Robust State Estimation for Natural Gas Networks: Integrating Graph Signal Processing and Noise Filtering

OMER BOKOBZA, TSLIL SARDAM

Link for the GitHub source code

Abstract

In the domain of natural gas network systems, precise state estimation amid measurement noise is critical for maintaining both operational efficacy and safety. This project introduces an approach that enhances the robustness of state estimation processes under noisy conditions by integrating sophisticated noise filtering techniques within a data-driven framework. Unlike conventional methods that often accept noisy measurements without modification, our method applies targeted filtering strategies to refine the measurements before processing. This proactive filtration is carefully balanced with a weighted low-rank approximation technique, designed to retain essential data and improve estimation accuracy by exploring potential enhancements with advanced models.

Index Terms— Natural Gas System, Filtering, Noise.

1 INTRODUCTION

In the domain of natural gas network systems, achieving precise state estimation amid measurement noise is essential for maintaining operational efficacy and safety. The paper [1] presents an innovative method that considerably improves the resilience of state estimation processes in the presence of measurement noise. This method eschews traditional extensive filtering and denoising techniques by adopting a data-driven approach that preserves the integrity of the original data. At the heart of this approach is the redefinition of the state estimation challenge into one of weighted low-rank approximation. Our work trying reproducing the results of this framework and extends it with alternative models to explore potential enhancements.

We explore a range of state estimation techniques tailored to the complexities of networked systems, specifically focusing on the application of graph signal processing to simulate natural gas networks. Our implementations tries address the challenges associated with dynamic systems interconnected through networks. Our simulations test the effectiveness of various noise reduction methods, including wavelet denoising, regularization, traditional Kalman filtering, the adapting of the Extended Kalman Filter (EKF) to integrate graph-theoretic concepts and applying GNN-GCN algorithm across different Signal-to-Noise Ratios (SNR). These methodologies are evaluated to demonstrate their potential in enhancing the accuracy and robustness of state estimations under varying noise conditions, providing insights into the performance of different strategies in practical scenarios.

2 BASELINE - BACKGROUND

In the quest to enhance state estimation accuracy for complex natural gas networks, particularly in the presence of measurement noise, a novel data-driven approach has been proposed in the literature. This approach leverages a weighted low-rank approximation framework to efficiently handle the complexities introduced by noise in measurement data, circumventing the need for traditional denoising and filtering methods [1].

2.1 State Estimation

State estimation is crucial in natural gas systems for ensuring operational safety and efficiency. It provides real-time monitoring and predictive insights into gas flows and pressures, facilitating optimal network management. This capability is vital for maintaining system integrity, preventing failures, and enhancing responsiveness to dynamic operational conditions, thereby securing the energy supply chain.

2.2 Data Driven Method

The data-driven method detailed in the referenced paper [1] transforms the state estimation problem into a weighted low-rank approximation problem [2]. This transformation is strategic, aiming to reduce the impact of measurement noise by minimizing the need for explicit noise filtration, thus preserving the integrity of the raw data.

The objective of the weighted low-rank approximation is to estimate the true state of pressures and flows in a natural gas network from noisy measurements. This is achieved by finding matrices U and V that minimize the mean squared error (MSE) between the estimated states and the true states.

Problem Formulation

Given a matrix X of noisy measurements, the goal is to find matrices U and V such that their product approximates X. Mathematically, this is expressed as:

$$\min_{U,V} \|X - UV^T\|_F^2 \tag{1}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, $U \in \mathbb{R}^{m\times r}$, and $V \in \mathbb{R}^{n\times r}$, with r being the rank of the approximation.

Iterative Solution

The algorithm iteratively updates matrices U and V to minimize the Frobenius norm of the difference between X and the product UV^T. The updates are performed using the following least squares solutions:

$$V = \arg\min_{V} \|X - UV^T\|_F^2 \tag{2}$$

$$U = \arg\min_{U} \|X - UV^T\|_F^2 \tag{3}$$

These updates are derived by fixing U and solving for V, and vice versa, in each iteration until convergence. **The specific steps include:**

1. Initialize U and V with random values.
2. Repeat until convergence:
   (a) Update V by solving $V = (U^TU)^{-1}U^TX$.
   (b) Update U by solving $U = XV^T(VV^T)^{-1}$.
3. Compute the mean squared error (MSE) as $\text{MSE} = \frac{1}{n}\|X - UV^T\|_F^2$.

