4, while the linear output function \( C \) consists of \( C_{dp} = -B_p \) and \( C_{dp} = B^T_q \).

The load vector \( F_c \in \mathbb{R}^{N_q} \) accumulates the respective discharge pressures \( \hat{p}_c \), as described in Sect. 2.5, for all compressors, and the initial state is given by a steady-state, whose computation is detailed in Sect. 2.8, depending on given steady-state boundary values \( \hat{s}_p, \hat{d}_q \).

Altogether, the gas network model is a generalized linear system \((E,A,B,C)\), together with a nonlinear part \( f \).

A control system formulation of transient gas network simulation was already formulated in [81], and recently in [140]. Also in [35], a so-called “systemic interpretation” is discussed; inspired by [35, Fig. 2], we schematically illustrate (8) in Fig. 1.

The specific structure and grouping for the single pipeline model (1), the index-reduced spatially discrete network models (4a)–(4b), (6a)–(6b), and the input-output model (8),
is justified by the numerical processing: only two model components depend on the parameters, temperature $T_0$ and specific gas constant $R_S$, as well as on the compressibility factor $z_0$, namely the mass matrix $E_p$ and the jointly treated retarding forces gravity and friction $f_q$. Hence, the linear part of the right-hand side vector field is non-parametric and compressibility-agnostic.

Overall, this system maps input boundary values, in the scope of this work, pressure at the inlets and mass-flux at the outlets, via the internal state, to output quantities of interest, here, mass-flux at the inlets and pressure at the outlets:

$$
\begin{pmatrix}
sp(t) \\
sp(t)
\end{pmatrix}
\rightarrow
\begin{pmatrix}
sq(t) \\
sp(t)
\end{pmatrix}
\rightarrow
\begin{pmatrix}
p(t) \\
q(t)
\end{pmatrix}

To this type of input-output system we can now apply (data-driven) system-theoretic model reduction methods, which preserve the input-to-output mapping $S$, but explicitly not the internal state $(p, q)^T$. Lastly, we note that based on [74], we added a model fact sheet in the Appendix.

### 2.8 Steady-state computation

After spatial discretization, the dynamic simulation becomes an initial value problem. Yet, only the boundary values of the network model are known a-priori. This means the internal state at time $t = 0$ is unknown. We assume simulations always start at a steady-state $\bar{p}, \bar{q}$ for which $\dot{\bar{p}} = \dot{\bar{q}} = 0$, given some (initial) boundary values $\bar{s}_p, \bar{d}_q$. The internal state is then computable as a steady-state problem. Since the employed model is nonlinear, we approximate the steady-state by a two-step procedure:

1a. Linear mass-flux steady-state: $A_{pq}\tilde{q} = -B_{pq}\tilde{d}_q$.
1b. Linear pressure steady-state: $A_{qp}\tilde{p} = -(B_{qs}\tilde{s}_p + F_c)$.
2. Corrected pressure steady-state: $A_{qp}\tilde{p} = -(B_{qs}\tilde{s}_p + F_c + f_q(\tilde{p}, \tilde{q}, \tilde{s}_p, \theta))$.  

Figure 1 Schematic illustration of gas network input-output model
Step 2 can be repeated until an error threshold is met by using the previously approximated pressure steady-state. Practically, the linear problems in Step 1 and Step 2 are solved by a QR-based least-norm method [23]. Note, that Step 1a and Step 1b can be solved in parallel and that the QR decomposition of Step 1b can be recycled in Step 2 because of the chosen model structure.

While this method works well for rooted-tree pipe-networks, it is not sufficient for cyclic networks with multiple supply nodes and non-pipe elements such as compressors. In this case, the resulting state after a limited number of the above algorithm’s iterations is used as an initial value for the first order IMEX integrator detailed in Sect. 5.3.3, which time-steps until a steady-state is sufficiently approximated. This approximate steady-state, associated to a fixed set of boundary values and parameters, is used as initial value for the simulations:

\[
\begin{pmatrix}
    p_0 \\
    q_0
\end{pmatrix}
= \begin{pmatrix}
    \tilde{p}(\tilde{s}_p, \tilde{d}_q) \\
    \tilde{q}(\tilde{s}_p, \tilde{d}_q)
\end{pmatrix}.
\]

While other time steppers are applicable, too, the first order IMEX solver is related to the initial (two-step) algebraic approximation, due to the synthesis of the linear / input / source and nonlinear / reaction terms.

2.9 Port-Hamiltonian structure

An interesting class of models are port-Hamiltonian systems, which have already been used for gas network modeling [39, 86]. Such port-controlled Hamiltonian models result from a system-theoretic approach to energy-based modeling, and are square, passive, stable and feature certain symmetries, besides their physical interpretability [15, 103]. To exploit results from port-Hamiltonian theory in the context of data-driven model reduction, we regiment the previous modeling approach into the port-Hamiltonian framework.

A linear input-state-output port-Hamiltonian model [135, Ch. 4] has the form:

\[
\begin{align*}
    E \dot{x}(t) &= (J - R)Q x(t) + (G - P) u(t), \\
    y(t) &= (G + P)^T Q x(t),
\end{align*}
\]

with a symmetric positive definite mass matrix \( E = E^T, E > 0 \), a skew-symmetric energy flux \( J = -J^T \), a symmetric, positive, semi-definite energy dissipation \( R = R^T, R \geq 0 \), a symmetric, positive definite energy storage \( Q, Q > 0 \), resistive port matrix \( P \) and control port matrix \( G \).

Here, we generalize the energy dissipation \( R \in \mathbb{R}^{N \times N} \) to a nonlinear mapping \( R : \mathbb{R}^N \rightarrow \mathbb{R}^{N \times N} \), this means the linear constraints become [38]:

\[
\begin{align*}
    R &= R^T & \Rightarrow & \langle R(x)x', x'' \rangle = \langle x', R(x)x'' \rangle, & \forall x, x', x'' \in \mathbb{R}^N, \\
    R \geq 0 & \Rightarrow & \langle R(x)x', x' \rangle \geq 0, & \forall x, x' \in \mathbb{R}^N.
\end{align*}
\]

Typically, the symbol \( \delta \) is used for this port matrix.
With this set up, we test the two index-reduced gas network model discretizations presented in Sect. 2.4.2 and Sect. 2.4.1 for compliance with the above port-Hamiltonian properties.

**Proposition 1** The endpoint discretization (6a)–(6b) is a port-Hamiltonian model.

**Proof** We define the port-Hamiltonian state as \( x := (p \ q)^T \), which induces the remaining components. The matrix mass

\[
E = \begin{bmatrix}
(A_{0,r}D_{q}^{-1}d_{0}(\theta)A_{0,e}) & 0 \\
0 & D_{q}^{-1}
\end{bmatrix}
\]

is symmetric positive definite, if its diagonal blocks are. Given that the \( D_{e} \) are diagonal, and thus symmetric, as well as positive definite, both blocks are symmetric positive definite.

The energy flux

\[
J = \begin{bmatrix}
0 \\
-A_{0}
\end{bmatrix}
\]

is skew-symmetric by definition. The energy dissipation (see \( f_{q} \))

\[
R(x) := \begin{bmatrix}
0 \\
0 & (D_{q} \text{diag}(d_{0}(\theta)A_{0,e}^{T}B_{q}p) + D_{q}^{-1}D_{f} \text{diag}(\frac{qL}{d_{0}(\theta)A_{0,e}^{T}B_{q}p}))
\end{bmatrix}
\]

with the \( \text{diag} : \mathbb{R}^{N} \rightarrow \mathbb{R}^{N \times N} \) operator mapping a vector \( v \) to a diagonal matrix \( D \) such that \( v_{k} \mapsto D_{kk} \), and element-wise (fraction) nonlinearities, results in one non-zero diagonal block and thus fulfills (10). The condition (11) is fulfilled since in the friction term of \( R \), the absolute value of the mass-flux (numerator), and the nodal pressure variable (denominator) are always non-negative.

Here, the energy storage represents the scale homogenization from Sect. 2.2,

\[
Q = \begin{bmatrix}
(10^{5} \cdot I_{N_{p}}) & 0 \\
0 & (10^{-5} \cdot I_{N_{q}})
\end{bmatrix}
\]

which is a diagonal matrix of positive entries, and due to same block structure in \( E \) also fulfills \( Q^{T}E = E^{T}Q \). Lastly, the port matrix configuration

\[
P := \begin{bmatrix}
0 & 0 \\
B_{S}^{T} & 0
\end{bmatrix}, \quad G := \begin{bmatrix}
0 & B_{D}
\end{bmatrix},
\]

complies to the port-Hamiltonian form.

□

Some remarks are in order on this result: From the previous proof it is also immediately clear that the midpoint discretization cannot be a port-Hamiltonian model, due to the input dependence of the energy dissipation. Furthermore, this derivation tests if the endpoint discretization has the mathematical port-Hamiltonian structure, but does not verify a physical energy-based model.
The somewhat nonphysical treatment of the gravity term as dissipating instead of storing ([38]), is done with regard to the parametrization. Including the parametric gravity term as a retarding or damping force, and thus keeping the linear energy flux parameter-free, enables the previous steady-state computation.

Compressors, as modeled in Sect. 2.5 can be included by an additional summand inside the energy dissipation component, i.e. $\frac{\dot{p}}{\dot{q}}$, similar to the gravity term. This exhibits an unphysical negative sign inside the dissipation, as a compressor introduces energy. Furthermore, this compressor model requires to remove components from the $A_{qp}$ block of the system matrix (7), and thus perturbs the skew-symmetry of $J$.

Lastly, this notation for the dissipation can also be used for linearization, by constraining the argument of $R$ to the steady-state $\bar{x}$,

$$\bar{R} := R(\bar{x}) \approx R(x).$$

Given the port-Hamiltonian model with a nonlinear resistive term, an (approximate) adjoint system can be derived by treating $R$ as its image – a diagonal matrix. Transposing the (primal) port-Hamiltonian system’s (9) transfer function $h(s) = (G + P)^T Q (E_s - J - R Q)^{-1} (G - P)$, and exploiting the system components properties, yields the dual system:

$$\begin{align*}
E\dot{x}(t) &= Q(-J - R(x(t))) x(t) + Q(G + P) u(t), \\
y(t) &= (G - P)^T x(t).
\end{align*}
$$

Hence, for the (nonlinear) endpoint discretization, its observability can be (approximately) quantified by the dual system’s reachability, as for linear systems. Conceptually, this could also be done with the midpoint discretization, as it supplies the same model components. However, it has no theoretical justification, as a dual system may not be accessible for (general) nonlinear systems.

3 Model reduction for gas networks

In this section, we summarize the principal approach behind all presented model reduction methods that are extended and tested in this work. The structure of the model laid out in Sect. 2 is given by (8). For large (expansive) networks, the differential equations in $p$ and $q$ become high dimensional, which impedes their solution [54, Sect. 7] and hence repeated simulations of scenarios. The aim of model reduction is to reduce the dimensionality of the differential equations, by computing subspaces of the phase space on which the trajectories evolve suitably similar (with regard to the quantities of interest). Furthermore, the reduced order model shall have the same form as the original model, and since two physical quantities are (bi-directionally) coupled in this system, the model reduction for interconnected systems [120] approach is used, yielding reduced operators for each sub-
system:
\[
\begin{pmatrix}
\tilde{E}_p(\theta) & 0 \\
0 & \tilde{E}_q
\end{pmatrix}
\begin{pmatrix}
\tilde{p} \\
\tilde{q}
\end{pmatrix}
= \begin{pmatrix}
0 & \tilde{A}_{pq} \\
\tilde{B}_{pd} & 0
\end{pmatrix}
\begin{pmatrix}
\tilde{p} + \tilde{p} \\
\tilde{q} + \tilde{q}
\end{pmatrix}
+ \begin{pmatrix}
0 & \tilde{B}_{pq} \\
\tilde{B}_{qd} & 0
\end{pmatrix}
\begin{pmatrix}
\tilde{s}_p \\
\tilde{d}_q
\end{pmatrix}
+ \begin{pmatrix}
0 \\
\tilde{f}_c(\tilde{p} + \tilde{p}, \tilde{q} + \tilde{q}, \tilde{s}_p, \theta)
\end{pmatrix}
\]
\[
\begin{pmatrix}
\tilde{s}_p \\
\tilde{d}_q
\end{pmatrix}
\approx \begin{pmatrix}
\tilde{c}_{pq} \\
\tilde{c}_{dp}
\end{pmatrix}
\begin{pmatrix}
\tilde{p} + \tilde{p} \\
\tilde{q} + \tilde{q}
\end{pmatrix},
\]
\[
\begin{pmatrix}
\tilde{p}_0 \\
\tilde{q}_0
\end{pmatrix}
= \begin{pmatrix}
0 \\
0
\end{pmatrix},
\]
centered around the steady-state \((\tilde{p}, \tilde{q})\). This structure preserving model order reduction was already used in [18, 53] in the context of model reduction for gas networks, while the centering has been used in [7] for gas network simulation and in [65] for nonlinear model order reduction. In the following, the general ansatz to obtain these reduced quantities (denoted by \(\tilde{\cdot}\)) is summarized.

