

Efficient state estimation for gas pipeline networks via low-rank approximations

Nadine Stahl^{1*}, Nicole Marheineke¹,

¹Chair of Modeling and Numerics, FB IV – Mathematics, Trier University, D-54286 Trier, Germany

*nadine.stahl@uni-trier.de

Abstract. In this paper we investigate the performance of projection-based low-rank approximations in Kalman filtering. For large-scale gas pipeline networks structure-preserving model order reduction has turned out to be an advantageous way to compute accurate solutions with much less computational effort. For state estimation we propose to combine these low-rank models with Kalman filtering and show the advantages of this procedure to established low-rank Kalman filters in terms of efficiency and quality of the estimate.

Introduction

Efficient state estimation plays an important role for model-predictive control of gas pipeline networks, [1, 8, 18]. Gas network models are large-scale systems of high complexity. They consist of nonlinear partial differential equations coupled with algebraic constraints, whose simulation is computationally very expensive. To reduce the effort, model hierarchies have been established [3] and, recently, structure-preserving model order reduction has been developed [6, 14]. The reduced (low-rank) models are in particular mass-conserving and energy-dissipating.

In state estimation Kalman filtering is popular. The Kalman Filter, developed in the 1960s [10], is still one of the most prominent filtering algorithms because of its optimality property for linear systems. Since it is not suitable for large-scale systems due to computational reasons, low-rank Kalman filters have been developed, see [16] and references within. They reduce the computational effort by simplifying the costly and memory-demanding calculations of the error covariances in the filtering process. For example, the Compressed State Kalman Filter [12] reduces the covariance matrices via projection techniques and the Spectral Kalman Filter [7] by help of a Taylor expansion, whereas the Ensemble Kalman Filter [9] considers a sample covariance by

collecting an ensemble of possible states with a Monte-Carlo method. Recent work addresses the use of a reduced-order model in the filtering process for linear and nonlinear parabolic systems [5, 17].

In this paper we investigate the use of a structure-preserving reduced (low-rank) model in Kalman filtering for gas networks. Since a reduced nonlinear model has proven to be still too computationally expensive in the application [19], we consider a linearized model variant. We deduce our low-rank approximation by projection-based model order reduction. Our approach has hence similarities with the Compressed State Kalman Filter being a projection-based low-rank filter. However, we show that our approach is significantly superior to established low-rank Kalman filters in terms of computational speed and memory demands, while the obtained state estimates are of competitive accuracy. We also study the combination of a low-rank model with a low-rank filter.

The paper is structured as follows: After introducing a model hierarchy for gas pipeline networks and the filtering in Sec. 1, we present our Kalman filtering approach with a reduced model and discuss its similarities and differences to the Compressed State Kalman Filter in Sec. 2. A numerical performance study for the low-rank approximations – in terms of quality and efficiency – is carried out in Sec. 3, using an academic benchmark example as well as a real gas pipeline network from western Germany.

1 Problem setting

Modeling a gas pipeline network, we proceed from the nonlinear isothermal Euler equations and establish a model hierarchy by help of linearization and structure-preserving reduction.

1.1 Hierarchical gas network models

The gas pipeline network is described by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the pipes are represented by the edges \mathcal{E} and the junctions \mathcal{V}_I and inlets/outlets \mathcal{V}_B by the nodes \mathcal{V} , in particular $\mathcal{V} = \mathcal{V}_I \cup \mathcal{V}_B$, $\mathcal{V}_I \cap \mathcal{V}_B = \emptyset$. The gas dynamics in a pipe e of length l^e is modeled by nonlinear isothermal Euler-type equations for pressure p^e and mass flux q^e , i.e., for $e \in \mathcal{E}$, $(x, t) \in [0, l^e] \times [0, T]$,

$$a^e \partial_t p^e = -\partial_x q^e, \quad b^e \partial_t q^e = -\partial_x p^e - d^e \frac{|q^e|}{p^e} q^e, \quad (1)$$

with constant pipe parameters a^e, b^e, d^e and coupled via Kirchhoff's conditions in every junction $v \in \mathcal{V}_I$

$$\sum_{e \in \delta_v^-} q^e(l^e, t) = \sum_{e \in \delta_v^+} q^e(0, t),$$

$$p^e(l^e, t) = p^v(t), e \in \delta_v^+, \quad p^e(0, t) = p^v(t), e \in \delta_v^-,$$

where δ_v^+, δ_v^- denote the sets of all topologically ingoing and outgoing edges to v . At inlets/outlets $v \in \mathcal{V}_B$ we prescribe the pressure p^v as