2.3 Reproduction and comparing

In the pursuit of enhancing state estimation techniques for network systems, particularly those involving natural gas distribution, we have explored and implemented two distinct graph-based modeling approaches. These methods leverage the inherent structure and dynamics of network graphs to simulate and predict system behaviors more accurately.

The first approach focuses primarily on nodes by directly modeling both pressures and flows at these junctions[Fig 1]. Here, each node encapsulates the state of the network at a specific point, integrating both pressure data and flow data into a single model. This approach offers a more consolidated view of the network's state at each node, simplifying the complexity involved in managing separate models for flows and pressures. The integration of both sets of data at the node level potentially enhances the accuracy of predictions and state estimations, making it a robust alternative to traditional methods.

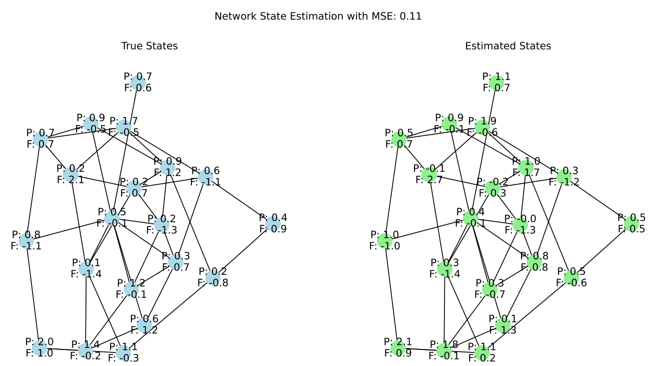

Figure 1: **True and Estimated States- Approach 1 Example for SNR = 10 dB**

In contrast, our second approach we employed [Fig 2] involves the generation of a random graph, where nodes represent critical junctions within the network (such as distribution points or valves) and edges symbolize the connectivity or pipelines between these points. In this model, each edge is assigned a representation of flow dynamics, while node pressures are modeled distinctly. This edge-centric approach allows us to closely simulate the flow dynamics between different

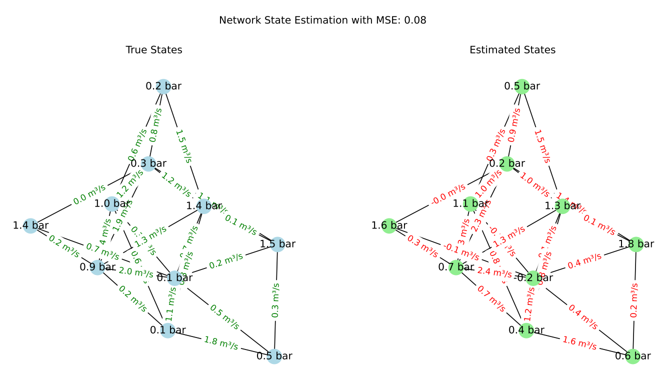

Figure 2: **True and Estimated States- Approach 2 Example SNR = 10 dB**

points in the network, capturing the essence of physical gas flow through pipelines.

Accompanying our modeling efforts, the third figure provides a quantitative analysis of the system's performance across varying levels of signal-to-noise ratio (SNR). This graph plots the mean squared error (MSE) Vs different SNR levels, offering insight into the robustness of our state estimation model under various noise conditions.
As we can see in [Fig 3], low SNR conditions result in significantly higher errors, which tend to increase as expected. Given that real-world scenarios often present less-than-ideal conditions with low SNR's, **our focus has shifted towards these challenging environments**. Consequently, the models we develop and discuss in the subsequent sections are specifically tailored to operate under low SNR conditions, allowing us to thoroughly evaluate their performance in such settings.