### 3.1 Projection-based model reduction

The reduced order model is computed by projecting the high-dimensional dynamics evolving in the (coupled) pressure and mass-flux phase spaces (of dimension \(N_p\) and \(N_q\)) to low(er)-dimensional subspaces (of dimension \(n_p\) and \(n_q\)), which capture the principal components of the respective trajectories. Given suitable discrete projection mappings from the original space to the reduced space \(V_a^T\) and mappings from the reduced space back to the original space \(U_a\):

\[
U_p : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{N_p}, \quad V_p^T : \mathbb{R}^{N_p} \rightarrow \mathbb{R}^{n_p} : \quad V_p^T \cdot U_p = I_{n_p},
\]
\[
U_q : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{N_q}, \quad V_q^T : \mathbb{R}^{N_q} \rightarrow \mathbb{R}^{n_q} : \quad V_q^T \cdot U_q = I_{n_q}.
\]

Thus, the reduced trajectory results from applying \(V_a\) to the original trajectory’s steady-state deviation, while the original trajectory is approximately recovered by applying \(U_a\) to the reduced trajectory:

\[
\begin{pmatrix}
\tilde{p} \\
\tilde{q}
\end{pmatrix}
:= \begin{pmatrix}
V_p^T(p - \bar{p}) \\
V_q^T(q - \bar{q})
\end{pmatrix}
\rightarrow \begin{pmatrix}
\tilde{p} + U_p \tilde{p} \\
\tilde{q} + U_q \tilde{q}
\end{pmatrix}
\approx \begin{pmatrix}
p \\
q
\end{pmatrix}.
\]

the initial condition is also reduced by application of \(V_a\). Similarly, the components of the reduced system result from applying the \(U_a\) map to the argument of the respective operators, and the \(V_a\) map to the result of the operation.

For the linear operators, the matrices \(E_a, A_a, B_a\) and \(C_a\) and the vector \(F_a\), this leads conveniently to pre-computable reduced matrices and vector respectively,

\[
\tilde{A}_{pq} := V_p^T \cdot A_{pq} \cdot U_q \in \mathbb{R}^{n_p \times n_q}, \quad \tilde{A}_{qp} := V_q^T \cdot A_{qp} \cdot U_p \in \mathbb{R}^{n_q \times n_p},
\]
\[
\tilde{B}_{pd} := V_p^T \cdot B_{pd} \in \mathbb{R}^{n_p \times N_p}, \quad \tilde{B}_{qd} := V_q^T \cdot B_{qd} \in \mathbb{R}^{n_q \times N_q},
\]
\[
\tilde{C}_{dp} := C_{dp} \cdot U_p \in \mathbb{R}^{N_d \times n_p}, \quad \tilde{C}_{dq} := C_{dq} \cdot U_q \in \mathbb{R}^{N_d \times n_q},
\]
\[
\tilde{E}_p(\theta) := V_p^T \cdot E_p(\theta) \cdot U_p \in \mathbb{R}^{n_p \times n_p}, \quad \tilde{E}_q := V_q^T \cdot E_q \cdot U_q \in \mathbb{R}^{n_q \times n_q},
\]
\[
\tilde{F}_c := V_q^T \cdot F_c \in \mathbb{R}^{n_q},
\]
yet, the nonlinear component \( \tilde{f}_q \) remains a composition operation:

\[
\tilde{f}_q := V_q^T \cdot f_q(p + U_p \tilde{p}, \tilde{q} + U_q \tilde{q}, s_p, \theta) : \mathbb{R}^{np} \times \mathbb{R}^{nq} \times \mathbb{R}^{Ns} \times \mathbb{R}^2 \rightarrow \mathbb{R}^{nq}.
\]  

(13)

### 3.2 Structure preserving model order reduction

In this specific context, the term structure preserving model order reduction (SPMOR) has two meanings: first and foremost, SPMOR refers to the separate reduction of the state components, as above in the case of gas networks, the individual reduction of the discretized pressure \( p \) and mass-flux \( q \) variables. Second, SPMOR can also refer to preserving the port-Hamiltonian form (9). For projection-based model reduction, the former is generally ensured by separate projectors [44] (or an overall block diagonal projection). The latter is guaranteed by using Galerkin projections, which implies stability preservation [16], given a port-Hamiltonian full order model. Both SPMOR interpretations are jointly fulfilled if a block-diagonal (w.r.t. \( p \) and \( q \)) Galerkin projection is used.

### 3.3 The lifting bottleneck and hyper-reduction

The gas network models considered for reduction are nonlinear (and potentially non-smooth), hence the reduced order nonlinear part \( \tilde{f}_q \) involves lifting the reduced state up to the original high-dimensional space, evaluating the nonlinearity and projecting the result back down to the reduced low-dimensional space (13). As the high-dimensional space is involved, this is typically computationally demanding and may eat up the gains from the reduction of the linear part. To mitigate this so-called lifting bottleneck, hyper-reduction methods can be employed, which construct low-dimensional surrogates for nonlinearities.

In this work we discard (or rather defer) hyper-reduction due to the following reasoning: The purpose of this work is to determine which method constructs the best reduced order models, a hyper-reduction may inhibit comparability due to, for example, a dominating hyper-reduction approximation error. Second, various hyper-reduction methods for this setting are applicable (i.e. DEIM [26], Q-DEIM [36], DMD [142] or numerical linearization [98]), which may interact differently with the different model reduction methods. So as a first step, the bare model reduction methods are tested here (this means: which method’s linear subspaces capture the nonlinear dynamics best), at a later stage the best hyper-reduction method can then be determined. Lastly we note, the nonlinear part of the vector field consists exclusively of element-wise operations (see Sect. 2.9), a system with repeated scalar nonlinearities (SRSN) [28], which are less difficult to handle due to “locality” of the nonlinearity, and hence, its vectorization.

### 3.4 Parametric model reduction

There are two common approaches for projection-based parametric model order reduction: averaging and accumulating [67]. For the selected data-driven methods, averaging means that for a set of parameter samples the associated trajectories or derived quantities (such as the utilized system Gramians) are averaged, while accumulating refers to the concatenation of trajectories or derived quantities (such as the projectors).

Generally, each of the structure-preserving model order reduction methods in Sect. 4 can be used with either, we opted to use the averaging ansatz for all of the following methods since their computation is without exception based on parametric empirical Gramians [71].
4 Model reduction methods

In this section, we briefly summarize the employed model reduction methods from a practical point of view. For theoretical details and backgrounds we refer to the relevant works, cited in the respective subsections. Due to the non-differentiable nonlinearity (friction), the sought projections $U_*$, $V_*$ for all tested model reduction techniques are constructed from (transformed) time-domain trajectory data obtained from numerical simulations, which is given by discrete-time snapshots of the internal pressure nodes $\hat{X}_p$ and mass-flux edges $\hat{X}_q$:

$$\hat{X}_p(t; \theta_k) = \begin{bmatrix} p^1(t; \theta_k) & \ldots & p^{N_p+N_d}(t; \theta_k) \end{bmatrix},$$

$$\hat{X}_q(t; \theta_k) = \begin{bmatrix} q^1(t; \theta_k) & \ldots & q^{N_p+N_d}(t; \theta_k) \end{bmatrix},$$

the external demand node pressure $\hat{Y}_p$ and supply node mass-flux $\hat{Y}_q$,

$$\hat{Y}_p(t; \theta_k) = \begin{bmatrix} (s_{1p}^1(t; \theta_k)) & \begin{bmatrix} d_{1p}^1(t; \theta_k) \end{bmatrix} & \ldots & (s_{N_p}^{N_p}(t; \theta_k)) & \begin{bmatrix} d_{N_p}^{N_p}(t; \theta_k) \end{bmatrix} \end{bmatrix},$$

$$\hat{Y}_q(t; \theta_k) = \begin{bmatrix} (s_{1q}^1(t; \theta_k)) & \begin{bmatrix} d_{1q}^1(t; \theta_k) \end{bmatrix} & \ldots & (s_{N_q}^{N_q}(t; \theta_k)) & \begin{bmatrix} d_{N_q}^{N_q}(t; \theta_k) \end{bmatrix} \end{bmatrix},$$

as well as dual state components $\hat{Z}_p$ and $\hat{Z}_q$, if available,

$$\hat{Z}_p(t; \theta_k) = \begin{bmatrix} p^1(t; \theta_k) & \ldots & p^{N_p+N_d}(t; \theta_k) \end{bmatrix},$$

$$\hat{Z}_q(t; \theta_k) = \begin{bmatrix} q^1(t; \theta_k) & \ldots & q^{N_p+N_d}(t; \theta_k) \end{bmatrix}.$$
from plain linearization: instead, the implemented methods assemble linear subspaces of the model’s phase space that, in a method specific sense, approximately enclosing the relevant nonlinear system evolution. Moreover, all methods are SVD-based [4], and their majority is originally based on (empirical) system Gramian matrices, for details see [66].

We highlight here that the time horizon for the training data is significantly shorter than for the actual simulations the reduced order model is targeted at. Furthermore, generic training inputs (transient boundary values), such as impulse, step or random signals, are utilized to avoid a model reduction crime [67] (comparable to an inverse crime): Test a reduced order model using the training parameters or inputs.

4.1 Empirical system Gramians
All model reduction methods currently included in morgen are computationally realized using empirical system Gramian matrices, which are system-theoretic operators encoding reachability and observability. From these, information on importance of linear combinations of states can be extracted. For linear systems, these system Gramians are typically computed via matrix equations, for general nonlinear systems there is no feasible closed form. However, the empirical system Gramians approximate the nonlinear Gramians based on state and output trajectory data. In case of a port-Hamiltonian (nonlinear) systems, the approximate dual system (12) enables substituting expensive state by port perturbations, and thus severely reduce computation times. Empirical Gramians are described in-depth in [66]; following only a brief summary is given.

4.1.1 Empirical reachability Gramian matrix
Reachability quantifies how well a system can be driven by the inputs, which is encoded by the reachability Gramian. The empirical reachability Gramian is an approximation based on state trajectory data [84]:

$$\hat{W}_{R,e} := \sum_{k=1}^{K} \sum_{m=1}^{N_e+N_d} \int_{0}^{\infty} \hat{X}_m^*(t;\theta_k) \hat{X}_m(t;\theta_k)^T \, dt \in \mathbb{R}^{N_e \times N_e}.$$ (14)

Given a suitable set of input perturbations, $\hat{W}_{R,e}$ approximates the nonlinear reachability Gramian near a steady-state [56].

4.1.2 Empirical observability Gramian matrix
Observability quantifies how well the state can be characterized from the outputs, which is encoded by the observability Gramian. The empirical observability Gramian is an approximation based on output trajectory data [84]:

$$\hat{W}_{O,e} := \sum_{k=1}^{K} \int_{0}^{\infty} \hat{Y}_e(t;\theta_k)^T \hat{Y}_e(t;\theta_k) \, dt \in \mathbb{R}^{N_e \times N_e}.$$ (15)

For a suitable set of steady-state perturbations, $\hat{W}_{O,e}$ approximates the nonlinear observability Gramian near a steady-state [56].

Given the port-Hamiltonian structure of a discretization, the empirical reachability Gramian of the dual system (12) can be used to compute the empirical observability Gramian [67, Sect. 2] with $\hat{Z}_e^m(t;\theta_k)$ instead of $\hat{X}_e^m(t;\theta_k)$.
4.1.3 Empirical cross Gramian matrix

The empirical cross Gramian concurrently encodes reachability and observability, which in conjunction quantifies redundancy also known as minimality; however, the (empirical) cross Gramian is only applicable for square systems, systems with the same number of inputs and outputs (which the gas network model (8) fortunately is), and an approximation is based on simulated state and output trajectory data [70]:

\[
\hat{W}_{X,*} := \sum_{k=1}^{K} \sum_{m=1}^{N_s} \sum_{q=1}^{N_s} \int_0^\infty \hat{X}_m(t;\theta_k) \hat{Y}_q(t;\theta_k) \, dt \in \mathbb{R}^{N_s \times N_s}. \tag{16}
\]

For a suitable set of input and steady-state perturbations, \( \hat{W}_{X,*} \) approximates the nonlinear cross Gramian near a steady-state [70].

Given the port-Hamiltonian structure of a discretization, the linear empirical cross Gramian exploiting the dual system (12) can be used to compute the empirical cross Gramian [14] with \( \hat{Z}_m(t;\theta_k)^T \) instead of \( \hat{Y}_s(t;\theta_k) \).