$$p^v(t) = u^v(t) = u_D^v(t) + u_S^v(t),$$

$$du_S^v = \kappa^v (\mu^v - u_S^v) dt + \sigma^v dW_t^v.$$

The time-varying input u^v consists of a deterministic part u_D^v and a stochastic one u_S^v , where we particularly model u_S^v by an Ornstein-Uhlenbeck process with constant parameters $\kappa^v, \mu^v, \sigma^v$ and a standard Wiener process W_t^v . The stochastic boundary data allows the incorporation of, e.g., market strategies or the concept of Power-to-Gas [4]. Certainly, also other boundary types can be considered, see e.g., [14]. We initialize the transient model with the stationary solution associated to the unperturbed input $u^v = u_D^v(0)$, $v \in \mathcal{V}_B$. A linear model variant is obtained when considering a simplified linearized friction term in (1) with respective constant d_l^e ,

$$a^e \partial_t p^e = -\partial_x q^e, \quad b^e \partial_t q^e = -\partial_x p^e - d_l^e q^e. \quad (2)$$

For an overview on further hierarchical models see, e.g., [3].

1.2 Structure-preserving reduced models

The unperturbed network models can be embedded in a port-Hamiltonian framework and allow for structure-

preserving and robust spatial Galerkin approximations. The Galerkin approximation is applicable to finite element discretization and projection-based model order reduction. Under mild assumptions on the ansatz spaces, mass conservation and energy dissipation are ensured [6, 14, 15].

As in [6], we use mixed finite elements and the moment matching method proposed there for the network model (2), yielding linear time-invariant descriptor systems of the form

$$\mathbf{E} \dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} (\mathbf{u}_D + \mathbf{u}_S), \quad (3a)$$

$$\mathbf{y} = \mathbf{C} \mathbf{x}, \quad (3b)$$

$$d\mathbf{u}_S = \mathfrak{K} (\boldsymbol{\mu} - \mathbf{u}_S) dt + \boldsymbol{\Sigma} d\mathbf{W}_t. \quad (3c)$$

The input functions u_D, u_S account for the boundary conditions with diagonal matrices $\mathfrak{K}, \boldsymbol{\Sigma} \in \mathbb{R}^{|\mathcal{V}_B| \times |\mathcal{V}_B|}$ containing the entries κ^v and σ^v , respectively, and $\boldsymbol{\mu} = (\mu^v)_{v \in \mathcal{V}_B}$. The output $\mathbf{y}(t) \in \mathbb{R}^R$ with certain pressure or flux values corresponds to measurement data in the context of filtering, $t \in [0, T]$. In case of the full order model the state $\mathbf{x}(t) \in \mathbb{R}^N$ represents the space-discrete pressure and flux, its size is determined by the number of pipes and finite elements for each pipe. In the model reduction the system structure is kept. We introduce an orthonormal projection matrix $\mathbf{V} \in \mathbb{R}^{N \times n}$, $n \ll N$ with the property $\mathbf{x} \approx \mathbf{V} \hat{\mathbf{x}}$, i.e., $\mathbf{V}^T \mathbf{V} = \mathbf{I}$, \mathbf{I} identity. The reduced quantities are indicated by $\hat{\cdot}$, the system matrices particularly read

$$\hat{\mathbf{E}} = \mathbf{V}^T \mathbf{E} \mathbf{V}, \quad \hat{\mathbf{A}} = \mathbf{V}^T \mathbf{A} \mathbf{V}, \quad \hat{\mathbf{B}} = \mathbf{V}^T \mathbf{B}, \quad \hat{\mathbf{C}} = \mathbf{C} \mathbf{V}. \quad (4)$$

Note that we preserve a block structure by setting up the reduced spaces for pressure V_p and flux V_q separately. Imposing a compatibility condition on the spaces, $\partial_x V_p = V_q$, ensures stable reduced models [6, 15].

1.3 Stochastically forced model for filtering

The state estimation is based on a filtering model, for which we use the linear spatially discretized network model in the full or a reduced version. Uncertainties are incorporated via a system noise being modeled as (driving) uncorrelated Gaussian process in (3a).

Let $\mathbf{x}(t) = (\mathbf{x}^T, \mathbf{u}_S^T)^T(t) \in \mathbb{R}^{N+|\mathcal{V}_B|}$, $t \in [0, T]$, we consider equidistant time points $t_k = k\tau$, $\tau = T/K$, $k = 0, \dots, K$, then our filtering model in time-discrete

form reads

$$\begin{aligned} \mathbf{x}_{k+1} &= \Phi \mathbf{x}_k + \Psi \mathbf{u}_k + \mathbf{w}_k \\ \Phi &= \begin{pmatrix} \mathbf{A}_\tau^{-1}(\mathbf{E} + \tau(1-\theta)\mathbf{A}) & \tau\mathbf{A}_\tau^{-1}\mathbf{B} \\ \mathbf{0} & (\mathbf{I} + \tau\mathfrak{K})^{-1} \end{pmatrix} \\ \Psi &= \begin{pmatrix} \tau\mathbf{A}_\tau^{-1}\mathbf{B} \\ \tau(\mathbf{I} + \tau\mathfrak{K})^{-1}\mathfrak{K}\mu \end{pmatrix}. \end{aligned} \quad (5)$$