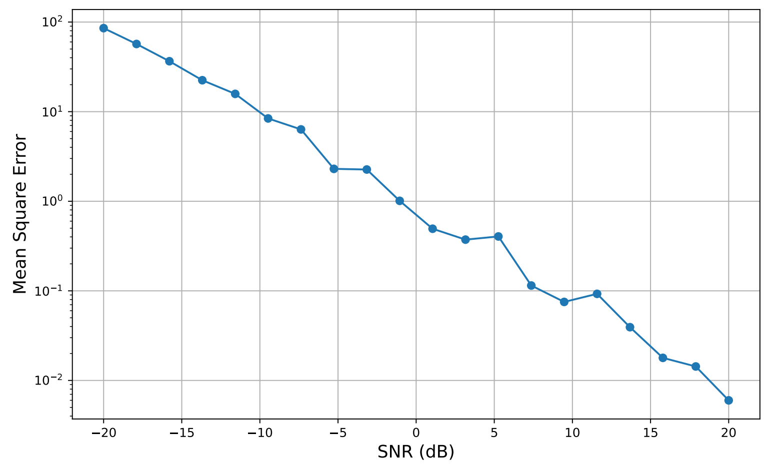

Figure 3: **MSE vs. SNR for Different Scenarios**

3 MODELS - IMPLEMENTATIONS ON NATURAL GAS NETWORKS FOR GSP

To enhance the original algorithm presented in the section 2, we've implemented several algorithms and ideas that will be presented next.

3.1 RWLR Approximation

The regularized low-rank approximation [1, 2] method is designed to estimate the state of a system by approximating a noisy measurement matrix with a product of two lower-rank matrices, thus balancing the fidelity to the data against overfitting. This technique is particularly useful in scenarios where the data may be corrupted by noise [3].

Given a noisy measurement matrix $X \in \mathbb{R}^{m\times n}$, the goal is to find matrices $U \in \mathbb{R}^{m\times r}$ and $V \in \mathbb{R}^{n\times r}$ such that the product UV^T approximates X. Here, r is the rank of the approximation, and it is assumed $r \ll \min(m, n)$ to ensure a reduction in dimensionality.

The algorithm iteratively updates U and V by solving the following regularized least squares problems:

$$V = \arg\min_{V} \left\|U^T U + \lambda \mathbf{I}_r\right\|^{-1} U^T X, \tag{4}$$

$$U = \arg\min_{U} \left\|V^T V + \lambda \mathbf{I}_r\right\|^{-1} XV, \tag{5}$$

where λ is a regularization parameter that controls the trade-off between the fit and the complexity of the model, helping to prevent overfitting by penalizing the magnitudes of the entries in U and V.

λ in low-rank approximation algorithms fundamentally balances data fitting with model simplicity, effectively controlling overfitting by penalizing the magnitude of the matrix factors U and V in the approximation $X \approx UV^T$. While an optimal λ can enhance generalization, reduce numerical instabilities, and facilitate algorithmic convergence, setting it improperly can introduce bias, leading to oversimplification or underfitting. Therefore, the choice of λ requires careful tuning, typically via cross-validation, to optimize the trade-off between bias and variance, ensuring the model captures essential data structures without overfitting to noise. This careful adjustment is crucial, especially in scenarios involving noisy or complex datasets.

The convergence of the algorithm is typically monitored by assessing the change in the Frobenius norm of the difference between successive approximations, and the process is halted when this change falls below a predefined threshold.

This approach is particularly advantageous in handling large-scale data by reducing the computational complexity, making it feasible to apply to large matrices where traditional singular value decomposition would be computationally prohibitive.

3.2 Wavelet Denoising

Wavelet Denoising [4] is a powerful statistical tool used to remove noise from a signal while preserving the original signal characteristics as much as possible. The process exploits the ability of the wavelet transform to compactly represent the signal's structure at multiple levels of detail.

Wavelet Denoising Process

The Wavelet Denoising process involves several key steps:

1. **Wavelet Decomposition:** The signal is decomposed into a series of wavelet coefficients at various scales using a discrete wavelet transform (DWT). This is mathematically represented as:

$$c_{j,k} = \int x(t)\,\psi_{j,k}(t)\,dt, \tag{6}$$

   where $c_{j,k}$ are the wavelet coefficients, $x(t)$ is the input signal, and $\psi_{j,k}$ are the wavelets at scale j and position k.

2. **Noise Estimation:** The standard deviation of the noise, σ, is estimated from the median absolute deviation (MAD) of the finest scale wavelet coefficients:

$$\sigma = \frac{\text{Median}(|c_{J,k} - \text{Median}(c_{J,k})|)}{0.6745}, \tag{7}$$

   where $c_{J,k}$ are the coefficients at the finest scale J.

