Indian Institute of Technology Ropar

Term Project

EE309: Power Systems

Shashwat Srivastava 2021EEB1210

Yash Agarwal 2021EEB1225

Ranjeet Singh 2021EEB1203

Abhinav Barman 2021EEB1145

Gaurav Wani 2021EEB1171

Submitted to

Dr. Ranjana Sodhi

May 9, 2024

Acknowledgement

We would like to express our deepest gratitude and appreciation to Professor **Dr. Ranjana Sodhi** and **Manish Sir** for granting us the opportunity to work on this project. Their unwavering support, vast knowledge, and mentorship have been invaluable in guiding us through the complexities of the project. We are grateful for their exceptional ideas and expertise, which contributed to the successful completion of the project. Their continuous support, constructive feedback, and innovative ideas have not only helped us hone our skills but also broadened our understanding. We consider ourselves fortunate to have had the privilege of working under their guidance and have gained invaluable insights from the experience.

1 Introduction

The rise of devices like phasor measurement units (PMUs) has led to an increase in synchrophasor data, presenting challenges in storing and processing it efficiently. The paper we chose suggests a way to compress synchrophasor data using a method based on singular value decomposition (SVD). This method evaluates the data's dimensions and reduces them, considering measurement uncertainty and signal-to-noise ratio (SNR). It retains high-SNR modes and discards those affected by errors to compress the data while keeping crucial information intact. Additionally, it uses a real-time algorithm to partition data based on normal and disturbance conditions, further increasing accuracy and compression ratio (CR).
The use of synchrophasor data has grown rapidly with the increasing deployment of phasor measurement units (PMUs) in power systems. To manage this vast amount of data effectively, advanced techniques for storing and handling big data are crucial. The high reporting rates of synchrophasor data, particularly for wide-area applications involving numerous PMUs, generate a massive volume of data that needs to be archived for analysis and validation.
Data compression is a method to reduce the size of large datasets for storage or transmission purposes. Compression techniques are typically categorized as lossy or lossless. Lossless methods aim to compress data without losing any information, but they struggle to achieve the high compression ratios needed for wide-area synchrophasor networks. On the other hand, lossy techniques can achieve high compression ratios but at the expense of some accuracy in the reconstructed data.

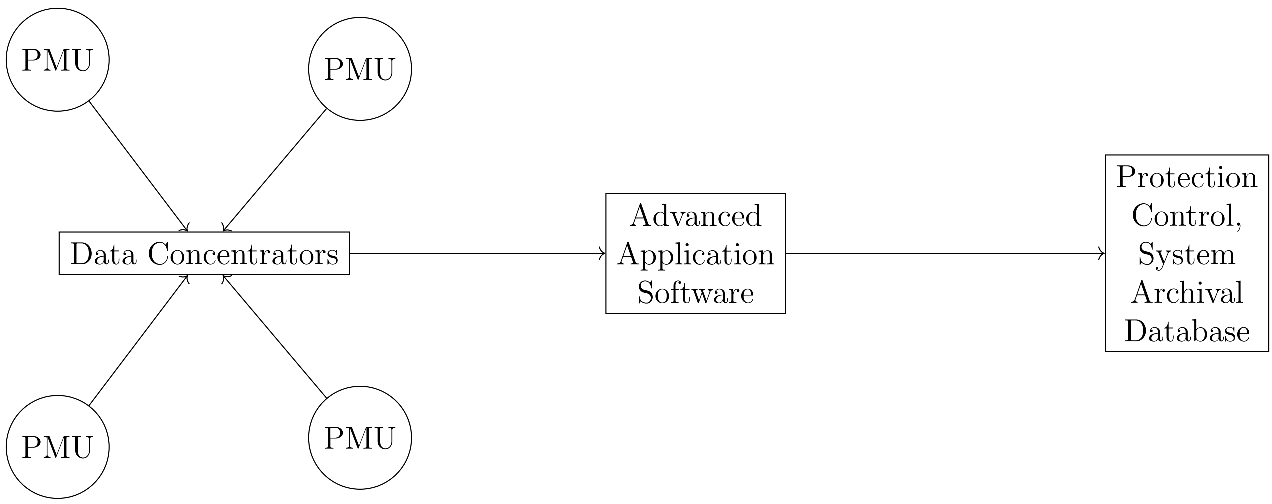

Figure 1: Application of PMUs in power system

Various compression methods have been proposed for synchrophasor data, particularly those based on principal component analysis (PCA) and singular value decomposition (SVD). These methods exploit the correlations and low-rank properties of synchrophasor data to reduce its dimensionality. However, there are practical challenges in determining the reduced dimension and partitioning the data effectively.