with $\mathbf{A}_\tau = \mathbf{E} - \tau\theta\mathbf{A}$. Here, \mathbf{x}_k denotes the approximation $\mathbf{x}_k \approx \mathbf{x}(t_k)$, and $\mathbf{u}_k = ((\theta\mathbf{u}_D(t_{k+1}) + (1-\theta)\mathbf{u}_D(t_k))^T, \mathbf{1}^T)^T$ the respective input at time t_k with vector of ones $\mathbf{1} \in \mathbb{R}^{|\mathcal{B}|}$. The underlying time-integration is based on a θ -scheme for (3a) and an Euler(-Maruyama)-scheme for (3c). Our different treatment is motivated by the observation that the Euler-scheme is sufficient for the computation of the boundary data, whereas the more sophisticated θ -scheme allows for a better capturing of the dynamic behavior of the flow quantities. The state noise $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ is a normal distributed centered random variable with constant diagonal covariance matrix \mathbf{Q} for every time t_k , in particular

$$\mathbf{Q} = \tau \begin{pmatrix} \mathbf{Z}\mathbf{Z}^T & \\ & \Sigma\Sigma^T \end{pmatrix}.$$

It results from the system noise with amplitude $\mathbf{Z} \in \mathbb{R}^{N \times N}$ (diagonal matrix) added to (3a) and the scaled Wiener process in (3c).

The output is assumed to be measurable with suitable devices. To account for measurement errors, noise is added to the output equation (3b), i.e.,

$$\mathbf{y}_k = \mathbf{H} \mathbf{x}_k + \mathbf{v}_k \quad (6)$$

with \mathbf{y}_k output at time t_k and $\mathbf{H} = (\mathbf{C}, \mathbf{0})$. The discrete time noise $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$, with constant covariance matrix $\mathbf{R} \in \mathbb{R}^{R \times R}$, is assumed to be uncorrelated in time and to be component-wisely independent of \mathbf{w}_k . In this study the measurement data $\{\mathbf{y}_k\}$ for the filtering is provided by the outputs of the nonlinear network model (1).

Using the reduced-order model for filtering, the system matrices are replaced by their reduced counterparts, yielding $\hat{\Phi}$, $\hat{\Psi}$ and $\hat{\mathbf{H}}$ in (5) and (6). Since the input process is not affected by the model order reduction, the reduced covariance of the state noise is given by $\hat{\mathbf{Q}} = \mathbf{V}_x^T \mathbf{Q} \mathbf{V}_x$ with

$$\mathbf{V}_x = \begin{pmatrix} \mathbf{V} & \\ & \mathbf{I} \end{pmatrix}. \quad (7)$$

2 Low-rank approximations

The focus of this work is on Kalman filtering algorithms, since our filtering model is linear. The Kalman Filter (KF) is known to be optimal for linear systems with white noise as state and measurement noise in the sense that it is unbiased and minimizes the error variance [10]. However, it is not applicable to large-scale systems for computational reasons. We investigate a filtering approach based on projection-based model order reduction involving low-rank models and discuss its relation to the established Compressed State Kalman Filter [12, 13] being a projection-based low-rank Kalman filter.

Kalman filtering consists of a prediction and a correction step. In each of these steps a state estimate and its error covariance are computed. At time point t_k , the state and the error covariance matrix of the prediction step are denoted by $\mathbf{x}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$, where the first index indicates the current time point t_k and the second index stands for the time up to which measurement data is considered, i.e., t_{k-1} . In the correction step additional measurement data of the current time point is taken into account, yielding the state estimate $\mathbf{x}_{k|k}$ and the corrected error covariance matrix $\mathbf{P}_{k|k}$. The correction ratio between prediction and correction is given by the so-called Kalman gain \mathbf{K}_k , see Algorithm 1.

2.1 Kalman Filter using reduced-order models

In the Kalman Filter the computational effort comes from the determination of the error covariances (and the Kalman gains) being fully occupied matrices of the system's size. The matrix evaluations become memory-intensive and expensive for large-scale systems. Establishing a model hierarchy, we derive a low-rank approximation from projection-based model order reduction. Using a low-rank model to estimate the state and to compute the error covariances significantly reduces the effort and makes the Kalman Filter applicable. Prolongating the resulting low-rank estimate to the high-dimensional space yields then the desired state estimate for the large-scale system.