3. **Thresholding:** A threshold θ is calculated using the estimated noise standard deviation:

$$\theta = \sigma\sqrt{2\log(N)}, \tag{8}$$

   where N is the number of data points in the original signal. Coefficients smaller than this threshold are considered to be dominated by noise and are set to zero or shrunk towards zero:

$$\hat{c}_{j,k} = \begin{cases} c_{j,k} - \theta, & \text{if } c_{j,k} > \theta \\ 0, & \text{if } |c_{j,k}| \leq \theta \\ c_{j,k} + \theta, & \text{if } c_{j,k} < -\theta \end{cases} \tag{9}$$

   using soft thresholding.

4. **Signal Reconstruction:** The signal is reconstructed from the modified coefficients using the inverse discrete wavelet transform (IDWT):

$$\hat{x}(t) = \sum_{j,k} \hat{c}_{j,k}\psi_{j,k}(t). \tag{10}$$

The outcome of this process is a denoised signal that retains the important features of the original signal but with reduced noise. This technique is particularly useful in applications where signal quality is paramount, such as in audio processing or medical imaging.

Following denoising, a regularized low-rank approximation refines the signal further. This approach optimizes the signal representation by enforcing a simpler, lower-dimensional structure through regularization, which penalizes complexity and mitigates overfitting. The entire sequence from noise addition, through denoising, to regularization aims to reconstruct the signal as faithfully as possible, ensuring high fidelity and utility in subsequent analyses.

Techniques such as wavelet denoising and regularized low-rank approximation are modified to leverage the graph structure, ensuring that local features and community structures within the graph are preserved. This integration proves essential in fields where data is inherently relational, offering a robust framework for handling complex datasets.

3.3 Kalman Filter

The Kalman filter [5] is an iterative algorithm widely used in signal processing and control systems to estimate the state of a linear dynamic system from a series of noisy measurements. Unlike the methods discussed in [1] this implementation integrates the Kalman filter to enhance state estimations under noisy conditions, providing a robust alternative that accommodates dynamic changes and uncertainty effectively.

Kalman Filter Algorithm

The algorithm progresses through a series of measurements, updating estimates and reducing uncertainties. The main steps involve:

1. **Initialization:** Set initial estimates for the state and variance:

$$\hat{x}_0 = initial\_estimate, \quad P_0 = process\_variance$$

2. **Prediction Step:** Predict the next state estimate and variance without yet seeing the measurement:

$$\hat{x}_{k|k-1} = \hat{x}_{k-1}, \quad P_{k|k-1} = P_{k-1} + Q \tag{11}$$

where Q is the process noise variance, representing model uncertainty.

3. **Update Step:** Update the estimate with the new measurement:

$$K_k = \frac{P_{k|k-1}}{P_{k|k-1} + R}, \quad \hat{x}_k = \hat{x}_{k|k-1} + K_k(y_k - \hat{x}_{k|k-1}) \tag{12}$$

$$P_k = (1 - K_k)P_{k|k-1} \tag{13}$$

where K_k is the Kalman gain, R is the measurement noise variance, and y_k is the actual measurement.

By incorporating the Kalman filter into the framework for state estimation in natural gas networks, this approach aims to mitigate the limitations observed in the purely data-driven methods from the referenced paper, particularly in scenarios with significant measurement noise. This adaptive filtering method provides a mathematically grounded, probabilistic approach to refining estimates and is especially suitable for systems where the state evolution is subject to uncertainties and external disturbances.

3.4 Extended Kalman Filter

The Extended Kalman Filter (EKF) [6] is an adaptation of the traditional Kalman filter 3.3 designed to handle nonlinearities in the system and measurement models. It extends the linear framework to nonlinear systems by linearizing about the current estimate. This is particularly useful in graph-based models where dynamics may not conform to linear behaviors.

EKF Implementation

The EKF algorithm involves a series of steps where predictions and updates are made iteratively:

1. **State Transition and Prediction:** The state prediction in EKF involves applying the nonlinear state transition model $x_{t+1} = f(x_t, u_t)$ and linearizing it around the current estimate using the Jacobian:

$$x_{\text{pred}} = f(\hat{x}, u), \quad F_t = \left.\frac{\partial f}{\partial x}\right|_{\hat{x},u} \tag{14}$$

$$P_{\text{pred}} = F_t P F_t^T + Q \tag{15}$$

where F_t is the Jacobian of f with respect to x, P is the covariance of the estimate, and Q is the process noise covariance matrix.