4.1.4 Empirical non-symmetric cross Gramian matrix

A generalization of the empirical cross Gramian for non-square systems is the empirical non-symmetric cross Gramian, which is an approximation based on simulated state and (averaged) output trajectory data [72]:

\[
\hat{W}_{Z,*} := \sum_{k=1}^{K} \sum_{m=1}^{N_s} \sum_{q=1}^{N_s} \sum_{q=1}^{N_s} \int_0^\infty \hat{X}_m(t;\theta_k) \hat{Z}_q(t;\theta_k) \, dt \in \mathbb{R}^{N_s \times N_s}. \tag{17}
\]

For a suitable set of input and steady-state perturbations, \( \hat{W}_{Z,*} \) approximates the nonlinear cross Gramian near a steady-state. Even though the gas network model (8) is square, the empirical non-symmetric cross Gramian is included here, since, heuristically, it could provide better results than the regular cross Gramian [72]. Furthermore, an empirical non-symmetric linear cross Gramian is computable by similarly averaging over the dual states and replacing \( \hat{Y}_s(t;\theta_k) \) by \( \hat{Z}_s(t;\theta_k)^T \).

4.2 Structured proper orthogonal decomposition

Proper orthogonal decomposition (POD) is a basic data-driven method for model reduction: given a matrix of state snapshots over time, the dominant left singular vectors are computed as a basis via a singular value decomposition (SVD) [97]. The basis vectors are assigned their principality with respect to the conveyed energy by the associated (relative) singular value magnitude.

In the context of this work, the POD is constructed from a system-theoretic point of view, that connects to the system property of reachability. Due to the overall structured approach to model reduction, a structured POD refers in this context to the separate PODs for pressure and mass-flux variables \( p \) and \( q \) as in [18, 51, 53].

4.2.1 Reachability-Gramian-based

The singular vectors to the principal singular values of the empirical reachability Gramian (14) correspond to the POD modes. To obtain a reduced order model, first, a
truncated SVD (tSVD) of the empirical reachability Gramian,

\[ \tilde{W}_{R,*}^{tSVD} = U_{R,*} D_{R,*} U_{R,*}^T, \]

reveals the principal subspace of the respective trajectories \( \tilde{X}_m^*(t) \), whereas the importance of each basis vector (column) in \( U_{R,*} \) is determined by (the square-root of) the associated singular value \( \sigma_i := D_{R,ii}^{1/2} \).

\[ U_\ast := U_{R,*}. \]

The matrix of basis vectors (POD modes) constitutes a Galerkin projection \( V_\ast := U_\ast \). Notably, (structured) POD only considers the input-to-state mapping, not the state-to-output mapping, and hence approximates the state variables, \( p \) and \( q \), not the output quantities of interest \( s_q \) and \( d_p \).

The POD could also be computed directly from an SVD of the trajectory data, yet the computational overhead of using the empirical reachability Gramian is small compared to the trajectory simulation runtimes, and the systematic perturbations of the empirical Gramian approach [66] are exploited.

### 4.3 Structured empirical dominant subspaces

The previous (structured) POD method considers only the reachability information, hence the data only reflects the input-to-state mappings, and thus the POD derived ROMs (Reduced Order Model) approximate the state variables \( p \) and \( q \). To approximate the outputs \( s_q \) and \( d_p \), the state-to-output mappings, encoding observability information, need to be considered, too.

The (empirical) dominant subspaces method initially developed in [110], and originally named DSPMR (Dominant Subspace Projection Model Reduction), conjoins and compresses the dominant reachability and observability subspaces of an input-output system, such as the gas network model (8), obtained from (empirical) system Gramians. Heuristically, this method seems to be useful for hyperbolic input-output systems [49]. Here, we consider three variants: first, based on the empirical reachability and observability Gramians, second, based on the empirical cross Gramian and third, based on the empirical non-symmetric cross Gramian.

#### 4.3.1 Reachability- and observability-Gramian-based

The singular vectors associated to the principal singular values of the empirical reachability and observability Gramians span these dominant subspaces, which are first extracted by tSVDs, and then, after concatenation, compressed by orthogonalization via another tSVD:

\[ \tilde{W}_{R,*}^{tSVD} = U_{R,*} D_{R,*} V_{R,*}^T, \]

\[ \tilde{W}_{O,*}^{tSVD} = U_{O,*} D_{O,*} V_{O,*}^T, \]

\[ \begin{bmatrix} \omega_R U_{R,*} D_{R,*} \\ \omega_O U_{O,*} D_{O,*} \end{bmatrix} \overset{tSVD}{=} U_{RO,*} D_{RO,*} V_{RO,*}, \]
from which the singular vectors \( V_* = U_* := U_{RO,*} \) make up a Galerkin projection. The weights \( \omega_R := \|\hat{W}_{r,*}\|_F^{-1} \) and \( \omega_O := \|\hat{W}_{o,*}\|_F^{-1} \) equilibrate the potentially mismatched scales of the respective Gramians, akin to the refined DSPMR method from [110].

4.3.2 Cross-Gramian-based

A truncated SVD of the cross Gramian also engenders the sought dominant subspaces [20]: The (empirical) cross Gramian’s left and right singular vectors (approximately) span the reachability and observability subspaces, respectively, and their orthogonalized concatenation, via a truncated SVD,

\[
\hat{W}_{X,*}^\text{tSVD} \equiv U_{X,*} D_{X,*} V_{X,*}^T,
\]

\[
\begin{bmatrix}
(U_{X,*} D_{X,*}) & (V_{X,*} D_{X,*})
\end{bmatrix}^\text{tSVD} \equiv U_{RO,*} D_{RO,*} V_{RO,*},
\]

yields a Galerkin projection \( V_* = U_* := U_{RO,*} \). The cross-Gramian-based variant does not need to additionally weight the reachability subspace \( U_{X,*} \) and observability subspace \( V_{X,*} \), as they are both extracted from the same matrix.

This cross-Gramian-based empirical dominant subspaces method seamlessly extends to the empirical non-symmetric cross Gramian \( \hat{W}_{Z,*} \).

4.4 Structured empirical balanced truncation

The dominant subspaces method combines separately quantified input-to-state and state-to-output energies, but not the actual input-to-output energy. Such can be accomplished by balanced truncation and based on the Hankel operator, which maps past inputs to future outputs. This operator’s singular values measure the sought input-to-output energy, and the singular vectors constitute a basis. To obtain the Hankel operator’s truncated SVD, first, the underlying system needs to be balanced, and then singular vectors associated to small magnitude Hankel singular values are truncated.

Balanced truncation is the reference model reduction method for linear input-output systems, due to stability preservation in the ROMs and a computable error bound. For (control-affine) nonlinear input-output systems, such as the gas network model (8), balanced truncation can be generalized to empirical balanced truncation [84], while a structured variant is introduced as interconnected system balanced truncation [136], which we combine.

Again, we consider three variants: First, based on the empirical reachability and observability Gramian, second, based on the empirical cross Gramian and third, on the empirical non-symmetric cross Gramian; additionally, the reachability and observability-based balanced POD variant is included.

4.4.1 Reachability- and observability-Gramian-based

The original balanced truncation method is based on the reachability and observability Gramians [99]. A transformation into a balanced coordinate system in which both system Gramians are diagonal and equal is obtained via simultaneous diagonalization. Various balancing algorithms are available for this task [137], in this setting we selected the general balancing algorithm [118, 119], which utilizes the magnitude-based truncated eigenvalue-decomposition (tEVD) of the Gramians: The matrices \( U_* \), \( V_* \) constitute a Petrov-Galerkin
projection, whereas the importance of each column is determined by the approximate Hankel singular values, the (diagonal) elements of $D_{B,*}$:

$$\hat{W}_{R,*} \hat{W}_{O,*} T_{R,*}^\text{tEVD} = \Lambda_{R,*} T_{R,*},$$

$$(\hat{W}_{R,*} \hat{W}_{O,*})^\top T_{O,*}^\text{tEVD} = \Lambda_{O,*} T_{O,*},$$

$$T_{O,*}^\top T_{R,*}^\text{tSVD} = U_{B,*} D_{B,*} V_{B,*}^\top,$$

$$U_* := T_{R,*} V_{B,*} D_{B,*}^{-\frac{1}{2}},$$

$$V_* := T_{O,*} U_{B,*} D_{B,*}^{-\frac{1}{2}}.$$

(18)

4.4.2 Cross-Gramian-based

For linear, symmetric systems, alternatively a balanced and truncated reduced order model can be computed via the cross Gramian. Yet, the gas network model (8) is neither linear nor symmetric, not even in a nonlinear sense of symmetric systems, i.e. gradient systems [75], but the considered input-output system is square. Hence, an empirical cross-Gramian-based reduced model is computable, but it will differ from the reduced model obtained by reachability- and observability-Gramian-based balanced truncation.

Given a cross Gramian with full rank, an (approximate) balancing projection is computable in a similar manner as for the (empirical) balanced truncation (18), but based on the left and right eigen spaces of the cross Gramian [77]:

$$\hat{W}_{X,*} T_{R,*}^\text{tEVD} = \Lambda_* T_{R,*},$$

$$\hat{W}_{X,*}^\top T_{O,*}^\text{tEVD} = \Lambda_* T_{O,*},$$

$$T_{O,*}^\top T_{R,*}^\text{tSVD} = U_{X,*} D_{X,*} V_{X,*}^\top,$$

$$U_* := T_{R,*} V_{X,*} D_{X,*}^{-\frac{1}{2}},$$

$$V_* := T_{O,*} U_{X,*} D_{X,*}^{-\frac{1}{2}}.$$

(19)

The matrices $U_*$, $V_*$ again constitute a Petrov-Galerkin projection, and the importance of each column is determined by the absolute value of the (diagonal) elements of $D_{X,*}$, which are only equal to the Hankel singular values for linear and symmetric systems.

As for the empirical dominant subspaces method, the cross-Gramian-based empirical balanced truncation variant directly extends to the non-symmetric cross Gramian $\hat{W}_{Z,*}$.

One could assume that if only one system Gramian has to be computed, instead of two for dominant subspaces or balanced truncation, that the cross-Gramian-based computation is significantly faster, but the overall number of simulated trajectories is the same for both methods, which causes the dominant fraction of computational cost. Thus, the empirical cross Gramian computation is merely somewhat quicker than the computation of both empirical reachability and observability Gramians.

4.4.3 Structured balanced POD

Instead of balancing the system using the product of the (empirical) reachability and observability Gramians, the dominant subspaces of the respective Gramians can be used
to approximately balance the system. This variant of reachability- and observability-
Gramian-based balanced truncation is called balanced POD [141], and the approximate
balancing algorithm reads:

\[
\tilde{\mathbf{W}}_{R,*} T_{R,*} \overset{\text{EVD}}{=} \Lambda_{R,*} T_{R,*},
\]

\[
\tilde{\mathbf{W}}_{O,*} T_{O,*} \overset{\text{EVD}}{=} \Lambda_{O,*} T_{O,*},
\]

\[
T_{O,*} T_{R,*} \overset{\text{SVD}}{=} U_{R,*} D_{B,*} V_{B,*},
\]

\[
U_* := T_{R,*} V_{B,*} D_{B,*}^{-1},
\]

\[
V_* := T_{O,*} U_{R,*} D_{B,*}^{-1},
\]

(20)

with the matrices \(U_*, V_*\) inducing a Petrov-Galerkin projection.

Here, we categorized balanced POD as a variant of balanced truncation, due the algo-
rithmic similarity to the balancing algorithm (18). Alternatively, we could have classified
balanced POD as a variant of POD, since it can be described as POD with the observability
Gramian defining the POD’s inner product [116].

We leave it to the reader to test other balancing algorithms, e.g. [17, 137], while we
excluded the modified POD [67, 102], as it is not a Petrov-Galerkin method.

4.5 Structured empirical balanced gains

The empirical balanced gains method is a variant of the empirical balanced truncation
method [30, 67]: While balanced truncation selects principality of subspaces based on the
Hankel singular value magnitude, balanced gains [30] sorts the balanced basis vectors by
the impulse response norm:

\[
\|x_p\|^2 \approx \text{tr}(C_{q, \tilde{\mathbf{W}}_R} C_{q, \tilde{\mathbf{W}}_R}^T) \approx \text{tr}(B_{q, \tilde{\mathbf{W}}_O} B_{q, \tilde{\mathbf{W}}_O}),
\]

\[
\|d_p\|^2 \approx \text{tr}(C_{d, \tilde{\mathbf{W}}_R} C_{d, \tilde{\mathbf{W}}_R}^T) \approx \text{tr}(B_{d, \tilde{\mathbf{W}}_O} B_{d, \tilde{\mathbf{W}}_O}),
\]

for a system in balanced form. Hence, either method, balanced truncation and balanced
gains compute the same balancing transformation, via (18) or (19), but the sequence of
basis vectors differs due to the variant measures.

We note, that due to the linear input and output operators, the balanced gains method
can be directly applied for the nonlinear gas network models (8).