The given paper introduced a new compression method based on SVD for both real and complex-valued PMU data. It had two main parts: a technique for reducing dimensionality and a real-time partitioning algorithm. Unlike existing methods, the dimensionality reduction technique considers measurement uncertainties and does not require iterative processing. The progressive partitioning algorithm monitors changes in dimensionality in real time and separates data into normal and event conditions. It further divides disturbance data into high and low dimensionality partitions based on severity. This approach eliminates the need for user-defined thresholds, prevents mixing data from different conditions, and ensures high compression ratios and accuracy.

The proposed compression method is evaluated using simulated PMU data, showing improvements in compression ratio without losing critical information compared to existing methods.

2 Singular Value Decomposition (SVD)

The Singular Value Decomposition (SVD) is a powerful matrix factorization technique that plays a crucial role in various applications, including data compression, dimensionality reduction, and signal processing. It decomposes a matrix X of size $h \times n$ into three constituent matrices:

$$X = U\Sigma V^* \tag{0.1}$$

where:

- U is an orthogonal matrix of size $h \times h$,
- Σ is a diagonal matrix of size $h \times n$ containing the singular values of X arranged in descending order,
- V^* is the conjugate transpose of an orthogonal matrix V of size $n \times n$.

The diagonal entries of Σ represent the singular values of X, which quantify the importance of each singular vector in expressing the matrix X. The SVD provides a natural way to approximate X by retaining only the most significant singular values and their corresponding singular vectors.

One common application of SVD is rank-r approximation, where X is approximated by a matrix $\hat{X}$ of rank r:

$$\hat{X} = \sum_{i=1}^{r} \sigma_i u_i v_i^* \tag{0.2}$$

where σ_i are the r largest singular values, and u_i and v_i are the corresponding left and right singular vectors, respectively.
This approximation captures the dominant features of the original matrix X while reducing its dimensionality. It finds extensive use in data compression and denoising, especially when dealing with large datasets.

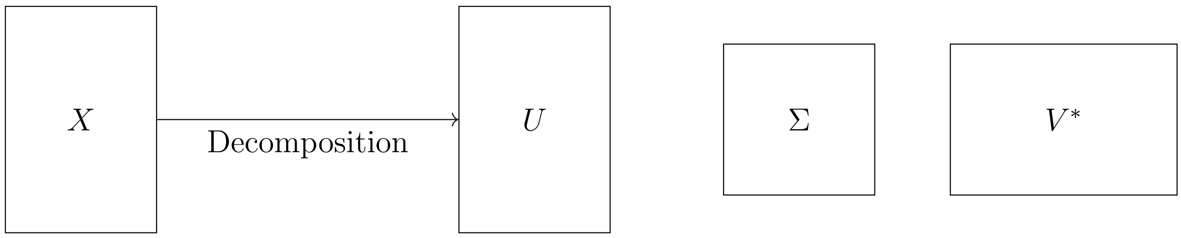

Figure 2: Decomposition of matrix X into its constituent matrices

SVD vs PCA

Principle Component Analysis is a linear transformation technique that projects high-dimensional data onto a lower-dimensional space while preserving the maximum amount of variance. It works by finding the principal components of the data, which are the linear combinations of the original features that capture the most variation in the data.

I. Unlike PCA, SVD is capable of capturing both magnitude and phase information inherent in the data, which is crucial for accurate representation in power systems applications. Moreover, SVD's ability to adaptively select singular vectors based on threshold criteria allows for better preservation of important information while compressing the data, addressing the challenge of maintaining fidelity in the presence of uncertainties and disturbances.