The Kalman Filter applied to the reduced-order model (RKF) gives an estimate for the reduced state $\hat{\mathbf{x}}_{k|k}$ and its associated error covariance matrix $\hat{\mathbf{P}}_{k|k}$ of small size. To get the quantities $\mathbf{x}_{k|k}$ and $\mathbf{P}_{k|k}$ of the full order (large-scale) system, we exploit the linearity of the expectation value and the approximation property of the reduced state $\mathbf{x} \approx \mathbf{V}_x \hat{\mathbf{x}}$. This leads to $\mathbf{x}_{k|k} \approx \mathbf{V}_x \hat{\mathbf{x}}_{k|k}$ and

Algorithm 1: Kalman Filter

```

// initialization
1  $\mathbf{x}_{0|0} = \mathbf{x}_0, \mathbf{P}_{0|0} = \mathbf{P}_0;$ 
2 for  $k = 0, \dots, K-1$  do
    // prediction
3    $\mathbf{x}_{k+1|k} = \Phi \mathbf{x}_{k|k} + \Psi \mathbf{u}_k;$ 
4    $\mathbf{P}_{k+1|k} = \Phi \mathbf{P}_{k|k} \Phi^T + \mathbf{Q};$ 
    // correction
5    $\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k} \mathbf{H}^T (\mathbf{H} \mathbf{P}_{k+1|k} \mathbf{H}^T + \mathbf{R})^{-1};$ 
6    $\mathbf{x}_{k+1|k+1} = \mathbf{x}_{k+1|k} + \mathbf{K}_{k+1} (\mathbf{y}_{k+1} - \mathbf{H} \mathbf{x}_{k+1|k});$ 
7    $\mathbf{P}_{k+1|k+1} = (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}) \mathbf{P}_{k+1|k};$ 
8 end
    
```

$$\mathbf{P}_{k|k} \approx \mathbf{V}_x \hat{\mathbf{P}}_{k|k} \mathbf{V}_x^T.$$

2.2 Compressed State Kalman Filter

The idea of the Compressed State Kalman Filter (CSKF) to overcome the computational complexity of the (classical) Kalman Filter is to replace the error covariance matrix with a low-rank approximation. Thus, it is assumed that there exists a constant orthonormal projection matrix \mathbf{V}_P , $\mathbf{V}_P^T \mathbf{V}_P = \mathbf{I}$, such that $\mathbf{P}_{k|l} \approx \mathbf{V}_P \tilde{\mathbf{P}}_{k|l} \mathbf{V}_P^T$ holds for both the predicted ($l = k-1$) and the corrected error covariance matrix ($l = k$) with $\tilde{\mathbf{P}}_{k|l}$ being of much smaller size. For the filter matrices, low-rank versions are precomputed offline, i.e., $\tilde{\Phi} = \mathbf{V}_P^T \Phi \mathbf{V}_P$, $\tilde{\mathbf{H}} = \mathbf{H} \mathbf{V}_P$ and $\tilde{\mathbf{Q}} = \mathbf{V}_P^T \mathbf{Q} \mathbf{V}_P$, that are then used for the determination of the low-rank error covariance $\tilde{\mathbf{P}}_{k|k}$ and the respective low-rank Kalman gain $\tilde{\mathbf{K}}_k$ in each time step (cf., Algorithm 1, lines 4, 5 and 7). The calculations for the state estimate are performed with the full-rank filter matrices. They stay the same as in the (classical) Kalman Filter, except for prolongating the Kalman gain to the high-dimensional space ($\mathbf{K}_k \approx \mathbf{V}_P \tilde{\mathbf{K}}_k$).

2.3 Relation between projection-based filtering variants

The Kalman Filter on a reduced-order model and the Compressed State Kalman Filter seem to be very similar at first glance, since they both deal with projection-based low-rank approximations. However, there exist severe differences.

CSKF still uses the large-scale system for the computation of the state estimates, while the reduction

only affects the error covariance matrices and thus the Kalman gain. In contrast, RKF uses a low-rank model for the entire filtering process, including the state computation. This is expected to further improve the computational speed, but probably at the cost of accuracy compared to CSKF. For a numerical study on this point see Sec. 3.

As far as the projection is concerned, there exist multiple techniques in the field of model order reduction on how to construct the projection matrix \mathbf{V}_x for a low-rank model, e.g., with Moment Matching, Balanced Truncation or Proper Orthogonal Decomposition (see, e.g., [2] for an overview). Usually, the special structure of the underlying large-scale system (differential, algebraic or additional input equations) is taken into account to achieve the best possible approximation quality. In CSKF, the choice of \mathbf{V}_P is not obvious, as it exclusively acts on a statistical quantity (error covariance matrix). In the following investigations we use $\mathbf{V}_x = \mathbf{V}_P$.