4.5.1 Structured goal-oriented proper orthogonal decomposition

Similar to the balanced gains method, (structured) POD basis vectors can also be sorted,
instead of by their singular value magnitude \(\sigma_k\), in terms of their impulse response, akin
to the simplified balanced gains in [30], by the index \(d_k\)

\[
d_k := \tilde{\mathbf{c}}_k \tilde{\mathbf{c}}_k^T \sigma_k,
\]

for rows \(\tilde{\mathbf{c}}_k\) of the POD-transformed output matrix \(\tilde{\mathbf{C}}\). Given the reachability-based POD,
this variant is related to the concept of output-reachability and \(H_2\)-norm model reduc-
tion [112].
4.6 Structured dynamic mode decomposition - Galerkin

In addition to the four energy-based method classes (Sects. 4.2, 4.3, 4.4, 4.5), also an alternative method from [2], based on system identification, is investigated. Yet, it is adapted as a structured projection-based model reduction method. After summarizing the parent system identification method, the derived model reduction technique is presented.

4.6.1 Dynamic mode decomposition

Dynamic mode decomposition (DMD) identifies a discrete-time operator from (discrete) trajectory time-series data, preserving certain modal behavior [117]. This system identification method is based on the Koopman operator, which is an infinite dimensional, but linear operator, mapping (a transformation, or observable of the) state $x^k$ at discrete-time step $k$ to $(k + 1)$. DMD yields a (linear) finite-dimensional approximation of the Koopman operator preserving its dominant eigenmodes. Here, using the identity observable and given time series data $X = [x^0 \ldots x^K]$, DMD identifies an operator $\hat{A}$, via least-squares:

$$x^{k+1} \approx \hat{A}x^k, \quad (21a)$$

$$\hat{A} := \begin{bmatrix} x^1 & \ldots & x^K \end{bmatrix} \begin{bmatrix} x^0 & \ldots & x^{K-1} \end{bmatrix}^T. \quad (21b)$$

Since the underlying model is a control system, DMD with control (DMDc) [113] (and known input operator $B$), i.e. using $X^c := [x^0 - Bu^0 \ldots x^K - Bu^K]$ instead of $X$ for DMD, is applicable, yet, due to the perturbed steady-state training of ROMs (see Sect. 6.1) not beneficial.

4.6.2 Reachability-Gramian-based DMD-Galerkin

DMD is rather a system identification than a model reduction method. To fit into the projection setting, we utilize the DMD-Galerkin method [2], which forms a Galerkin projection $U$ from the orthogonalized dominant eigenvectors (based on the associated eigenvalue magnitudes),

$$\hat{A}^T \overset{\text{SVD}}{=} T \Lambda,$$

$$T \Lambda \overset{\text{SVD}}{=} UDV^T.$$

Since only state-space trajectories are utilized, as for the POD, the DMD-Galerkin method approximates the state, not the output. Curiously, we note, that this method uses discrete-time information to assemble projections for continuous-time systems.

Practically, $U_\pi$ are computed as a subset of singular vectors for the largest magnitude singular values, instead of orthogonalized eigenvectors. In the structured setting at hand $\hat{X}_\pi = [x^0_\pi \ldots x^K_\pi]$, with $X_\pi$ representing the pressure and mass-flux states $X_p, X_q$, the respective Galerkin projection $U_\pi = V_\pi$ is given by:

$$\begin{bmatrix} x^1_\pi & \ldots & x^K_\pi \end{bmatrix} \begin{bmatrix} x^0_\pi & \ldots & x^{K-1}_\pi \end{bmatrix}^T \overset{\text{SVD}}{=} U_\pi D_\pi V_\pi^T.$$

Effectively, the approximate Koopman operator $\hat{A}$ is computed via an empirical reachability Gramian, yet instead of the standard inner product, the DMD-“kernel” (21b) is used. Additionally, and in line with the original empirical Gramians [84], the utilized trajectories
are centered, following [73]. This means theoretically, DMD-Galerkin is POD with a specific kernel, and practically, that by computation via an empirical Gramian the systematic perturbation properties can also be exploited for DMD-Galerkin.

5 morgen
The morgen (Model Order Reduction for Gas and Energy Networks) platform (version 1.0) implements the mathematical methods presented above, in MATLAB (≥ 2020b) and compatible to Octave (≥ 6.1). Compared to, for example [62], morgen does not feature a graphical user interface or Simulink integration, since it is designed for batch testing and multi-query use on (headless) workstations.

The source code is organized into five main components:
- **networks** (holds network and scenario data-sets)
- **models** (discretizes networks and assembles input-output systems)
- **solvers** (computes solution time series for discrete models and scenarios)
- **reductors** (reduces state variables of discrete models)
- **tests** (defines experiments for data-model-solver-reductor combinations).

These components are briefly described in the following. For an illustration of the internal structure of morgen, see Fig. 2. Further code that is used by the main function or multiple components is contained in an **utils** folder, and the stand-alone network format converters are stored in a **tools** folder.

5.1 Networks
The **networks** directory stores the network .net files, and a folder for each network with the same name as the associated network file’s base name, which hold the scenario definition files. The .net file, in comma-separated value (CSV) format, defines the network topology and gas network components through an edge list. Each row encodes one edge through the information: edge type (pipe, shortcut, compressor, valve), “from”-node, “to”-node, length, diameter, incline, roughness; the latter three are only relevant for pipe edges.

Note that boundary nodes have to be leaf nodes of the network graph in order to be identified by morgen as such. Furthermore, if a boundary node shall act as supply and

---

8 Converters for GasLib XML, SciGRID_gas CSV, and MathEnergy JSON are available.
demand, it needs to be artificially split into two leaf nodes, as the edge connecting a supply node has to be directed from the leaf, while the edge connecting a demand node has to be directed towards the leaf (Sect. 2.3).

5.1.1 Scenario
A scenario is defined via a set of key-value pairs in an .ini file, whereas the sequence of pairs does not matter. Each network has at least a generic training.ini scenario. The following keys are mandatory for a scenario definition:
- \( T_0 \) Mean temperature
- \( R_S \) Mean specific gas constant
- \( t_H \) Time horizon
- \( u_t \) List of time instances
- \( u_p \) List of supply node pressures at time instances \( u_t \)
- \( u_q \) List of demand node mass-fluxes at time instances \( u_t \).

Depending on the network composition, the following keys may need to be provided:
- \( c_p \) List of compressor discharge pressures
- \( v_S \) List of valve settings.

Note that amongst other configurations, the parameter ranges of temperature and specific gas constant for the parametric model reduction are set in the global morgen.ini file.

5.2 Models
A model encodes a spatial discretization of the simplified Euler equations (8) on a gas network topology in a structure with the members: \( A, B, C, E, F, f \) and the Jacobian \( J \), together with the system dimensions in terms of number of pressure-at-nodes and mass-flux-on-edges states, and total number of boundary / port nodes. It is ensured during the assembly of the model that the sparsity of the model components is preserved.

The model interface is given by the following signature:

\[
\text{discrete} = \text{model} (\text{network}, \text{config});
\]

While the linear components \( A, B, C, \) and \( F \) are provided as sparse matrices, the parameter-dependent linear component \( E \) is a closure\(^9\) returning a sparse matrix, and the nonlinear component \( f \) and the (nonlinear) Jacobian \( J \) are closures returning the application of a state (as well as steady-state, input, parameters, compressibility).

Two spatial discretizations are currently provided in morgen:
- \text{ode\_mid} Midpoint ODE discretization (Sect. 2.4.1)
- \text{ode\_end} Endpoint ODE discretization (Sect. 2.4.2).

Even though both provided models are ODEs, DAE models can also be implemented and tested, given a solver (and reductor) is available.

5.3 Solvers
The morgen platform provides four solvers: An adaptive step-size method, a fixed step-size explicit method and two fixed step-size implicit-explicit methods.

While explicit solvers only require vector field evaluations at the cost of smaller time steps, implicit solvers have to solve a root-finding problem in each time step for a nonlinear model. An IMEX with singly DIRK (SDIRK) solver turns out to be the most efficient for

\(^9\) A closure is a pair of a function together with its scoped environment.
this class of models, simplifying the nonlinear problem to a linear problem solvable by a single matrix decomposition per trajectory. Due to the non-diagonal mass matrix, even in the case of a sufficiently stable and accurate explicit method, at least one linear problem per trajectory would have to be solved.

The solver interface is given by the following signature:

```python
solution = solver(discrete, scenario, config);
```

with the return value `solution`, being a structure, and the arguments `discrete` (model), `scenario`, and `config`(uration).

All provided fixed step-size solvers cache matrix decompositions. The initial steady-state is also cached, as is the QR decomposition used to compute it.

Even though the overall model (8) has a two dimensional structure and thereducers exploit this structure, the simulations itself can be performed on a lumped model (we omit parametrization here for ease of notation):

\[
\dot{E}x(t) = Ax(t) + Bu(t) + f(x(t), u(t)),
\]

\[
y(t) = Cx(t).
\]

### 5.3.1 Second-order adaptive implicit solver: generic

For validation purposes, the adaptive step-size solver for stiff systems `ode23s`\(^{10}\) included in MATLAB (and Octave) [125, Sect. 2.3], based on a modified second order Rosenbrock formula is used and encapsulated as a generic solver. Due to preferential performance demonstrated in [51], `ode23s` is preferred over alternatives such as `ode15s` (or `ode45`).

### 5.3.2 Fourth-order “classic” Runge-Kutta solver: rk4

Since in [104, 105], the fourth-order explicit Runge-Kutta (RK) method [83] is employed, `morgen` provides it, too. This method is strong stability preserving [47], however it is not SSP-optimal and it works only for small time-steps.

### 5.3.3 First-order implicit-explicit solver: imex1

The lumped gas network model (22) can be split into a linear and a nonlinear part, of which the linear part is numerically stiff and hence should be solved with an implicit solver, while an explicit solver is preferred for the nonlinear part as to avoid solving a root-finding (optimization) problem in each time step.

An IMEX method allows this separate treatment of operators and is thus suitable for this hyperbolic and nonlinear system. Combining the first-order explicit Euler’s method with the first order implicit Euler’s method yields the first order IMEX method:

\[
Eh^{-1}(x_{k+1} - x_k) = (1 - \gamma)Ax_k + \gamma Ax_{k+1} + Bu_k + f(x_k, u_k)
\]

\[
\Rightarrow \quad E\dot{x}_{k+1} - \gamma hAx_{k+1} = E\dot{x}_k + (1 - \gamma)hAx_k + hBu_k + hf(x_k, u_k)
\]

\[
\Rightarrow \quad x_{k+1} = x_k + (E - \gamma hA)^{-1}h(Ax_k + Bu_k + f(x_k, u_k)),
\]

with the associated Butcher tableaus Table 2. Even though this IMEX method is not a Runge-Kutta method [5], it was successfully applied to gas network models in [52] with a relaxation parameter set to $\gamma = 1$.

\(^{10}\)https://mathworks.com/help/matlab/ref/ode23s.html (accessed: 2020-11-18)
Table 2  Butcher tableaus for the 1st-order IMEX method (Sect. 5.3.3)

| Explicit: | Implicit: |
|-----------|-----------|
| 0 0 1     | 1 1 1     |

Table 3  Parameter choices for the implicit 2nd-order IMEX-RK component

| Parameter Property Source |
|---------------------------|
| \( \lambda = 0.24 \) Efficient \([82]\) |
| \( \lambda \geq \frac{1}{4} \) A-stable \([76]\) |
| \( \lambda = \frac{1}{2} - \frac{\sqrt{2}}{2} = 1 - \frac{1}{\sqrt{2}} \) L-Stable \([109]\) |
| \( \lambda = \frac{1}{2} \) Stiffly accurate morgen, cf. \([76, \text{Sect. 5.1}]\) |
| \( \lambda = \frac{3 + \sqrt{3}}{6} = \frac{1}{2} + \frac{1}{\sqrt{12}} \) Passive \([101], \text{cf. [13, Sect. 3.2.2]}\) |

Table 4  Butcher tableaus for the 2nd-order IMEX-RK method (Sect. 5.3.4)

| Explicit: | Implicit: |
|-----------|-----------|
| 0 0 0 1 \( \frac{1}{2} \) \( \frac{1}{2} \) | \( \lambda \) \( \lambda \) 0 \( 1 - \lambda \) \( 1 - 2\lambda \) \( \lambda \) \( \frac{1}{2} \) \( \frac{1}{2} \) |

5.3.4 Second-order implicit-explicit Runge-Kutta solver: imex2

A second-order (two-stage) IMEX Runge-Kutta solver is provided, based on the combination of a second-order explicit SSP Runge-Kutta method \([47]\), and a second-order DIRK method. Following \([109]\), such an IMEX-SSP2(2,2,2) method with relaxation \( \gamma \) is given by:

\[
\begin{align*}
z_1 &= (E - h\gamma \lambda A)^{-1} Ex_k, \\
z_2 &= (E - h\gamma \lambda A)^{-1} (Ex_k + hBu_k + hf(x_k, u_k) + h\gamma (1 - 2\lambda)Az_1), \\
x_{k+1} &= x_k + E^{-1} \frac{H}{2} (Bu_k + f(x_k, u_k) + \gamma Az_1 + Bu_{k+1} + f(z_1, u_{k+1}) + \gamma Az_2).
\end{align*}
\]

The explicit component of this IMEX method is SSP-optimal \([47]\), while depending on the choice for the free parameter \( \lambda \), different properties of the implicit component can be achieved (see Table 3). Practically, we found \( \lambda = \frac{1}{2} \), making the implicit part SDIRK and stiffly accurate, to work best. Additionally, we would like to highlight passive Runge-Kutta methods \([101]\), specifically PDIRK (passive DIRK), as implicit IMEX component, which have various desirable stability and conservation properties \([43]\).