II. Although PCA offers efficient dimensionality reduction by projecting data onto orthogonal principal components, it may not be able to handle the inherent complexities and uncertainties of synchrophasor data. From various papers, we came up with the point that, for numerical reasons, it is preferred to use SVD. As it doesn't need to compute the covariance matrix, which can introduce some numerical problems. Because there are some pathological cases where the covariance matrix is very hard to compute. So, the SVD is numerically more efficient. Moreover, PCA requires additional mathematical operations, such as centering the data and calculating the covariance matrix.

III. Although, we were also given a paper in which an iteration enhanced phasor PCA-based compression technique that can also deal with the synchrophasor data in complex format was there. It calculates the total vector error (TVE) of the reconstructed data with respect to the observed measurements and increases the dimensionality until the reconstruction error falls below a threshold. However, in this method, the choice of the threshold is experimental, and the accuracy of PMU measurements is not taken into account. Whereas in the proposed SVD-based method, the PMU measurement uncertainty information is used to establish a new threshold criterion to reject SVD modes with a low signal-to-noise ratio (SNR).

3 Dimensionality Evaluation & Reduction using SVD

Let $X \in \mathbb{C}^{h \times n}$ be the **ground-truth data matrix** at time t.

$$X = [x_1 \; x_2 \; x_3 \; \dots \; x_n] \tag{1}$$

with,

$$x_i = \left[x_i \left(t - \frac{h-1}{F_s} \right) \dots \; x_i \left(t - \frac{1}{F_s} \right) \; x_i \right] \tag{2}$$

where x_i with $i = [1, \dots, n]$ are **temporal data vectors** representing the i-th column of matrix X, each containing h elements corresponding to different time instances.
The observed data reported by the PMUs is given by:

$$Y = X + E \tag{3}$$

with,

$$Y = [y_1 \; y_2 \; y_3 \; \dots \; y_n], \; E = [e_1 \; e_2 \; e_3 \; \dots \; e_n] \tag{4}$$

where $E \in \mathbb{C}^{h \times n}$ is the **measurement error matrix**, predominantly caused by the limited accuracies of the PMUs and other measurement instruments.
Given $m = \min(h, n)$, the reduced SVD (RSVD) of Y is:

$$Y = U \sum V^{\mathrm{H}} = \sum_{r=1}^{m} u_r \sigma_r v_r^H \tag{5}$$

where Σ is a diagonal matrix with diagonal entries $\sigma_1, \dots, \sigma_m$ representing the singular values of Y. Moreover, without loss of generality, we assume that $\sigma_1 \geq \dots \geq \sigma_m \geq 0$. The matrices $U \in \mathbb{C}^{h \times m}$ and $V \in \mathbb{C}^{n \times m}$ are orthogonal, and $(\cdot)^H$ is the conjugate transpose operator.
The RSVD decomposes Y into m modes with mode r consisting of a singular value $\sigma_r =$ (r,r) (mode magnitude), a left singular vector $u_r = U(:, r)$ (temporal mode shape), and a right

singular vector $v_r = V(:,r)$ (spatial mode shape). The left and right singular vectors are normalized such that $u_r^{\mathrm{H}} u_r = 1$ and $v_r^{\mathrm{H}} v_r = 1$. Moreover, the contribution of the r-th SVD mode to Y is defined as $Y_r \triangleq u_r \sigma_r v_r^{\mathrm{H}}$.

The best rank-ρ approximation of Y can be constructed using the first ρ SVD modes (lower modes) of Y. Therefore, the rank-ρ approximation of Y can be written as:

$$\hat{Y} = \sum_{r=1}^{\rho} Y_r = \sum_{r=1}^{\rho} u_r \sigma_r v_r^H \tag{6}$$

where $\rho \in \{1, \ldots, m\}$ is the number of modes used to construct $\hat{Y}$. The reconstruction error matrix is defined as $\hat{E} \triangleq Y - \hat{Y}$.