Applying the same projection matrix, one might expect the same low-rank approximations for the error covariances in both filtering variants when starting with the same initialization. But this is in general not the case, as a closer look at their computation in the prediction step reveals. We find

$$\begin{aligned} \hat{\mathbf{P}}_{k+1|k} &= \hat{\Phi} \hat{\mathbf{P}}_{k|k} \hat{\Phi}^T + \hat{\mathbf{Q}} \quad \text{for RKF,} \\ \tilde{\mathbf{P}}_{k+1|k} &= \tilde{\Phi} \tilde{\mathbf{P}}_{k|k} \tilde{\Phi}^T + \tilde{\mathbf{Q}} \quad \text{for CSKF.} \end{aligned}$$

While the state noise covariances satisfy $\tilde{\mathbf{Q}} = \hat{\mathbf{Q}}$, the filter state matrices differ, $\hat{\Phi} \neq \tilde{\Phi}$, in case of an underlying implicit time-discretization. See, e.g., the first diagonal block matrix for our pipe network model and the projection matrix (7),

$$\begin{aligned} \hat{\Phi}_{11} &= (\mathbf{V}^T (\mathbf{E} - \tau \theta \mathbf{A}) \mathbf{V})^{-1} \mathbf{V}^T (\mathbf{E} + \tau (1 - \theta) \mathbf{A}) \mathbf{V} \\ &\neq \mathbf{V}^T (\mathbf{E} - \tau \theta \mathbf{A})^{-1} (\mathbf{E} + \tau (1 - \theta) \mathbf{A}) \mathbf{V} = \tilde{\Phi}_{11}. \end{aligned}$$

In this context, the projection might be interpreted as a model reduction before or after time-discretization for RKF and CSKF, respectively.

2.4 Reduced Ensemble Kalman Filter

Additionally to the projection-based approaches, we consider the Ensemble Kalman Filter on a reduced-order model (REnKF) for comparison reasons in Sec. 3. The Ensemble Kalman Filter is probably the most used

Kalman filter variant for large-scale systems, [9]. This filtering algorithm avoids the costly evaluation of the algebraic expression of the error covariance matrix in the prediction step (matrix-matrix-matrix product of full occupied matrices of system's size, cf. Algorithm 1, line 4). Applying a Monte-Carlo method, the error covariance is numerically approximated on basis of an ensemble of samples of the stochastic filter model. The calculation of Kalman gain and corrected error covariance follows the procedure of the classical Kalman Filter. The state estimate is given as ensemble average.

This procedure usually requires only a very small amount of memory, as not the whole covariance matrix, but only a few samples, whose numbers are usually much less compared to the state dimension, have to be stored. Moreover, there exist observation matrix-free implementations. As for computational time, there is not much gain to be expected from parallelizing the ensemble filter, since collecting all samples after each time step involves a massive overhead. The Ensemble Kalman Filter can be combined straightforward with a reduced-order (low-rank) model, as filter model and filter technique are independent of each other.

3 Numerical investigations

We investigate the performance of the low-rank filtering techniques in terms of accuracy of the state estimates and computational effort. As gas pipeline networks we consider an academic example of small size (diamond topology) for which the classical Kalman Filter is still applicable as well as a large-scale real network from western Germany. As measurement data we take the mass flux values at the boundary nodes obtained from the nonlinear network model which also serves as reference. For REnKF we always use 100 samples in accordance to the recommendations in [9]. All computations are performed in MatlabR2017b on an Intel Xeon with 2.2GHz using 12 Cores.

3.1 Academic diamond network

The small diamond network of 7 edges and 6 nodes serves as academic benchmark setting, see Fig. 1. The pipe parameters are taken from [4]. For the linear model we use $d_i^e = d^e |q_{av}^e| / p_{av}^e$, where p_{av}^e and q_{av}^e are the average pressure and mass flux on edge e of the stationary solution associated to the input $u^v = u_D^v(0)$, $v \in \mathcal{V}_B$. In the subsequent example each pipe is discretized with

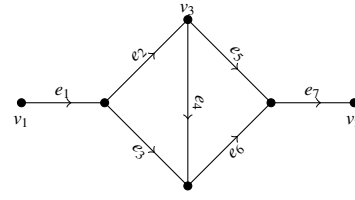


Figure 1: Academic example: diamond network topology.