2. **Measurement Update:** Similarly, the measurement update involves the nonlinear measurement model $y_t = h(x_t)$ and its Jacobian:

$$H_t = \left.\frac{\partial h}{\partial x}\right|_{x_{\text{pred}}} \tag{16}$$

$$S = H_t P_{\text{pred}} H_t^T + R, \quad K = P_{\text{pred}} H_t^T S^{-1} \tag{17}$$

$$\hat{x} = x_{\text{pred}} + K(y - h(x_{\text{pred}})), \quad P = (I - KH_t)P_{\text{pred}} \tag{18}$$

where H_t is the Jacobian of h with respect to x, R is the measurement noise covariance, K is the Kalman gain, and y is the actual measurement.

These steps are repeated for each time step, with the EKF providing a powerful means to integrate noisy measurements into the state estimates of nonlinear systems. This method is especially effective in graph-based signal processing where the relationship between nodes can influence the state dynamics in complex ways.

3.5 GNN - GCN

Graph Neural Networks (GNNs) with Graph Convolutional Networks (GCNs) [7] are powerful tools for processing data that is inherently structured as graphs. In this implementation, a GNN model utilizing GCNs is developed to handle graph data where nodes represent measurements with inherent noise, specifically pressures and flows in a network.

Model Architecture

The GNN model consists of several components:

- **Graph Data Generation:** The model begins by generating an random graph with specified node features (pressures and flows). These features are then subjected to noise based on a defined signal-to-noise ratio (SNR), simulating real-world data acquisition challenges.
- **Graph Convolutional Networks:** The model comprises two GCN layers:

  1. The first GCN layer processes the noisy node features, applying a rectified linear unit (ReLU) activation function and a dropout technique to manage overfitting and enhance model generalization.

2. The second GCN layer further transforms the features to produce outputs that could be tailored for specific tasks such as regression or classification.

Training and Loss Minimization

The model is trained using the mean squared error (MSE) loss, which is calculated as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2, \tag{19}$$

where $\hat{y}_i$ are the predicted node features by the model, y_i are the actual (target) node features, and n is the total number of nodes in the graph. The MSE loss function encourages the network to predict node features accurately by minimizing the difference between the noisy input features and the model's predictions. Training involves adjusting the weights of the GCN layers to minimize this loss over a series of epochs, using the Adam optimizer for efficient stochastic optimization. .

This implementation of GNN using GCNs is particularly suited for tasks where the data exhibits graph-structured dependencies and noise, making it a robust framework for a wide range of applications in network analysis and beyond.

4 NUMERICAL STUDY

4.1 Common Seed - Dataset

The methodology for modeling the natural gas distribution network in our simulation employs several systematic steps, integrating mathematical formulations to enhance clarity and precision:

1. **Graph Generation:** A random Erdős-Rényi graph $G(n, p)$ with $n = 20$ nodes is generated, where each node represents a critical point in the network such as distribution centers or junctions. Each edge between nodes symbolizes the pipeline connections facilitating gas flow, with the edge existence probability p dictating the network's connectivity density.

2. **Assignment of Node States:**

- *Pressures:* Pressures at each node are simulated as:

$$P_i = |\mathcal{N}(\mu_p, \sigma_p^2)|$$

where P_i is the pressure at node i, and $\mathcal{N}(\mu_p, \sigma_p^2)$ represents normally distributed random values with mean μ_p and variance σ_p^2, ensuring realistic, non-negative readings.

- *Flows:* Flows between nodes, F_{ij}, through the pipelines are modeled using:

$$F_{ij} = \mathcal{N}(\mu_f, \sigma_f^2)$$

for each edge (i, j), reflecting the variable flow rates that occur in real-life scenarios.

3. **Noise Addition:** Gaussian noise is added to the simulated pressures and flows to emulate real-world sensor inaccuracies, modeled as:

$$P_i' = P_i + \epsilon_p, \quad F_{ij}' = F_{ij} + \epsilon_f$$

where $\epsilon_p \sim \mathcal{N}(0, \sigma_n^2)$ and $\epsilon_f \sim \mathcal{N}(0, \sigma_n^2)$ represent the noise added to pressures and flows, respectively.

4. **Simulation Goals:** The primary objective is to validate our state estimation algorithms under realistic conditions, focusing on their performance in environments with low signal-to-noise ratios (SNR):

$$\text{SNR} = 10 \log_{10} \left(\frac{\text{Signal Power}}{\text{Noise Power}} \right)$$

This ensures the robustness and reliability of the models in practical applications.