To efficiently determine ρ, we evaluate whether the magnitude of an SVD mode is larger than that of the measurement error E. Therefore, we only keep the SVD modes that satisfy the following condition.

$$Y_{rRMS} > E_{RMS},\ r = 1, \ldots, m \tag{7}$$

where Y_{rRMS} and E_{RMS} are the RMS values of Y_r and E, respectively. This condition can also be expressed in terms of the SNR of the r-th mode as,

$$SNR_r = 10\, log_{10} \left(\frac{Y_{rRMS}}{E_{RMS}} \right)^2 > 0\, dB \tag{8}$$

which implies that only the modes with positive SNR should be used in the construction of rank-ρ approximation of Y.

By noting that $Y_r = u_r \sigma_r v_r^{\mathrm{H}}$, the RMS contribution of the rth mode can be calculated as,

$$\begin{aligned} Y_{rRMS} &= \sqrt{\frac{1}{hn} \sum_{i=1}^{h} \sum_{j=1}^{n} |Y_r(i,j)|^2} \\ &= \sqrt{\frac{\sigma_r^2}{hn} \sum_{i=1}^{h} \sum_{j=1}^{n} |u_r(i) v_r^*(j)|^2} \\ &= \frac{\sigma_r}{\sqrt{hn}} \sqrt{\sum_{i=1}^{h} |u_r(i)|^2 \sum_{j=1}^{n} |v_r^*(j)|^2} \\ &= \frac{\sigma_r}{\sqrt{hn}} \sqrt{u_r^H u_r} \sqrt{v_r^H v_r} \\ &= \frac{\sigma_r}{\sqrt{hn}} \end{aligned} \tag{9}$$

Moreover, we assume that E is a random process with

$$E_{RMS} \leq \epsilon \tag{10}$$

So, we must determime the maximum possible error in measurements, often called RMS measurement error. Manufacturers typically assess the accuracy of Phasor Measurement Units (PMUs) using various performance metrics. These metrics help estimate the maximum error for different types of signals. If multiple devices, like PMUs, contribute to the error, we calculate the total error using the triangle inequality.

$$\epsilon \leq \sum_{i=1}^{n_e} \epsilon_i \tag{11}$$

This calculation involves summing up the individual RMS errors of each device. This total error serves as a threshold for our analysis. Since the information on the accuracies of the devices is not available (we used simulated data), one can use the maximum acceptable errors defined by IEEE standards.

Therefore, $Y_{rRMS} > E_{RMS}$ is satisfied if $Y_{rRMS} = \frac{\sigma_r}{\sqrt{hn}} > \epsilon$. The proposed threshold criterion is then:

$$\rho = \max(r), \text{ such that } \sigma_r > \epsilon\sqrt{hn} \tag{12}$$

Thus, ρ is the required number of SVD modes to construct the approximation.

The CR represents the reduction in the physical space required for storing the data. The number of measurement points in Y is $h \times n$. However, using the rank-ρ estimate $\hat{Y}$, and considering the sparsity of $\sum$, the number of data points to be stored becomes $\rho \times (h + n + 1)$ which yields a CR of

$$CR = \frac{h \times n}{\rho \times (h + n + 1)} \tag{13}$$

It should be noted that to achieve a CR larger than one, the reduced dimensionality should satisfy,

$$\rho \leq \left\lfloor \frac{h \times n}{(h + n + 1)} \right\rfloor \tag{14}$$

where the $\lfloor \cdot \rfloor$ function is the floor function.

Moreover, the root mean square error (RMSE) and maximum absolute deviation error (MADE) are defined as:

$$\text{RMSE} = \frac{\|Y - \hat{Y}\|_F}{\sqrt{hn}} = \frac{\|\hat{E}\|_F}{\sqrt{h \times n}} \tag{14}$$

$$\text{MADE} = \max|Y - \hat{Y}| = \max|\hat{E}| \tag{15}$$

can be used to evaluate the accuracy of the compression method with respect to the original observations Y. The term $\hat{E}$ is the Frobenius norm of the reconstruction error. RMSE represents the dissimilarity between the observed and reconstructed data on average, and MADE shows the worst reconstruction error among all data points. Higher CR and lower RMSE and MADE represent a more efficient compression.