250 equidistant finite elements, yielding a full order system with $N = 3511$ degrees of freedom. The reduced model is of size $n = 29$, which implies a low-rank filter model of size $n + |\mathcal{V}_B| = 31$ when adding the two equations for the boundary data. The reduction error is of order $\mathcal{O}(10^{-4})$. Concerning the filter model, the system noise is exclusively added to the dynamic equations, it is set up with respect to the maximal deviation of the stationary solution from its average values (pressure, mass flux) on each pipe. The algebraic constraints are not perturbed, as they describe mass conservation across junctions. Note that they are preserved under the applied model order reduction. The measurement noise accounts for 1% deviation from the maximal measured flux values. As input functions we choose

$$u_D^{v_1}(t) = \begin{cases} 2+t, & 0 \leq t < 1, \\ 3, & 1 < t < 5, \\ 1.5 - 0.1t, & 5 \leq t < 10, \\ 2, & t \geq 10, \end{cases} \quad u^{v_2}(t) \equiv 2,$$

$$(\mu, \kappa, \sigma)^{v_1} = (0, 3, 0.2)$$

for $t \in [0, 20]$. We apply 1000 time steps and set $\theta = 0.51$.

Figure 2 illustrates the temporal evolution of the mass flux at the inner node v_3 on edge e_2 (cf., Fig. 1) that is obtained from the nonlinear network model (1) (reference solution) and from the linear model (2) as well as is estimated by the different low-rank filtering techniques. Although the estimates are computed on basis of the linear model whose solution differs strongly from the nonlinear one, they capture the nonlinear behavior being included via the measurements very well. Regarding the estimation errors (i.e., temporal mean of the relative spatial L^2 -errors wrt. reference) in Table 1, the (classical) Kalman Filter shows the best approximation properties. CSKF achieves a comparable error, whereas the errors of the other low-rank filters (Kalman and Ensemble Filter on reduced-order model)

Table 1: Errors and CPU time for diamond network

Filter	$\text{mean}_j \frac{\ \mathbb{E}[\mathbf{x}_j - \hat{\mathbf{x}}_j]\ _{L^2}}{\ \mathbb{E}[\mathbf{x}_j]\ _{L^2}}$	Offline [s]	Online [s]
KF	$3.7 \cdot 10^{-2}$	$8.4 \cdot 10^2$	$5.2 \cdot 10^1$
RKF*	$6.0 \cdot 10^{-2}$	$0.2 \cdot 10^0$	$7.4 \cdot 10^{-2}$
CSKF	$3.8 \cdot 10^{-2}$	$2.1 \cdot 10^0$	$1.5 \cdot 10^0$
REnKF* \diamond	$9.8 \cdot 10^{-2}$	$1.2 \cdot 10^{-1}$	$3.2 \cdot 10^{-1}$

* additional post-processing for prolongation is required

\diamond offline time refers to sampling of random variables

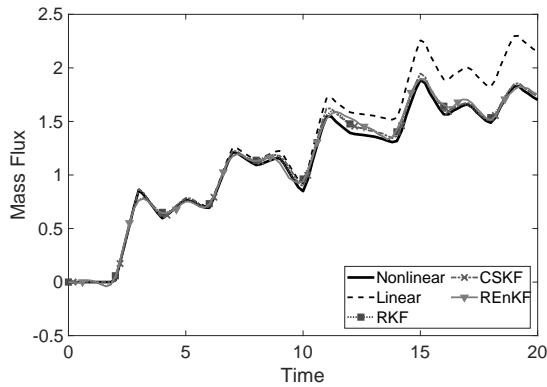


Figure 2: Simulation and estimation results for mass flux $q^{e2}(v_3)$, cf., Fig. 1.

are of same order but slightly worse. The better accuracy of CSKF towards the other low-rank filters might be expected as it estimates the states on basis of the large-scale model. However, all results are astonishingly good regarding the underlying model order reduction error (cf., Fig. 3).

Concerning the computational time, RKF – but also REnKF – clearly outperforms the filters operating on the large-scale model by several orders of magnitude, Table 1. We distinguish between offline (pre-computation) and online (actual runtime) phases. For the (classical) Kalman Filter as well as its projection-based low-rank variants, the error covariance matrices and the Kalman gains are precomputed offline. So, the runtime only consists of the state estimation taking into account the actual inputs and measurements. In the ensemble filtering this pre-computation of covariances and Kalman gains is not possible, since all quantities depend on the chosen samples. Changing input or measurement data hence requires a completely new filtering process. The effort of REnKF to approximate the sam-

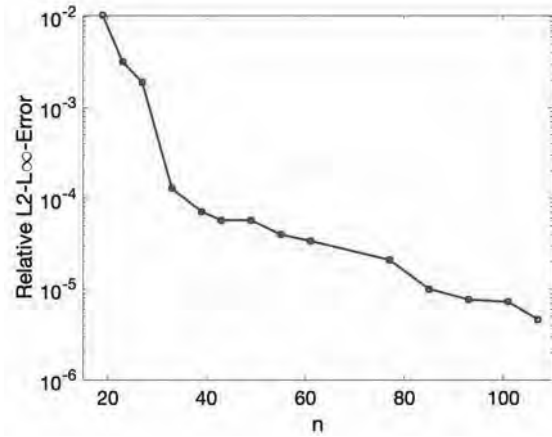


Figure 3: Model order reduction error for p and q (L^2 in space, L^∞ in time) over system size.