The associated Butcher tableaus are given in Table 4.

In our experiments, the first-order IMEX method Sect. 5.3.3 allowed larger time-steps and exhibited less numerical oscillations or artifacts compared to the second-order IMEX-RK methods. The generic (adaptive) method Sect. 5.3.1 also solves sufficiently accurate, but takes longer to compute. Thus, by default, we recommend the first-order IMEX integrator for gas network simulations.

5.4 Reductors

The reductor module provides methods that compute structured\(^{11}\) (struct.) projectors for a given discretization. These projectors can be stored on disk for reuse. Currently, the reductors, described in Sect. 4 are included:

\(^{11}\)Originally, all methods were also tested in an unstructured variant, but showed insufficient accuracy.
Struct. proper orthogonal decomposition
struct. empirical dominant subspaces
Struct. balanced proper orthogonal decomposition
struct. empirical balanced truncation
Struct. goal-oriented proper orthogonal decomposition
struct. empirical balanced gains
Struct. dynamic mode decomposition Galerkin.

Each reductor variant has a suffix characterizing the employed empirical Gramians: _r for only reachability, _ro for reachability and observability, _wx for cross, and _wz for non-symmetric cross Gramian. For each reductor utilizing observability information (this includes the cross Gramians), a linear variant using the dual system is available, and signified with the additional suffix _l.

The reductors have the interface:

\[ \text{[proj, name]} = \text{reductor(solver, discrete, scenario, config)}; \]

returning a (cell) array of projectors with maximum configured column rank, as well as the reductor’s full name. These projectors specific to model and solver defining a ROM are then stored in a .rom file.

5.4.1 Empirical Gramian framework
The compute back-end for the model reduction methods is emgr – empirical Gramian framework [66]; currently in version 5.9 [68]. This (open-source) Octave and MATLAB toolbox computes the empirical Gramians, which are essential to construct the reduced order models via the Gramian-based model reduction methods from Sect. 4, including the structured DMD-Galerkin method.

5.5 Tests
A test is a script defining an experiment by specifying network, scenario, model, solver and reductors. The tests component is a collection of test scripts probing primarily model reduction for various networks. A typical test contains two calls to the main morgen function. The first call computes the reduced order model (offline phase):

morgen(network, training_scenario, model, solver, reductors);

which computes a ROM from a short, generic, steady-state training_scenario. The projectors defining the reduced order models are then stored in rom_files. The second call tests the reduced order model(s) on a longer test_scenario (online phase):

morgen(network, test_scenario, model, solver, rom_files);

Generally, a model reduction method can also be tested in a single call, disregarding that scenario’s boundary value time series, yet for productive use, a reduced order model is constructed once (first call), and then employed for many different scenarios (second call).

Included in morgen are two types of tests: First, tests prefixed with “sim_” only simulate the test scenarios, second, tests prefixed with “mor_” compute the reduced order models using training scenarios, and benchmark these ROMs on the test scenarios.

6 Numerical experiments
In the following, we present three sets of numerical experiments, with the purpose of demonstrating the reducibility of gas network models via the data-driven, parametric, system-theoretic model order reduction algorithms from Sect. 4, and illustrating the capabilities of the morgen framework summarized in Sect. 5. The first set uses a pipeline...
“network”, which is interesting in the context of model reduction, while the second set tests an academic toy network as a sanity check and a simple functionality test. Lastly, a realistic gas network topology is evaluated.

We note that various further networks are included for testing in morgen; among others: the Canvey-Leeds network [55, 79], the Belgium transport network [31, 95] and a part of the Fermaca network [114]. A synthetic pipeline model and associated simulation results were provided by the PSI Software AG for validation of morgen against the commercial PSIganesi\textsuperscript{12} solver.

6.1 Workflow
For each of the numerical tests, the same workflow is employed, which is composed of a training phase (offline phase), in that the ROMs are computed, using a generic test scenario, with a (virtual) time horizon of 1 h, and boundary value input functions typical for system identification, i.e. Dirac impulse, step signal, random-binary signal or Gaussian noise [100, Ch. 16]. In the test phase (online phase), the ROMs are tested on scenarios with a (virtual) time horizon of 24 h [27, 90], (starting at 6 am [40]). In addition to shorter offline phases, this difference in training and test time horizons emphasizes generality of the ROMs.

To verify models and solvers [107] this offline/online procedure is performed for all combinations of:

Models: ode_mid, ode_end;
Solvers: imex1, imex2;
Reductors: pod_r, eds_ro, eds_wx, eds_wz,
bpod_ro, ebt_ro, ebt_wx, ebt_wz,
gpod_ro, ebg_ro, ebg_wx, ebg_wz,
dmd_r;

whereas the port-Hamiltonian ode_end model is tested with the nonlinear as well as the linear reductor variant (_l suffix) if available, while the ode_mid model is only tested with the nonlinear reductor variant.

The models are specialized by the Schiffrinson friction, and the AGA88 compressibility factor formula. We excluded the generic and rk4 solvers in this comparison as they are too slow or too fragile, respectively. Yet, the test scenario visualizations in Fig. 4(a) and Fig. 6(a) are computed by the generic solver.

For the parametric model reduction, the temperature range for training and testing is set to \([0^\circ, 15^\circ]\)C, while the specific gas constant range is chosen as \([500, 600]\text{JkgK}\). During training, samples from the parameter space are drawn from a sparse grid, whereas for the tests, parameters are drawn from a uniform random distribution. For either test and training, five parameters are sampled. The input perturbations for the steady-state training scenario are selected to be a step function, which heuristically works well for hyperbolic systems [49].

\textsuperscript{12}https://www.psigasandpipelines.com
The reduced order models are compared via the approximate, discrete, parametric, \((L_2 \otimes L_2)\)-norm of the output error [67]:

\[
\| y - \tilde{y} \|_{L_2 \otimes L_2} \approx \sqrt{\sum_{\theta_h \in \Theta_1} \Delta t \left\| \text{vec}(y_h(\theta_h) - \tilde{y}_h(\theta_h)) \right\|^2_2},
\]

for a finite sample \(\Theta_1\) of the parameter space \(\Theta\) and discrete output samples \(y_h(\theta_h), \tilde{y}_h(\theta_h)\). This energy norm is chosen, since all methods are at least related to an energy-based method. However, morgen can also provide the errors in the approximate parametric \((L_k \otimes L_\ell)\) parameter-space-state-space norms for \(k \in \{1, 2, \infty\}\), \(\ell \in \{0, 1, 2, \infty\}\), cf. [50]. Note, that due to nonlinearity of the considered models, and the averaging nature of the norm, a monotonic error decay cannot be expected. To enhance comparability of the results, also the reducibility measure MORscore [67] for each experiment is computed. The MORscore for a certain method and model is essentially the area above the method’s error graph in the relative error plot such as Fig. 4(c).

Lastly, we note that the following numerical experiments are conducted using a computer with an AMD Ryzen 4500U @ 2.3Ghz hexa-core processor and 16 GiB memory running MATLAB 2021a on Ubuntu 20.04 Linux.

6.2 Yamal-Europe pipeline

First, a pipeline is tested, which is an interesting test case, since the trivial topology (Fig. 3) comprises little redundancy, hence pipelines are a useful benchmark for model reduction methods.

The Yamal-Europe pipeline connects gas fields in Russia with western Europe. A section of this pipeline was also benchmarked in [19, 24, 25, 107], from which the technical properties and test scenario are taken. The considered pipeline section is 363 km long, has a diameter of 1.422 m, no (reported) inclination, and a pipe roughness of 0.01 mm. A steady-state, used as initial state, is set by a supply pressure of 84 bar and demand mass-flux of 46.3 kg/s.

The semi-discrete nonlinear state-space system has two inputs and outputs as well as 908 states; and a time step width of 20 s is used. The employed test scenario is taken from [25], compressed to 24 h, and shown in Fig. 4(a), the associated model reduction errors are given in Fig. 4(c), Fig. 4(d), Fig. 4(e), and Fig. 4(f), for up to reduced order 150, while the resulting MORscores are listed in Table 5.

Generally, the choice of solver is more relevant than the choice of model: while the MORscores for different models but same solver are similar, for the same model but different solver, they are significantly dissimilar. Also, the tested balancing (Petrov-Galerkin) methods perform worse than the Galerkin methods.

\[13^{\text{th}}\] See also https://en.wikipedia.org/wiki/Yamal-Europe_pipeline.
For both models, and the first-order IMEX solver, the structured empirical dominant subspaces methods perform best, followed (closely) by the DMD-Galerkin and (goal-oriented) POD method; then, among the balancing methods, the balanced POD and cross-Gramian-based variants. The most overall accurate method is the cross-Gramian-based dominant subspaces method.

For both models, in combination with the second-order IMEX-RK solver, the structured POD and goal-oriented methods lead, followed by the DMD-Galerkin reductor. For both solvers, the endpoint model performs better than the midpoint model. In case of the port-

---

**Figure 4** Visualization of the test scenario, and model reduction errors of the tested ROMs for the Yamal-Europe pipeline from Sect. 6.2

(a) Boundary values and quantities of interest plots for the test scenario.

(b) Common legend for the error plots Fig. 4c, Fig. 4d, Fig. 4e, Fig. 4f.

(c) $L_2 \otimes L_2$ error between ROM and FOM for the *ode_mid* model, *imex2* solver, and nonlinear reductors versus reduced order.

(d) $L_2 \otimes L_2$ error between ROM and FOM for the *ode_end* model, *imex2* solver, and linear reductors versus reduced order.

(e) $L_2 \otimes L_2$ error between ROM and FOM for the *ode_mid* model, *imex2* solver, and nonlinear reductors versus reduced order.

(f) $L_2 \otimes L_2$ error between ROM and FOM for the *ode_end* model, *imex2* solver, and linear reductors versus reduced order.
Hamiltonian endpoint model, the linear Galerkin reductors are about as accurate as the nonlinear Galerkin reductors. We note that the cross-Gramian-based dominant subspaces methods produce the lowest errors, and since for the linear reductors used in combination with the endpoint model, the dominant subspaces methods are as efficient as the purely reachability-based DMD-Galerkin, and (goal-oriented) POD methods.

Interestingly, while the second-order IMEX-RK solver is better suited for simulations of the full order model, in terms of data-driven model reduction and/or reduced order model simulation it is significantly worse than the first-order IMEX solver. This is also demonstrated in the subsequent experiments.

### 6.3 MORGEN network

The second set of tests encompasses a synthetic network for testing morgen’s capabilities. This “MORGEN” network, with topology as in Fig. 5, tests the interaction of simplified compressors from Sect. 2.5 for various network features, such as cycles, multiple supply and demand nodes, and is in the spirit of a test network from [39].