4 Real-Time Progressive Data Partitioning

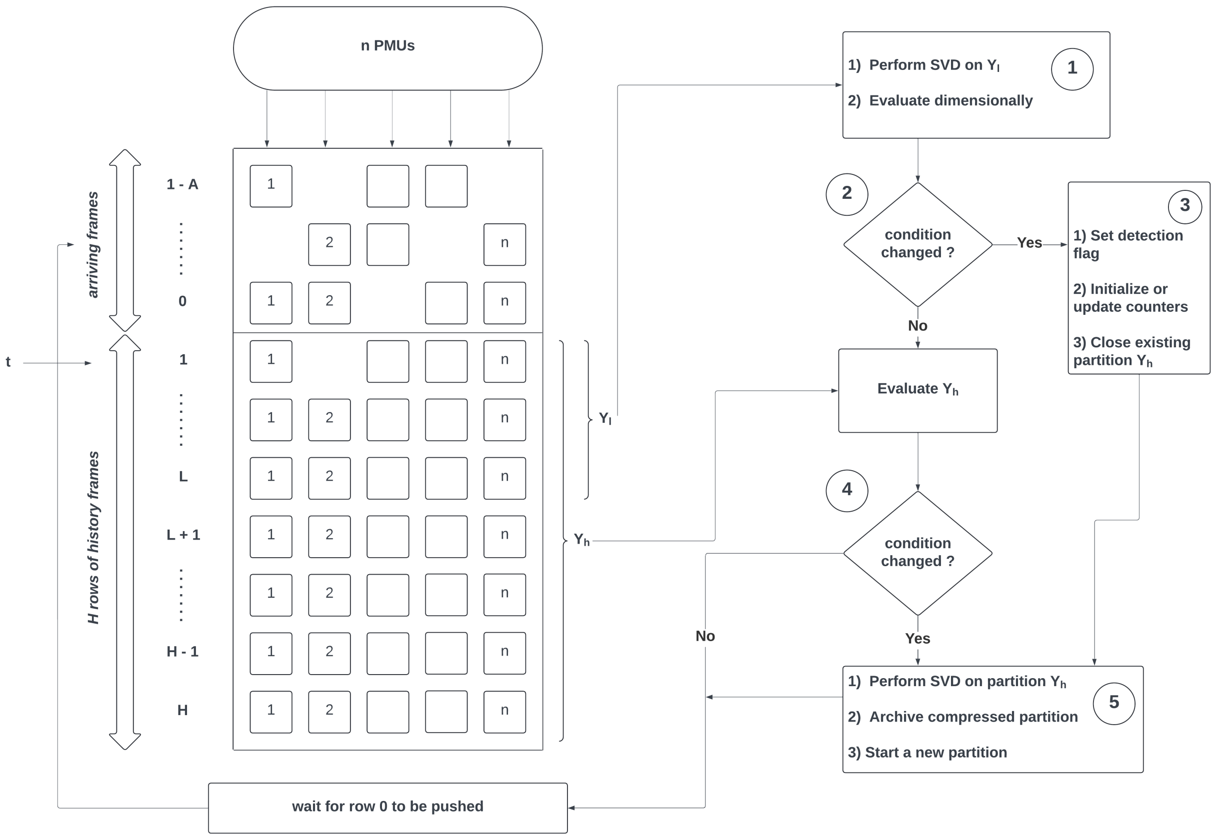

Figure 3: Outline of PDC as a Circular Buffer

The existing "classical partitioning" technique made the partition in the data based on the disturbance present. This meant that the rank for normal and event data was evaluated separately. However, since the disturbance could be severe or minor and also for different time intervals, this method of partitioning leads to overestimation or underestimation of the rank for event data. To counter this problem, we implemented the Progressive Partitioning Algorithm proposed in the given paper. PDC is constructed using a circular buffer, which collects data in the buffer. The rows correspond to the data at that particular time step and the columns refer to the values from different PMUs.

In Progressive Partitioning, the data is not only split between normal and event data, but the event data is further split into severe and less severe disturbances. This leads to proper estimation of the rank in these corresponding situations. The following flowchart gives an overview of this method.