ple covariance is slightly higher as running the online phase of RKF. REnKF scales here with the number of samples. As a low-rank filtering method operating on a low-rank model, better performance could possibly be achieved by tuning. For sophisticated tuning of the filter, i.e., reducing the required sample size for a comparatively good approximation and achieving better runtimes, we refer to [9, 11]. The offline time of REnKF listed in Table 1 refers to the sampling of the random variables which can become time-consuming for large vectors in Matlab. Note that filtering on a reduced-order model requires an additional post-processing step, i.e., prolongation of the reduced variables, to obtain the estimate for the high-dimensional state and its error covariance.

3.2 Large-scale real pipeline network

The partDE network from western Germany consists of 636 pipes connected in 487 junctions with 47 in-/outlets, see Fig. 4. The pipes have a maximal length of about 120 km, the specific pipe parameters can be found in [20]. In the subsequent example each pipe is discretized equidistantly with a maximal element length of 100 m, yielding a full order model of dimension $N = 86160$. The reduced model is of size $n = 1356$, the reduction error is of order $\mathcal{O}(10^{-4})$. As input we prescribe the pressure in bar, particularly we impose

$$\begin{aligned} u_D^{v1}(t) &= 60 + 5 \sin(0.03t), \\ u_D^{v2}(t) &= 70 + 7 \cos(0.1t), \end{aligned}$$



Figure 4: Gas pipeline network from western Germany "partDE" [20].

$$\begin{aligned} u_D^{V3}(t) &= 65 + 2 \sin(0.05t), \\ u_D^{V4}(t) &= 80 + 4 \sin(0.008t), \\ u_D^{V5}(t) &= 55 + 5 \sin(0.017t), \end{aligned}$$

$(\mu, \kappa, \sigma)^{V_i} = (0, 3, 0.2)$, $i = 1, \dots, 5$, at the boundary nodes marked in Fig. 4 and $u^{V_j} \equiv 60$ at all others, $j = 6, \dots, 47$. A time horizon of 12 hours is covered by 720 time steps of 1 minute length. The other model parameters are set and the computations are performed in the same way as in the academic example.

This real large-scale network makes it necessary for computational reasons to use low-rank approximations for state estimation. The performance of the filters is as in the academic example, see Table 2. Compared to filtering with the reduced-order (low-rank) model, the approximation quality of CSKF is slightly better (by less than one order) since the state is estimated on the large-scale model. The price to pay, however, is a significantly longer computing time. An additional drawback, getting more importance the larger the full order model becomes, is the amount of memory needed. This results from the Kalman gains being matrices in dimension of the large-scale model, which are saved in the pre-computation. The other two methods do not have this disadvantage, since they operate exclusively on the

Table 2: Errors and CPU time for partDE network

Filter	mean _j $\frac{\ \mathbb{E}[x_j - \hat{x}_{j i}]\ _{L^2}}{\ \mathbb{E}[x_j]\ _{L^2}}$	Offline [s]	Online [s]
RKF *	$2.2 \cdot 10^{-1}$	$2.5 \cdot 10^2$	$3.4 \cdot 10^1$
CSKF	$6.3 \cdot 10^{-2}$	$2.6 \cdot 10^4$	$2.5 \cdot 10^4$
REnKF * \diamond	$3.5 \cdot 10^{-1}$	$7.1 \cdot 10^1$	$1.4 \cdot 10^2$

* additional post-processing for prolongation is required

\diamond offline time refers to sampling of random variables

low dimension and the Kalman gains are hence of small size. The time for prolongation of state and error covariance is not included in the listed CPU time as it can be performed for selected time points in a post-processing step.

RKF and REnKF yield comparable results in terms of accuracy and efficiency. For ensemble filtering, one might expect that the computational costs could be further reduced by tuning (sample selection), but this is left to future research. However, RKF – with its separation in offline and online phases – is clearly preferable, when input or measurement data changes, as is the case for control of gas pipeline networks.

4 Conclusion

In this work we presented an efficient approach for state estimation in gas networks. Using a low-rank model derived by linearization and projection-based model order reduction, the Kalman Filter, which is unfeasible for the original large-scale system, becomes applicable. We compared our approach to the Compressed State Kalman Filter whose low-rank also comes from projection-based reduction. Although the underlying approximation idea is very similar, our approach yielding estimates of comparable accuracy proved itself to be much more efficient and less memory-intensive in a numerical study on a German pipeline network. Moreover, the structure-preserving model order reduction can be straightforward combined with other low-rank Kalman filter variants, such as the Ensemble Kalman Filter.