4.2 Results

We compared three different signal processing models as depicted in the plot from figure 4[Sec.3.1,3.2, 3.3]. This plot illustrates the Mean Square Error (MSE) across varying Signal-to-Noise Ratio (SNR) levels for three methods: Wavelet Denoising, Regularization, and the Kalman Filter. From the plot, it is evident that the Kalman Filter consistently outperforms the other two methods across most SNR scenarios, exhibiting the lowest MSE values, especially in higher SNR conditions. This suggests that the Kalman Filter is more robust and effective in noise reduction and signal estimation.

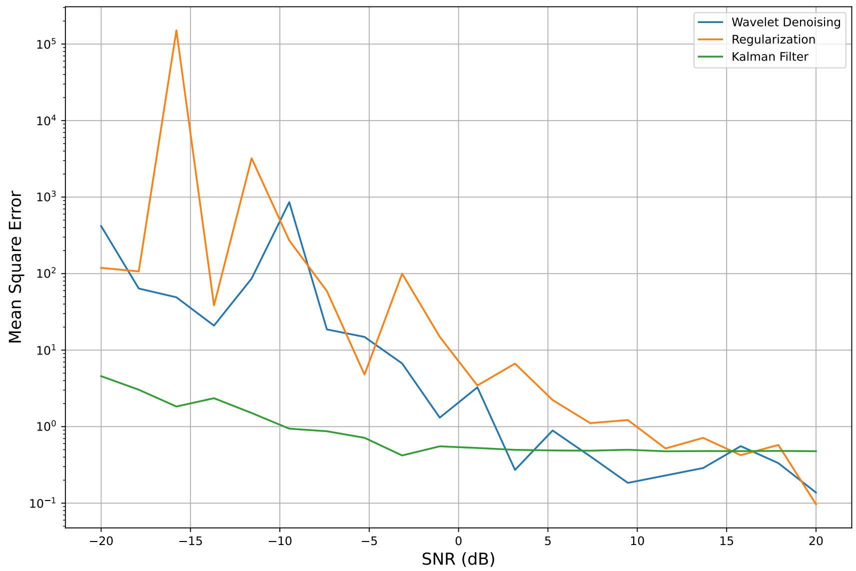

Figure 4: **Comparison of MSE Across Different Methods at Low SNR.**

A deeper evaluation of the Kalman Filter, detailed in figure 5, shows its performance against the original model across a spectrum of SNR values. The graph highlights a significant improvement in MSE reduction when utilizing the Kalman Filter, affirming its superiority over the conventional model. This comparative analysis underlines the Kalman Filter's enhanced capability to handle variabilities within the signal effectively, making it an optimal choice for applications requiring high precision in noisy environments.

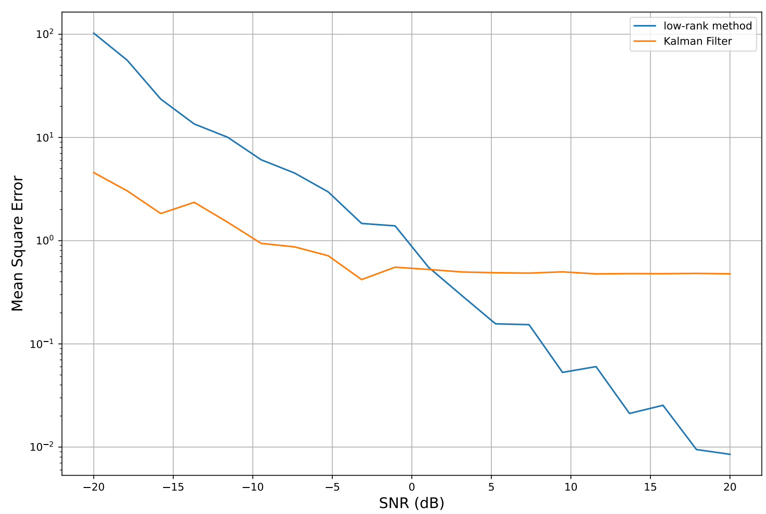

Figure 5: **Comparative MSE Analysis: Kalman Filter vs. Original Method in Low SNR.**

Figure 6 illustrates the MSE reduction over epochs for a Graph Convolutional Network (GCN) model trained under a low SNR = -10 dB [sec. 3.5] . This plot captures the model's progression in learning to mitigate noise effects on graph-based features such as pressures and flows.