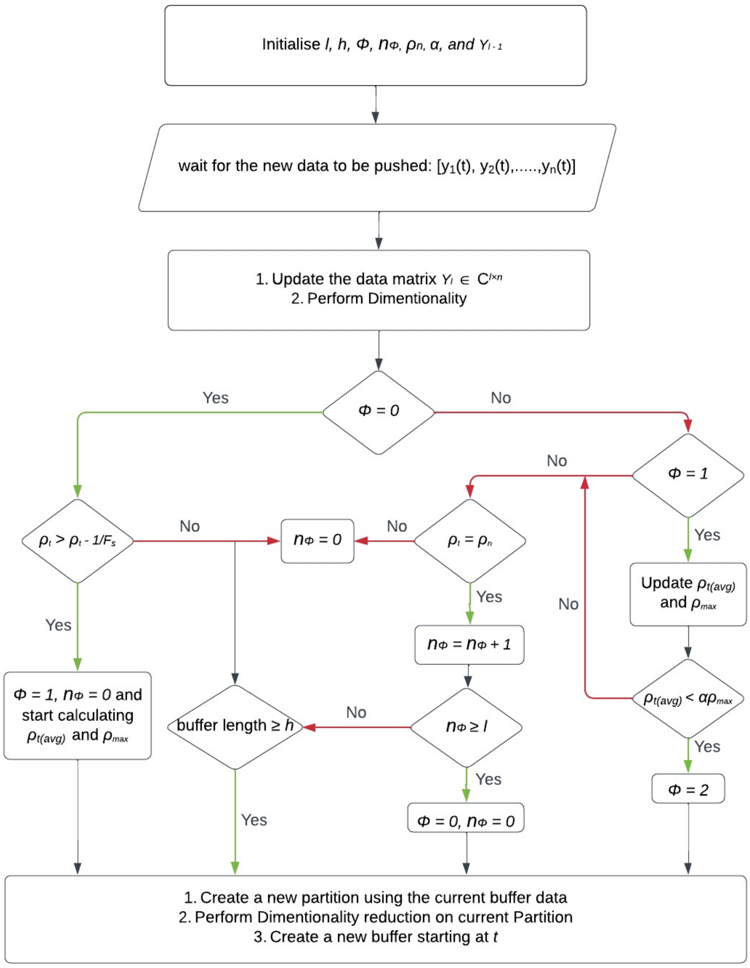

Figure 4: Flowchart for Progressive Partitioning

First of all, this algorithm takes the buffer size l and h, detection flag ϕ, α, and sliding matrix Y_l, normal rank ρ_n as inputs. These inputs are to be chosen such that this algorithm works effectively. The value of ϕ is first assumed to be zero. The data is pushed into Y_l, and rank is calculated for that timestep t. If no change of rank is detected from its previous timestep, ϕ is still kept as zero, and more data is awaited in the buffer. In case the entire buffer is filled up till Y_h, this data is marked as a normal partition and passed to data compression.

If at $\phi = 0$, the rank at any timestep increases from its previous timestep, then ϕ is set to 1. Now, it calculates the average estimated rank ρ_{avg} as well as maximum rank ρ_{max}. Also, as a

safety margin, the last l/2 rows are kept stored to prevent errors during compression, and the rest of the rows are sent as a normal partition for compression.
While ϕ is non-zero, the average estimated rank is continuously compared with ρ_{max}. If $\rho_{avg} \leq \alpha\rho_{max}$ where $0 \leq \alpha \leq 1$, ϕ is set to 2, and the resultant partition is sent for compression. This partition is done to separately compress severe and damping disturbances.
While $\rho_{avg} \geq \alpha\rho_{max}$ and $\phi \neq 0$, the algorithm counts the number of times ρ_t has decreased to ρ_n and is stored in n_ϕ.If $n_\phi \geq l$, the current partition is sent for compression and new buffer is started. Hence, this algorithm helps us partition data in terms of the degree of disturbance and helps us accurately compress.

5 Simulated Data Used

Here, we present the simulated data used in our study. The simulated results include measurements of voltage magnitude, voltage angle, and frequency for two different cases.