Specifically, six sub-networks (in the shape of letters) are connected, the second and third sub-network contain a cycle, a compressor connects the third and fourth sub-network, and the fourth and fifth sub-network contain additional supply and demand nodes. The edges vary in length between 20 km and 60 km, while the diameter and roughness are consistently 1 m and 0.01 mm, respectively. A steady-state, used as initial state, is set by supply (and discharge) pressures of 50 bar at both supply nodes and the compressor, and demand mass-fluxes of $30 \frac{kg}{s}$ at all demand nodes.

| Table 5 | MORSCORES $\mu(150, \varepsilon_{mach}(16))$ in the $L_2 \otimes L_2$ error norm for the “Yamal-Europe” pipeline network from Sect. 6.2; * notes linear reductors |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
|                 | ode_mid | ode_end | ode_end | ode_mid | ode_end | ode_end | ode_mid | ode_end | ode_end |
|-----------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| pod_r            | 0.40    | 0.40    | 0.19    | 0.22    |         |         |         |         |         |
| eds_ro           | 0.52    | 0.54    | 0.51    | 0.07    | 0.09    | 0.04    |         |         |         |
| eds_wx           | 0.51    | 0.55    | 0.57    | 0.06    | 0.13    | 0.20    |         |         |         |
| bpod_ro          | 0.19    | 0.28    | 0.14    | 0.03    | 0.08    | 0.11    |         |         |         |
| ebt_ro           | 0.05    | 0.06    | 0.03    | 0.07    | 0.12    | 0.17    |         |         |         |
| ebt_wx           | 0.17    | 0.11    | 0.10    | 0.00    | 0.00    | 0.00    |         |         |         |
| bpts_wz          | 0.24    | 0.30    | 0.15    | 0.00    | 0.00    | 0.00    |         |         |         |
| gopod_r          | 0.40    | 0.41    | 0.08    | 0.19    |         |         |         |         |         |
| ebg_ro           | 0.05    | 0.11    | 0.02    | 0.05    | 0.11    | 0.15    |         |         |         |
| ebg_wx           | 0.08    | 0.23    | 0.14    | 0.00    | 0.00    | 0.00    |         |         |         |
| ebg_wz           | 0.17    | 0.29    | 0.11    | 0.00    | 0.00    | 0.00    |         |         |         |
| dmd_r            | 0.50    | 0.53    | 0.08    | 0.15    |         |         |         |         |         |

Figure 5: MORGEN network topology
In semi-discrete form, the nonlinear state-space system features six inputs and outputs as well as 901 states; and a time discretization with 60 s time steps. The employed 24 h test scenario is made from hourly standard load profiles [60] and shown in Fig. 6(a), the associated model reduction errors are given in Fig. 6(c), Fig. 6(d), Fig. 6(e), and Fig. 6(f), for up to reduced order 200, while the resulting MORscores are listed in Table 6.
Table 6  MORscores $\mu(200, \nu_{\text{mach}(16)})$ in the $L_2 \otimes L_2$ error norm for the “MORGEN” test network from Sect. 6.3; * notes linear reductors

|                      | ode_mid | ode_end | ode_end | ode_mid | ode_end | ode_end |
|----------------------|---------|---------|---------|---------|---------|---------|
|                      | imex_1  | imex_1  | imex_1* | imex_2  | imex_2  | imex_2* |
| pod_r                | 0.16    | 0.16    | 0.02    | 0.04    | 0.04    | 0.08    |
| eds_ro               | 0.20    | 0.20    | 0.32    | 0.04    | 0.03    | 0.06    |
| eds_wx               | 0.09    | 0.10    | 0.15    | 0.01    | 0.02    | 0.09    |
| eds_wz               | 0.04    | 0.09    | 0.13    | 0.02    | 0.04    | 0.06    |
| bpod_ro              | 0.13    | 0.13    | 0.03    | 0.00    | 0.02    | 0.00    |
| ebt_ro               | 0.00    | 0.00    | 0.00    | 0.01    | 0.02    | 0.00    |
| ebt_wx               | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| ebt_wz               | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| gopod_r              | 0.14    | 0.14    | 0.01    | 0.01    | 0.01    | 0.00    |
| ebg_ro               | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| ebg_wx               | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| ebg_wz               | 0.00    | 0.00    | 0.01    | 0.00    | 0.00    | 0.00    |
| dmd_r                | 0.16    | 0.21    | 0.06    | 0.05    | 0.05    | 0.00    |

Again for this comparison, the choice of solver is more relevant than the choice of model, yet the MORscores are much lower, due to the complexities (cycles, compressor, multiple demands) of the network.

Both models in conjunction with the first-order IMEX solver only produce workable results with Galerkin methods. The endpoint ROMs are again more accurate than the midpoint ROMs. Notably, the linear reachability-and-observability dominant subspaces method for the endpoint model, is leading the MORscores.

The ROMs for both models with the second-order IMEX-RK solver perform worse, with the exception of the balanced truncation ebt_ro and balanced gains ebg_ro variants. However, the second-order IMEX-RK solver related ROMs are of no practical use due to the high(er) error.

Overall, the (linear) eds_ro dominant subspaces reductor produces the lowest error, followed by the DMD-Galerkin, (goal-oriented) POD, and cross-Gramian-based dominant subspaces methods. As for the pipeline, the endpoint model is better suited for model reduction, while the first-order IMEX solver results in significantly more accurate ROMs than the second-order IMEX-RK solver.

6.4 GasLib network

Lastly, a network topology derived from real-life is tested. The GasLib-134v2 network [122], modeling a part of the Greek natural gas transport system, is overlayed on a map of Greece in Fig. 7. The network has a total length of 1412 km and features a compressor. A steady-state, used as initial state, is set by supply (and discharge) pressures of 80 bar at supply nodes and the compressor, and demand mass-fluxes up to $16\frac{kg}{s}$ at all demand nodes.

In semi-discrete form, the nonlinear state-space system has 48 inputs and outputs as well as 2682 states; and 30 s time steps are employed. For testing, a random (24 h) load profile is generated, by adding samples from a scaled uniform random distribution to the steady-state, shown in Fig. 8(a), the associated model reduction errors are given in $\mu(200, \nu_{\text{mach}(16)})$. morgen can generate such profiles for all included networks.
As before, the choice of solver is more relevant than the choice of model. Challenges in this network, beyond the compressor, are the high number of boundary nodes, which are predominantly demand nodes \((N_d = 45)\).

First, we note that only Galerkin methods produce consistently stable ROMs. Furthermore, in comparison with the previous experiments, the dominant subspace methods perform worse, and all variants based on reachability and observability Gramians perform relatively better. The endpoint model seems to be better suited for the tested model reduction methods than the midpoint model. And as for the other experiments, the first order IMEX solver outmatches the second order IMEX-RK solver.

Considering all experiments, the DMD-Galerkin method performs best in terms of MOR score, accuracy and efficiency, followed by the dominant subspaces methods. We also note that the purely reachability-based as well as the linear reductors exploiting the port-Hamiltonian structure are the most efficient. Surprisingly, Galerkin methods perform better than the tested Petrov-Galerkin methods in terms of accuracy and stability, while in an unstructured, non-parametric, linearized setting all tested Petrov-Galerkin methods would be stability preserving. Yet, structured balancing methods are explicitly not guaranteed to be stability-preserving \([120, 136]\).

With regard to the computational complexity of the offline and online phase, we reiterate, that due to the absence of hyper-reduction, the online runtimes are not competitive (see Sect. 3.3), thus, we focus on the offline phase. Yet, due to the practical reducibility of the state-space dimension by more than one order of magnitude in the numerical experiments using the first order IMEX solver, a considerable speed-up is to be expected.

For the tested data-driven (time-domain) model reduction methods, the number of vector-field evaluations, or relatedly, the number of simulated trajectories measures the complexity, as these constitute their principal fraction. The empirical reachability Gramian requires \(N_s + N_d\) (number of ports) trajectories. The empirical observability Gramian requires \(N_p + N_q\) (number of states) trajectories. The empirical cross Gramian requires \(N_s + N_d + N_p + N_q\) trajectories, and the linear empirical cross Gramian requires \(2(N_s + N_d)\) trajectories. For the tested reductors this amounts to \(N_s + N_q\) trajectories for the POD, goal-oriented POD, and DMD-Galerkin method, while the port-Hamiltonian variants of the dominant subspaces, balanced POD, balanced truncation and balanced gains
methods need \(2(N_s + N_d)\) trajectories, and their non-port-Hamiltonian variants require \(N_s + N_d + N_p + N_q\) trajectories.

These predicted complexities are reflected in the offline runtimes, when computed sequentially. As the computation of trajectories is embarrassingly parallel, all trajectories are however computable simultaneously. Nonetheless, the complexities of the reachability-Gramian-only and port-Hamiltonian reductors are independent from the discretization, and thus most relevant for large-scale gas networks.
Table 7 MORs $\mu(250, \epsilon_{mach(16)})$ in the $L_2 \otimes L_2$ error norm for the "GasLib-134v2" benchmark network from Sect. 6.4; * notes linear reductors.

|            | ode_mid | ode_end | ode_end | ode_mid | ode_end | ode_end |
|------------|---------|---------|---------|---------|---------|---------|
| pod_ro     | 0.14    | 0.14    | 0.02    | 0.11    |
| eds_ro     | 0.04    | 0.07    | 0.16    | 0.13    | 0.12    | 0.11    |
| eds_wx     | 0.07    | 0.07    | 0.12    | 0.08    | 0.12    | 0.11    |
| eds_wz     | 0.09    | 0.08    | 0.12    | 0.11    | 0.12    | 0.11    |
| bpod_ro    | 0.06    | 0.06    | 0.00    | 0.04    | 0.05    | 0.01    |
| ebt_ro     | 0.05    | 0.05    | 0.00    | 0.05    | 0.07    | 0.02    |
| ebt_wx     | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| ebt_wz     | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| gopod_ro   | 0.09    | 0.10    | 0.01    | 0.07    |
| ebq_ro     | 0.02    | 0.03    | 0.00    | 0.00    | 0.02    | 0.01    |
| ebq_wx     | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| ebq_wz     | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    | 0.00    |
| dmd_r      | 0.20    | 0.19    | 0.12    | 0.13    |

7 Outlook

The next stage in the development of morgen involves testing larger real-life networks, such as the deliverable of the SciGRID_gas\textsuperscript{15} (Open Source Model of the European Gas Network) project. Yet, various further venues of linked modeling and model reduction questions are still not covered by morgen.

In terms of model reduction, an interesting issue are intraday switchable valves, which change the topology of the gas network graph and likely require to extend the utilized model reduction methods towards these switched systems.

Another interesting question in need of further investigation is the minimal time horizon of the training phase. A lower bound is the time step times the longest path from a supply to a demand node, but this is likely not sufficient.

Besides an additional hyper-reduction module (Sect. 3.3) post-processing the reduced order models, a decoupler module pre-processing (DAE) models as described in [11, 12] is projected.

Also as detailed in Sect. 3.4, the pipe roughness is a relevant parameter for (transient) simulations [130], yet the entailing high-dimensional parameter space, due to the locally differing roughness and attrition rates, would have to be treated, too. This in turn would raise the question for combined state and parameter reduction [65], and is postponed to future investigations.

Finally, using a tunable efficiency factor [105, 107, 111] that scales the model’s friction term, can be used to tune the models to match real data.

8 Conclusions

In more than half a century of computational transient gas network simulation research and industrial use, morgen seems to be the first open-source platform covering modeling, simulation, and model order reduction of gas (and energy) networks. The target applications for morgen are finding the best model reduction method or best reduced order model for a network by heuristic comparison, as well as comparing model-solver-reductor simulation ensembles.

\textsuperscript{15}https://www.gas.scigrid.de
From a mathematical point of view, a next generation gas network simulation stack should consist of a (endpoint) port-Hamiltonian model, a (first-order) IMEX solver, and a block-diagonal Galerkin projection reductor, which is confirmed by the numerical results. This results in the following heuristically determined but theoretically explainable recommended combination: The endpoint model together with the first-order IMEX solver, and a Galerkin reductor, specifically a structured dominant subspaces or structured DMD-Galerkin, exhibiting the highest MOR scores in the numerical experiments. The performance of structured balanced truncation and the related structured balanced gains may be improved in terms of stability(-preservation) either by a variant of the technique [88], a stabilizing inner product [124], an (energy-)stable inner product [78], or an optimization-based post-processing as in [21].

Lastly, we invite researchers, engineers and users to provide their reductors, solvers, networks and scenarios for expansion and testing with morgen for a broader view of this comparison.

Appendix: model fact sheet

**Basis:** Euler equations for cylindrical pipes

**Assumptions**
- Long pipes: One spatial dimension
- Kinetic term: Removed due to slow subsonic velocities
- Boundary values: Low-frequency (sum of) step functions

**Simplifications**
- Temperature: Isothermal (temperature is parameter)
- Gas composition: Constant global (specific gas constant is parameter)
- Compressibility: Constant global (derived from steady-state)
- Compressors: Affine / Additive

**Modularization**
- Friction: Hofer | Nikuradse | Altshul | Schifrinson | PMT1025 | IGT
- Compressibility: Ideal | DVGW-G-2000 | AGA88 | Papay

**Discretization**
- Spatial: 1st order upwind finite differences
- Temporal: RK-4 | IMEX-1 | IMEX-RK-2 | Rosenbrock-2

Acknowledgements
Not applicable.

Funding
This work is supported by the German Federal Ministry for Economic Affairs and Energy, in the joint project: “MathEnergy – Mathematical Key Technologies for Evolving Energy Grids”, sub-project: Model Order Reduction (Grant number: 03240198), and by the National Science Foundation under Grant number: DMS-1439786 while the author was in residence at the Institute for Computational and Experimental Research in Mathematics in Providence, RI, during the “Model and dimension reduction in uncertain and dynamic systems” program. Open Access funding enabled and organized by Projekt DEAL.

Abbreviations
- CFL: Courant-Friedrichs-Levy
- CSV: Comma Separated Value
- DAE: Differential Algebraic Equation
- DEIM: Discrete Empirical Interpolation Method
- DIRK: Diagonally Implicit RK
DMD  Dynamic Mode Decomposition
DMDc  DMD with Control
DSPMR  Dominant Subspace Projection Model Reduction
EVD  Eigenvalue Decomposition
IMEX  Implicit-Explicit
MOR  Model Order Reduction
ODE  Ordinary Differential Equation
PDAE  Partial DAE
PDE  Partial Differential Equation
PDIRK  Passive DIRK
POD  Proper Orthogonal Decomposition
RK  Runge-Kutta
ROM  Reduced Order Model
SDIRK  Singly DIRK
SPMOR  Structure-Preserving Model Order Reduction
SRSN  System with Repeated Scalar Nonlinearities
SSP  Strong Stability Preserving
SVD  Singular Value Decomposition
tEVD  truncated EVD
tSVD  truncated SVD

Availability of data and materials
The Matlab language source code of the morgen platform 1.0 is licensed under BSD-2-CLAUSE LICENSE, can be obtained from: https://doi.org/10.5281/zenodo.5012357 and is authored by: C. HIME and S. GRUNDEL.