Acknowledgement

This work is supported by the German Federal Ministry for Economic Affairs and Energy, project *MathEnergy – Mathematical Key Technologies for Evolving Energy Grids*.

References

- [1] Hesam Ahmadian Behrooz and R. Bozorgmehry Boozarjomehry. Modeling and state estimation for gas transmission networks. *Journal of Natural Gas Science and Engineering*, 22:551–570, 2015.
- [2] Peter Benner, Albert Cohen, Mario Ohlberger, and Karen Willcox, editors. *Model Reduction and Approximation: Theory and Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, 2017.
- [3] Jens Brouwer, Ingenuin Gasser, and Michael Herty. Gas pipeline models revisited: Model hierarchies, non-isothermal models, and simulations of networks. *Multi-scale Modeling & Simulation*, 9(2):601–623, 2011.
- [4] T. Clees, A. Baldin, P. Benner, S. Grundel, C. Himpe, B. Klaassen, F. Küsters, N. Marheineke, L. Nikitina, I. Nikitin, J. Pade, N. Stahl, C. Strohm, C. Tischendorf, and A. Wirsén. MathEnergy – Mathematical key technologies for evolving energy systems. In *Mathematical Modeling, Simulation and Optimization for Power Engineering and Management*, pages 233–262. Springer, 2021.
- [5] Markus Dihlmann and Bernard Haasdonk. A reduced basis Kalman filter for parametrized partial differential equations. *ESAIM - Control, Optimisation and Calculus of Variations*, 22(3):625–669, 2016.
- [6] Herbert Egger, Thomas Kugler, Björn Liljegren-Sailer, Nicole Marheineke, and Volker Mehrmann. On structure-preserving model reduction for damped wave propagation in transport networks. *SIAM Journal on Scientific Computing*, 40(1):A331–A365, 2018.
- [7] Hojat Ghorbanidehno, Amalia Kokkinaki, Judith Li, Eric Darve, and Peter K. Kitanidis. Real-time data assimilation for large-scale systems: The spectral Kalman filter. *Advances in Water Resources*, 86:260–272, 2015.
- [8] Martin Gugat and Rüdiger Schultz. Boundary feedback stabilization of the isothermal Euler equations with uncertain boundary data. *SIAM Journal on Control and Optimization*, 56(2):1491–1507, 2018.
- [9] P. L. Houtekamer and Herschel L. Mitchell. Ensemble Kalman filtering. *Quarterly Journal of the Royal Meteorological Society*, 131:3269–3289, 2005.
- [10] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.
- [11] Matthias Katzfuss, Jonathan R. Stroud, and Christopher K. Wikle. Understanding the Ensemble Kalman Filter. *American Statistician*, 70(4):350–357, 2016.
- [12] Peter K. Kitanidis. Compressed state Kalman filter for large systems. *Advances in Water Resources*, 76:120–126, 2015.
- [13] Judith Yue Li, Amalia Kokkinaki, Hojat Ghorbanidehno, Eric F. Darve, and Peter K. Kitanidis. The compressed state Kalman filter for nonlinear state estimation: Application to large-scale reservoir monitoring. *Water Resources Research*, 51(12):9942–9963, 2015.
- [14] Björn Liljegren-Sailer and Nicole Marheineke. Port-Hamiltonian approximation of a nonlinear flow problem on networks. *SIAM Journal on Scientific Computing*, 2020. accepted.
- [15] Björn Liljegren-Sailer and Nicole Marheineke. On snapshot-based model reduction under compatibility conditions for a nonlinear flow problem on networks. 2021. Preprint, arXiv:2110.04777.
- [16] Lars Nerger, Wolfgang Hiller, and Jens Schröter. A comparison of error subspace Kalman filters. *Tellus A: Dynamic Meteorology and Oceanography*, 57(5):715–735, 2005.
- [17] Stefano Pagani, Andrea Manzoni, and Alfio Quarteroni. Efficient state/parameter estimation in nonlinear unsteady PDEs by a Reduced Basis Ensemble Kalman filter. *SIAM-ASA Journal on Uncertainty Quantification*, 5(1):890–921, 2017.
- [18] H. Prashanth Reddy, Shankar Narasimhan, and S. Murty Bhallamudi. Simulation and state estimation of transient flow in gas pipeline networks using a transfer function model. *Industrial & Engineering Chemistry Research*, 45(11):3853–3863, 2006.
- [19] Nadine Stahl and Nicole Marheineke. A low-rank Extended Kalman filter for gas pipeline networks. In *Progress in Industrial Mathematics at ECMI 2021*. Springer, 2021.
- [20] TU Berlin (TUB-ER). Gas network – partDE data set, 2019. <https://www.er.tu-berlin.de/menu/home/parameter/en/>.