1. **Fault at Bus-14:** In the first case, we simulate a fault occurring at Bus-14.

2. **Load outage at Bus-12:** In the second case, we simulate a load outage at Bus-12.

For each case, the simulation is carried out for a duration of 20 seconds. During this time, measurements are taken at a frequency of 60 samples per second. This results in a total of 1200 samples being collected from each bus. In both cases, faults occur at 2 seconds.
We also use a third type of data, 180 sec of voltage magnitude data in the IEEE 14-bus system. There is Line 9-14 fault at t=55sec for a duration of 0.05sec i.e. 3 cycles of the system frequency of 60Hz. Additionally, we took $\epsilon_i = 3.29 \times 10^{-4}$ as given in the research paper, and the SNR added to the ground truth matrix to be 92 dB because in most of the works, this SNR was used.
The figure below illustrates the IEEE-14 bus system.

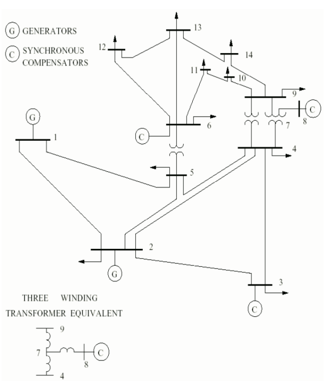

Figure 5: IEEE-14 Bus System

6 Numerical Results

Fig. 6 here shows the actual frequencies for the buses against time. We can see that the disturbance starts at around 2 sec, and at around 5 seconds, it settles down.

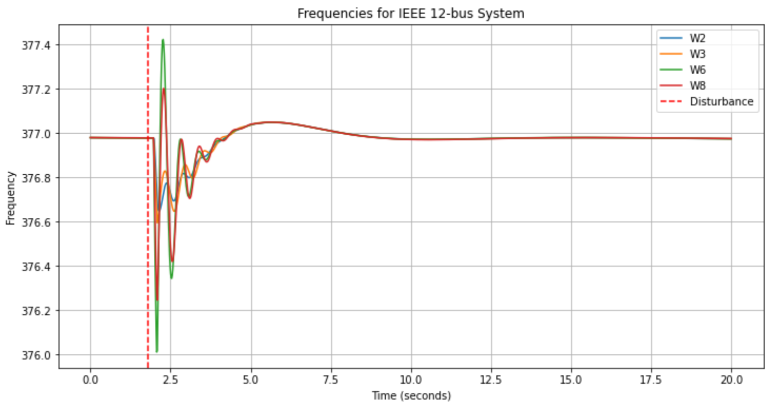

Figure 6: Bus frequencies against time

Here, we present the rank estimation graph against time. The flag φ is zero before the fault and after the disturbances are completely damped. It is set to $\varphi = 1$ after the fault occurs and changes to $\varphi = 2$ when severe disturbances are passed. Pre-fault, we approximate the data of normal conditions with a rank-1 matrix. During the event, whereas the progressive method estimates the first part (severe disturbances) and the second part (damped disturbances) with rank-4 and rank-3 matrices, respectively. This confirms that progressive partitioning does not lead to the underestimation and overestimation of the rank during severe and less severe disturbances, respectively.