As training advances, the consistent decrease in MSE highlights the GCN's ability to refine its predictive accuracy, showcasing effective adaptation to noisy environments. This trend is crucial for applications requiring precise signal estimation in noise-prevalent scenarios.

Finally, the performance of the Extended Kalman Filter (EKF) is specifically examined. As per the data presented in **Table 1**, the EKF model shows varying levels of MSE at different SNR dB levels: 0.35 at -10 dB, 0.22 at 1 dB, and 0.1 at 10 dB. This progression indicates an improvement in the model's performance as the SNR increases, demonstrating the EKF's adaptability and efficiency in higher quality signal conditions. These results confirm the EKF's practical utility in scenarios where maintaining accuracy with changing SNR levels is crucial.

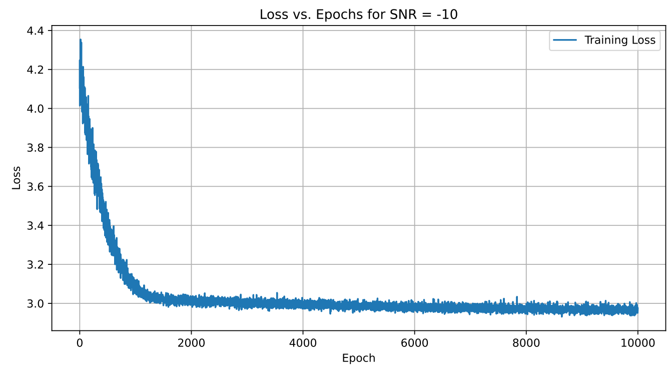

Figure 6: **Training Loss Evolution for GNN-GCN Model at Low SNR**

SNR [dB]	**-10 dB**	**1 dB**	**10 dB**
MSE	0.35	0.22	0.1

Table 1: MSE vs SNR for the EKF model [Sec 3.4]

The analyses conducted across various signal processing models highlight the Kalman Filter and its Extended version as notably effective in managing noise and enhancing signal estimation in environments characterized by variable or high SNR's. These models demonstrate robust performance, particularly noteworthy in high SNR conditions where they consistently outperform other methods like Wavelet Denoising and Regularization.

Moreover, the implementation of a Graph Convolutional Network (GCN) within a Graph Neural Network (GNN) framework has shown promising results in the adaptive filtering landscape. The MSE reduction observed during the GCN-GNN model's training underscores its capability to effectively learn and adapt to complex noise patterns within network data. This is evident from the progressive decrease in MSE over 10,000 training epochs, confirming the model's ability to enhance prediction accuracy even in low SNR conditions, which are typically challenging for traditional filtering approaches.

Unlike the original method, which accepts noisy measurements as given without attempting any filtration to preserve potential informative content, our approach actively filters noise from the data. This proactive noise management aims to clarify signal interpretation while being mindful to avoid discarding valuable information. It is crucial that our filtering strategy is carefully calibrated to ensure that essential data is not inadvertently removed during the noise reduction process.

5 APPLICATION - POWER SYSTEMS

The methodologies showed and tested within this project have direct applications in power systems, particularly in enhancing the reliability and accuracy of state estimations in natural gas networks. By integrating graph signal processing and advanced noise management techniques, these approaches offer significant improvements in the robust management of network dynamics, crucial for maintaining energy supply reliability and operational safety in real-world scenarios.

6 CONCLUSIONS

The project successfully demonstrates the effectiveness of advanced signal processing models, including Kalman and Extended Kalman Filters, and GCN-GNN frameworks, in managing noise and enhancing signal estimation under various SNR conditions. These models show promising capabilities in handling complex noise patterns and improving state estimation accuracy, making them invaluable in practical applications where precision and reliability are paramount.

Future work can include further optimization of these techniques to handle lower SNR levels and explore their broader applications across different noisy and dynamic environments.

References

[1] Y. Huang, L. Feng, and Y. Liu, "A data-driven state estimation framework for natural gas networks with measurement noise," *IEEE Access*, vol. 11, 2023.

[2] K. Werner and M. Jansson, "Weighted low rank approximation and reduced rank linear regression," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2004, pp. 501–504.

[3] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[4] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.

[5] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[6] G. Sagi, N. Shlezinger, and T. Routtenberg, "Extended kalman filter for graph signals in nonlinear dynamic systems," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2023.

[7] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.