Competing interests
The authors declare that they have no competing interests.

Authors’ contributions
CH has authored the manuscript, developed the morgen platform, and set up the numerical experiments. SG contributed to the writing of the paper and co-developed the morgen platform. PB was responsible for the conception of the project and contributed to the writing of the paper. All authors read and approved the final manuscript.

Publisher’s Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 25 November 2020 Accepted: 27 June 2021 Published online: 18 July 2021

References
1. Aalto H. Model reduction for natural gas pipeline systems. In: IFAC proceedings volumes (12th IFAC symposium on large scale systems: theory and applications). vol. 43. 2010. p. 468–73. https://doi.org/10.3182/20100712-3-FF-2020.00077.
2. Alla A, Kutz JN. Nonlinear model order reduction via dynamic mode decomposition. SIAM J Sci Comput. 2017;39(5):8778–96. https://doi.org/10.1137/16M1059308.
3. Altmann R, Zimmer C. Time discretization of nonlinear hyperbolic systems on networks. Proc Appl Math Mech. 2019;19:e201900057. https://doi.org/10.1002/pamm.201900057.
4. Antoulas AC. An overview of approximation methods for large-scale dynamical systems. Annu Rev Control. 2005;29:181–90. https://doi.org/10.1016/j.arcontrol.2005.08.002.
5. Ascher UM, Ruuth SJ, Spiteri RJ. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. Appl Numer Math. 1997;25(2–3):151–67. https://doi.org/10.1016/S0168-9274(97)00056-1.
6. Ashouri F. An expert system for predicting gas demand: a case study. Omega. 1993;21(3):307–17. https://doi.org/10.1016/0305-0483(93)90088-3.
7. Azevedo-Perdicoúlis TP, Jank G. Modelling aspects of describing gas networks through a DAE system. In: IFAC proceedings volume (3rd IFAC symposium on structure and control). vol. 40. 2007. p. 40–5. https://doi.org/10.3182/20071017-3-BR-2923.00007.
8. Azevedo-Perdicoúlis TP, Jank G, Lopes dos Santos P. The good behaviour of the gas network: boundary control, observability and stability. In: nDS’13: proceedings of the 8th international workshop on multidimensional systems. 2013. p. 87–92. https://ieeexplore.ieee.org/abstract/document/6623829.
9. Azevedo-Perdicoúlis TP, Jank G, Lopes dos Santos P. Modelling a gas pipeline as a repetitive process: controllability, observability and stability. Multidimens Syst Signal Process. 2015;26(4):967–84. https://doi.org/10.1007/s11045-015-0314-y.
10. Azevedo-Perdicoúlis TP, Perestrello F, Almeida R. A note on convergence of finite differences schema for gas network simulation. In: 22nd international conference on process control. 2019. p. 274–8. https://doi.org/10.1109/PC.2019.8815241.
11. Banagaaya N, Ali G, Grundel S, Benner P. Automatic decoupling and index-aware model-order reduction for nonlinear differential-algebraic equations. e-print 2002.09751, arXiv. 2020. cs.NA. URL: https://arxiv.org/abs/2002.09751.
12. Banagaaya N, Grundel S, Benner P. Index-aware MOR for gas transport networks with many supply inputs. In: IUTAM symposium on model order reduction of coupled systems. IUTAM bookseries. vol. 36. 2020. p. 191–207. https://doi.org/10.1007/978-3-030-21013-7_14.
13. Banda MK. Nonlinear hyperbolic systems of conservation laws and related applications. In: Banasiak J, Mokhtar-Kharroubi M, editors. Evolutionary equations with applications in natural sciences. Lecture notes in mathematics. vol. 2126. Berlin: Springer; 2015. p. 439–93. https://doi.org/10.1007/978-3-319-11322-7_9.
14. Baur U, Benner P, Haasdonk B, Himpe C, Martin I, Oihlbeger M. Comparison of methods for parametric model order reduction of time-dependent problems. In: Benner P, Cohen A, Ohlberger M, Willcox K, editors. Model reduction and approximation: theory and algorithms. Philadelphia: SIAM; 2017. p. 377–407. https://doi.org/10.1137/1.9781611974829.ch9.

15. Beattie C, Mehrmann V, Xu H, Zwart H. Linear port-Hamiltonian descriptor systems. Math Control Signals Syst. 2018;30:17. https://doi.org/10.1007/s00855-018-0229-3.

16. Beattie CA, Gugercin S. Structure-preserving model reduction for nonlinear port-Hamiltonian systems. In: 2011 50th IEEE conference on decision and control and European control conference. 2011. p. 6564–9. https://doi.org/10.1109/CDC.2011.6161504.

17. Benner P. System-theoretic methods for model reduction of large-scale systems: simulation, control, and inverse problems. In: ARGE SiN report (MATHMOD 2009 proceedings). vol. 35. 2009. p. 126–45. https://doi.org/10.11128/apep.35.

18. Benner P, Braukmüller M, Grundel S. A direct index 1 DAE model of gas networks. In: Keiper W, Milde A, Volkwein S, editors. Reduced-order modeling (ROM) for simulation and optimization. Cham: Springer; 2018. p. 99–119. https://doi.org/10.1007/978-3-319-75319-5_5.

19. Benner P, Grundel S, Himpe C, Huck C, Steudel T, Tischendorf C. Gas network benchmark models. In: Applications of differential-algebraic equations: examples and benchmarks, differential-algebraic equation forum. Cham: Springer; 2018. p. 171–97. https://doi.org/10.1007/11221_2018_5.

20. Benner P, Himpe C. Cross-Gramian-based dominant subspaces. Adv Comput Math. 2019;45(5):2533–53. https://doi.org/10.1007/s10444-019-09724-7.

21. Benner P, Himpe C, Mitchell T. On reduced input-output dynamic mode decomposition. Adv Comput Math. 2018;44(6):1821–44. https://doi.org/10.1007/s10444-018-9959-x.

22. Bermúdez A, Shahabi M. Finite element solution of isothermal gas flow in a network. J Comput Phys. 2019;396:616–52. https://doi.org/10.1016/j.jcp.2019.06.063.

23. Boyd S. Least squares and least norm solutions using Matlab. Additional notes to EE263: introduction to linear dynamical systems, 2008. http://ee263.stanford.edu/archive/fs_In_matlab.pdf.

24. Brouwer J, Gasser I, Herty M. Gaspipelinemodels revisited: model hierarchies, nonisothermal models, and damped wave propagation in transport networks. SIAM J Sci Comput. 2018;40(1):A331–65. https://doi.org/10.1137/17M1125303.

25. Chaczykowski M. Sensitivity of pipeline gas flow model to the selection of the equation of state. Chem Eng Res Des. 2009;87:1596–603. https://doi.org/10.1016/j.cherd.2009.06.008.

26. Chaturantabut S, Sorensen DC. Nonlinear model reduction via discrete-empirical interpolation. SIAM J Sci Comput. 2010;32(5):2737–64. https://doi.org/10.1137/090776649.

27. Chertkov M, Fisher M, Backhaus S, Bent R, Misra S. Pressure fluctuations in natural gas networks caused by gas-electric coupling. In: 48th Hawaii international conference on system sciences. 2015. p. 2378–47. https://doi.org/10.1109/HICSS.2015.330.

28. Chu Y-C, Glover K. Bounds of the induced norm and model reduction errors for systems with repeated scalar nonlinearities. IEEE Trans Autom Control. 1999;44(3):471–83. https://doi.org/10.1109/9.751342.

29. Clees T, Baldin A, Benner P, Grundel S, Himpe C, Klaassen B, Küsters F, Marheineke N, Nikitin L, Nikitin I, Pade J, Stahl N, Strohm C, Tischendorf C, Wissen A. MathEnergy – mathematical key technologies for evolving energy grids. In: Gottlich S, Herty M, Milde A, editors. Mathematical modeling, simulation and optimization for power engineering and management. Mathematics in industry: vol. 34. Cham: Springer; 2021. p. 233–62. https://doi.org/10.1007/978-3-030-62732-4_11.

30. Davidson A. Balanced systems and model reduction. Electron Lett. 1986;22(10):531–2. https://doi.org/10.1049/el:19860362.

31. De Wolf D, Smeers Y. The gas transmission problem solved by an extension of the simplex algorithm. Manag Sci. 2000;46(11):1454–65. http://www.jstor.org/stable/2661661.

32. Domschke P, Groß M, Hantke FM, Himpe C, Huck C, Herty M, Milde A, editors. Mathematical modeling, simulation and optimization for power engineering and management. Mathematics in industry: vol. 34. Cham: Springer; 2021. p. 233–62. https://doi.org/10.1007/978-3-030-62732-4_11.

33. Domschke P, Hantke FM, Hinter, Lang J, Tischendorf C. Modelling and simulation of gas networks: eine Übersicht. Technical Report 191. SFB/Transregio154.2017. https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/docId/191.

34. Dorao CA, Fernandino M. Simulation of transients in natural gas pipelines. J Nat Gas Sci Eng. 2011;3(1):349–55. https://doi.org/10.1016/j.jngse.2011.01.004.

35. Dorin BC, Toma-Leonida D. O. On modelling and simulating natural gas transmission systems (part I). Control Eng Appl Inform. 2008;10(3):27–36. http://www.ceai.srait.ro/index.php?journal=ceai&page=article&op=view&path%5B%5D=252&path%5B%5D=187.

36. Drmač Z, Gugercin S. A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions. SIAM J Sci Comput. 2016;38(2):A631–48. https://doi.org/10.1137/15M1019271.

37. Dymkou S, Leugering G, Jank G. Repetitive processes modelling of gas transport networks. In: 2007 international workshop on multidimensional (nD) systems. 2007. https://doi.org/10.1109/NDS.2007.4909556.

38. Egger H, Gieselmann J. Stability and asymptotic analysis for instationary gas transport via relative energy estimates. eprint 2012.14135, arXiv, 2020. math AP URL: https://arxiv.org/abs/2012.14135.

39. Egger H, Kugler T, Liljegren-Sailer B, Marheineke M, Mehrmann V. On structure-preserving model reduction for damped wave propagation in transport networks. SIAM J Sci Comput. 2018;40(1):A331–65. https://doi.org/10.1137/17M1253033.

40. Ehrlhardt K, Steinbach MC. Nonlinear optimization in gas networks. In: Modeling, simulation and optimization of technical processes. Berlin: Springer; 2005. p. 139–48. https://doi.org/10.1007/3-540-23710-8_11.

41. Ericsson SJ, Engel-Cox J, Arent DJ. Approaches for integrating renewable energy technologies in oil and gas operations. Technical Report 1491378. U.S. Department of Energy Office of Scientific and Technical Information; 2019. https://doi.org/10.2172/1491378.

42. Farzaneh-Gord M, Rafbafi HR. Unsteady natural gas flow within pipeline network, an analytical approach. J Nat Gas Sci Eng. 2016;28:379–409. https://doi.org/10.1016/j.jngse.2015.12.017.
43. Franken D, Ochs K. Numerical stability properties of passive Runge-Kutta methods. In: Proc. IEEE int. symp. Circuits syst. vol. 3. 2001. p. 473–6. https://doi.org/10.1109/SCAS.2001.921350.

44. Freund RW. SPRRIM: structure-preserving reduced-order interconnect macromodeling. In: Proc. Int. conf. on computer aided design (ICCAD). Los Alamitos: IEEE Computer Society Press; 2004. p. 80–7. https://doi.org/10.1109/ICCAD.2004.1382547.

45. Fugenschuh A, Geilberger B, Gollmer R, Morsi B, Pfetsch ME, Rövekamp J, Schmidt M, Sperdeldkens K, Steinbach MC. Physical and technical fundamentals of gas networks. MOS-SIAM series on optimization. Philadelphia: SIAM; 2015. p. 17–43. https://doi.org/10.1137/17611973693.9.

46. Goldwater MH, Rogers K, Turnbull DK. The PAN network analysis system – its development and use. Institution of Gas Engineers Communications. 1976;19:1–24.

47. Gottlieb S, Lehoucq RB, Tadmor E. Strong stability-preserving higher-order time discretization methods. SIAM Rev. 2001;43(1):89–112. https://doi.org/10.1137/S003614450036757X.

48. Grundel S, Herty M. Hyperbolic discretization via Riemann invariants: e-prints 2005.12158, arXiv, 2020. math.NA. URL: https://arxiv.org/abs/2005.12158.

49. Grundel S, Himpe C, Saak J. On empirical system Gramians. Proc. Appl Math Mech. 2019;19(1):e201900006. https://doi.org/10.1002/PAMM.201900006.

50. Grundel S, Horning N, Kläsens B, Benner P, Clees T. Computing surrogates for gas network simulation using model order reduction. In: Kooij S, Leifsson L, editors. Surrogate-based modeling and optimization. New York: Springer; 2013. p. 189–212. https://doi.org/10.1007/978-1-4614-7551-4_9.

51. Grundel S, Horning N, Roggenfors S. Numerical aspects of model order reduction for gas transportation networks. In: Kooij S, Leifsson L, Yang X-S, editors. Simulation-driven modeling and optimization. Berlin: Springer; 2016. p. 1–28. https://doi.org/10.1007/978-3-319-27517-8_1.

52. Grundel S, Jansen L. Efficient simulation of transient gas networks using IMEX integration schemes and MOR methods. In: 54th IEEE conference on decision and control (CDC). 2015. p. 4579–84. https://doi.org/10.1109/CDC.2015.7402934.

53. Grundel S, Jansen L, Horning N, Clees T, Tischendorf C, Benner P. Model order reduction of differential algebraic equations arising from the simulation of gas transport networks. In: Progress in differential-algebraic equations, differential-algebraic equations forum. Berlin: Springer; 2014. p. 183–205. https://doi.org/10.1007/978-3-662-44926-4_9.

54. Gugat M, Herty M, Otto K, Rantzer A. Control of gas networks: a survey. Surro. Model. Optim. 2019;2019(4):1–28. https://doi.org/10.1002/PAMM.201900006.

55. Herren-González A, De La Cruz JM, De Andrés-Toro B, Risco-Martín JL. Modeling and simulation of gas distribution systems using empirical Gramian framework (version 5.9). https://gramian.de. 2021.

56. Himpe C. emgr—EMpirical GRamian framework (version 5.9). https://gramian.de. 2021.

57. Himpe C, Grundel S, Benner P. Efficient gas network simulations. In: Maaß P, Küfer K-H, Schulz V, editors. German success stories in industrial mathematics. Mathematics in industry. Cham: Springer; 2021. https://doi.org/10.1007/978-3-030-81455-7_7. Accepted.

58. Himpe C, Ohlberger M. Cross-Gramian based combined state and parameter reduction for large-scale control systems. Math Prob Eng. 2014;2014:843869. https://doi.org/10.1155/2014/843869.

59. Himpe C, Ohlberger M. The empirical cross Gramian for parametrized nonlinear systems. In: IFAC-PapersOnLine (proceedings of the 8th Vienna international conference on mathematical modelling). vol. 48. 2015. p. 727–8. https://doi.org/10.1016/j.ifacol.2015.05.163.
72. Himpe C, Othberger M. A note on the cross Gramian for non-symmetric systems. Syst Sci Control Eng. 2016;1(1):199–208. https://doi.org/10.1007/21642583.2016.1215273.

73. Hirsch SM, Harris KD, Kutz JN. Brunton BW. Centering data improves the dynamic mode decomposition. SIAM J Appl Dyn Syst. 2020;19(3):1920–55. https://doi.org/10.1137/19M1289881.

74. Hülk L, Müller B, Glauser M. Transparency, reproducibility, and quality of energy system analyses -- a process to improve scientific work. Energy Strategy Rev. 2018;22:264–9. https://doi.org/10.1016/j.esr.2018.08.014.

75. Ionescu TC, Fujimoto K, Scherer JMA. Singular value analysis of nonlinear symmetric systems. IEEE Trans Autom Control. 2011;56(9):2073–86. https://doi.org/10.1109/TAC.2011.2126630.

76. Izio G, Jackiewicz Z. Highly stable implicit-explicit Runge-Kutta methods. Appl Numer Math. 2017;113:71–92. https://doi.org/10.1016/j.apnum.2016.10.018.

77. Jiang Y-L, Qi Z-Z, Yang P. Model order reduction of linear systems via the cross Gramian and SVD. IEEE Trans Circuits Syst II, Express Briefs. 2019;66(3):422–6 https://doi.org/10.1109/TCSII.2018.2864115.

78. Kalashnikova I, Barone ME, Anurajetasan S, van Bloemen Waanders BG. Construction of energy-stable projection-based reduced order models. Appl Math Comput. 2014;249:569–96. https://doi.org/10.1016/j.amc.2014.10.073.

79. Kuchti T. An implicit method for transient gas flows in pipe networks. Int J Heat Fluid Flow. 1994;15(S):378–93. https://doi.org/10.1016/0142-727X(94)90051-5.

80. Kräkel J, Stieger P, Vostry Z, Žávorka J. Modeling the dynamic flow of gas pipelines. IEEE Trans Syst Man Cybern. 1984;SMC-14(4):586–96. https://doi.org/10.1109/TSMC.1984.6313330.

81. Kräkel J, Stieger P, Vostry Z, Žávorka J. A universal dynamic simulation model of gas pipeline networks. IEEE Trans Syst Man Cybern. 1984;SMC-14(4):597–606. https://doi.org/10.1109/TSMC.1984.6313331.

82. Kulpa F, Happenhofer N, Higuera I, Koch O. Total-variation-diminishing implicit-explicit Runge-Kutta methods for the simulation of double-diffusive convection in astrophysics. J Comput Phys. 2012;231(9):3561–86. https://doi.org/10.1016/j.jcp.2011.12.031.

83. Kutta W. Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. Z Angew Math Phys. 1901;46(6):435–53. https://archive.org/stream/zeitsschrftfma12nunggpo/n449/mode/2up.

84. Lall S, Marsden JE, Glavaški S. Empirical model reduction of controlled nonlinear systems. In: IFAC proceedings volumes (14th IFAC world congress). vol. 32. 1999. p. 2598–603. https://doi.org/10.14744/667017564423.2.

85. Lang J, Leugering G, Martin M, Gasnetzwerke CT. Mathematische Modellierung, Simulation und Optimierung. Mitt Oirsch Math-Ver. 2015;23(1):30–5. https://doi.org/10.1515/mdmm-2015-0013.

86. Liljegren-Sailer B, Marheineke M. A structure-preserving model order reduction approach for space-discrete gas networks with active elements. In: Quintela P, Barral P, Gómez D, Pena FI, Rodríguez J, Salgado P, Vázquez-Méndez ME, editors. Progress in industrial mathematics at ECMI 2016. Mathematics in industry. vol. 26. Berlin: Springer; 2017. p. 439–46. https://doi.org/10.1007/978-3-319-63082-3_69.

87. Lu H, Guo I, Azimi M, Huang K. Oil and gas 4.0 era: a systematic review and outlook. Comput Ind. 2019;111:68–90. https://doi.org/10.1016/j.compind.2019.06.007.

88. Lu Y, Marheineke M, Mohring J. Stability-preserving interpolation strategy for parametric MOR of gas pipeline-networks. In: Quintela P, Barral P, Gómez D, Pena FI, Rodríguez J, Salgado P, Vázquez-Méndez ME, editors. Progress in industrial mathematics at ECMI 2016. Mathematics in industry. vol. 26. Berlin: Springer; 2016. p. 431–7. https://doi.org/10.1007/978-3-319-63082-3_68.

89. Lu Y, Marheineke M, Mohring J. Interpolation strategy for BT-based parametric MOR of gas pipeline-networks. In: Benner P, Othberger M, Patera A, Rozza G, Urban K, editors. Model reduction of parameterized systems. MS&A. vol. 17. Berlin: Springer; 2017. p. 267–80. https://doi.org/10.1007/978-3-319-58786-8_24.

90. Mak TWK, Van Hentenryck P, Zlotnik A, Bent R. Dynamic compressor optimization in natural gas pipeline systems. INFORMS J Comput. 2019;31(1):1–26. https://doi.org/10.1287/ijoc.2018.0821.

91. Mak TWK, Van Hentenryck P, Zlotnik A, Hijazi H, Bent R. Efficient dynamic compressor optimization in natural gas transmission systems. In: Proceedings of the American control conference. 2016. p. 7484–91. https://doi.org/10.1109/ACC.2016.7526855.

92. März B. The index of linear differential algebraic equations with properly stated leading terms. Results Math. 2002;40:308–38. https://doi.org/10.1007/BF03322858.

93. Mischner J. Notizen zur hydraulischen Berechnung von Gasleitungen. GWF, Gas - Erdgas. 2012;153(4):258–73. https://www.vulkan-shop.de/notizen-zur-hydraulischen-berechnung-von-gasleitungen-2591.

94. Mischner J, Fasold HG, Heymer J, editors. gas2energy.net Edition gas for energy. DIV, 2016. URL: https://www.vulkan-shop.de/gas2energy-net-0198.

95. Mohring J, Hoffmann J, Halfmann T, Zemits A, Basso G, Lagoni P. Automated model reduction of complex gas pipeline networks. In: PSIG annual meeting. 2004. p. PSIG-04B3. https://doi.org/10.1016/j.apnum.2016.10.018.

96. Möhring J, Hoffmann J, Halfmann T, Zemits A, Basso G, Lagoni P. Automated model reduction of complex gas pipeline networks. In: PSIG annual meeting. 2004. p. PSIG-04B3. https://doi.org/10.1016/j.apnum.2016.10.018.

97. Mohring J, Hoffmann J, Halfmann T, Zemits A, Basso G, Lagoni P. Automated model reduction of complex gas pipeline networks. In: PSIG annual meeting. 2004. p. PSIG-04B3. https://doi.org/10.1016/j.apnum.2016.10.018.

98. Mohring J, Hoffmann J, Halfmann T, Zemits A, Basso G, Lagoni P. Automated model reduction of complex gas pipeline networks. In: PSIG annual meeting. 2004. p. PSIG-04B3. https://doi.org/10.1016/j.apnum.2016.10.018.

99. Moore BC. Principal component analysis in linear systems: controllability, observability, and model reduction. IEEE Trans Autom Control. 1981;26(1):17–32. https://doi.org/10.1109/TAC.1981.1120568.

100. Nelles O. Nonlinear system identification. Berlin: Springer; 2001. https://doi.org/10.1007/978-3-662-04323-3.

101. Ochs K. Passive integration methods: fundamental theory. AEÜ, Int J Electron Commun. 2001;55(3):153–63. https://doi.org/10.1016/S1434-8411(00)00024.

102. Or AC, Speyer JL, Kim J. Reduced balancing transformations for large nonnormal state-space systems. J Guid Control Dyn. 2012;35(1):129–37. https://doi.org/10.2514/1.53777.
136. Vandendorpe A, Van Dooren P. Model reduction of interconnected systems. In: Schilders WHA, van der Vorst HA, Rommes J, editors. Model order reduction: theory, research aspects and applications. Mathematics in industry. vol. 13. Berlin: Springer; 2008. p. 305–21. https://doi.org/10.1007/978-3-540-78841-6_14.

137. Varga A. Minimal realization procedures based on balancing and related techniques. In: Pichler F, Diaz RM, editors. Computer aided systems theory – EUROCAST’91. Lecture notes in computer science. vol. 585. Berlin: Springer; 1991. p. 733–61. https://doi.org/10.1007/3-540-58422-0_56.

138. Vostrý Z, Záworka J. Simulation and control of gas transport and distribution by large-scale pipeline networks. In: Kárný M, Warwick K, editors. Mutual impact of computing power and control theory. Berlin: Springer; 1993. p. 65–75. https://doi.org/10.1007/978-1-4615-2968-2_4.

139. Vuffay M, Misra S, Chertkov M. Monotonicity of dissipative flow networks renders robust maximum profit problem tractable: general analysis and application to natural gas flows. In: 54th IEEE conference on decision and control (CDC). 2015. p. 4571–8. https://doi.org/10.1109/CDC.2015.7402993.

140. Wild A, le Roux JD, Craig K. Modelling of methane-rich gas pipeline networks for simulation and control. J Process Control. 2020;92:234–45. https://doi.org/10.1016/j.jprocont.2020.06.010.

141. Willcox K, Peraire J. Balanced model reduction via the proper orthogonal decomposition. AIAA J. 2002;40(1):2323–30. https://doi.org/10.2514/2.1570.

142. Williams MO, Schmid PJ, Kutz JN. Hybrid reduced-order integration with proper orthogonal decomposition and dynamic mode decomposition. Multiscale Model Simul. 2013;11(2):522–44. https://doi.org/10.1137/120874539.

143. Zerz E. Topics in multidimensional linear systems theory. Lecture notes in control and information sciences. vol. 256. London: Springer, 2000. https://doi.org/10.1007/BFb0110139.

144. Zlotnik A, Chertkov M, Backhaus S. Optimal control of transient flow in natural gas networks. In: 54th IEEE conference on decision and control (CDC). 2015. p. 4563–70. https://doi.org/10.1109/CDC.2015.7402932.

145. Zlotnik A, Dyachenko S, Backhaus S, Chertkov M. Model reduction and optimization of natural gas pipeline dynamics. In: Proceedings of the ASME dynamic systems and control conference. 2015. p. 1–10. https://doi.org/10.1115/DSCC2015-9683.