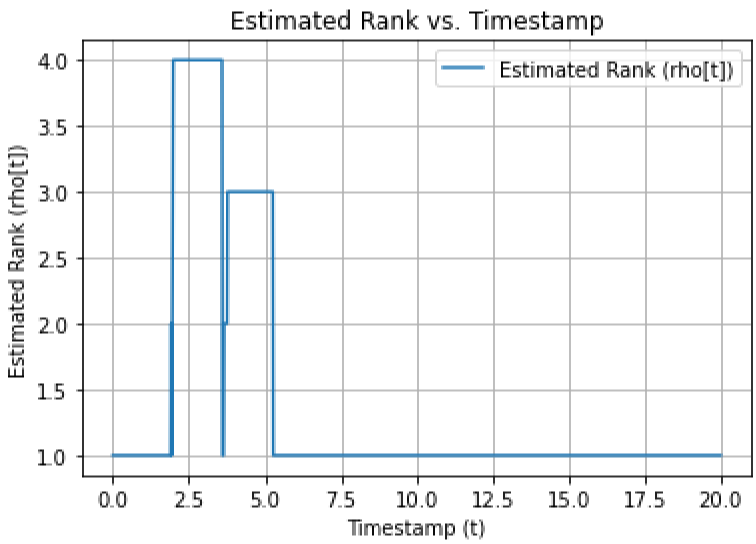

Figure 7: Estimated rank against time

The below figure is used to compare the original and reconstructed data; we can see that the reconstructed data clearly follows the original data, which further cements our confidence in this solution.

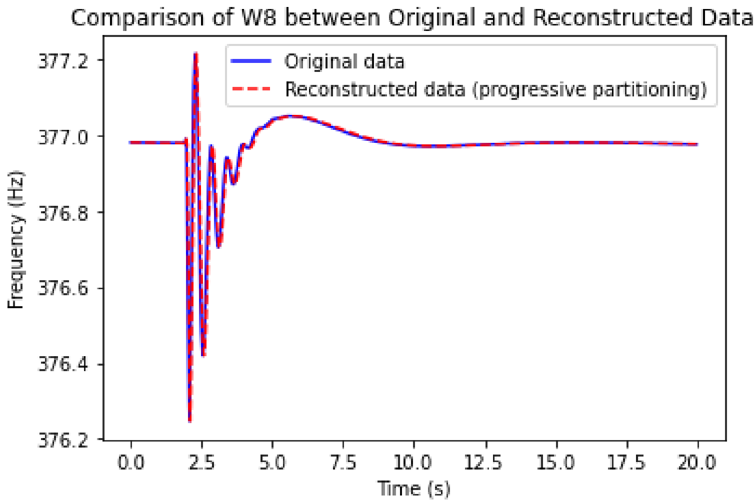

Figure 8: Comparison of original data with reconstructed data

Here, we present our results for Error and Compression ratio against IEEE-12 & 14 bus system for all three types of signals. We used RMSE & MADE here for performance evaluation.

Table 1: Comparison of RMSE, MADE, and CR for IEEE-14 and IEEE-12 buses

IEEE bus	**Signal**	**RMSE**	**MADE**	**CR**
IEEE-14 (20 sec)	frequency	0.00012	0.00182	2.689
	angle	0.00043	0.01596	5.8679
	voltage	7.93e-5	0.00145	5.132
IEEE-12 (20 sec)	frequency	0.00021	0.00205	3.0476
	angle	6.42e-5	0.00079	5.52995
	voltage	0.00027	0.00317	6.1172
IEEE-14 (180 sec)	voltage	4.6e-5	0.00572	8.44776

7 Limitations

The main computational burden of the partitioning and compression procedures is the calculation of SVD, which highly depends on the number of data points in data matrices Y_l and Y_h. The dimensions of the data matrices can change with the number of PMUs n and the selected parameters l and h. The number of rows in the data matrices also depends on the reporting rate f_s.

For 180-second data, it takes approximately 9 seconds, for the entire compression process currently takes approximately 9 seconds. However, this duration exceeds the expected time, indicating a need for optimization or refinement in the compression methodology to achieve faster processing times while maintaining accuracy and effectiveness.

Why lower CR than Research Paper?

As we know,

$$CR = \frac{h \times n}{\rho \times (h + n + 1)} \tag{15}$$

So, CR increases for a given rank if we increase h and n. Since we worked on IEEE-14 & 12 data, which has lower PMUs and hence lower n. Also, the simulation time was limited to 20 seconds which means we can't have really big h here. On the contrary, authors of the paper took IEEE-118 bus systems with their simulation running for over 400 seconds, hence they were able to show such high compression ratios.
As we increased the simulation time to 180 seconds and increase h as well, we get an increased compression ratio.

8 Future Work

Randomized SVD

When dealing with large datasets, computing the regular Singular Value Decomposition (SVD) can become prohibitively expensive. This challenge becomes more pronounced with the increasing volume of data generated by Phasor Measurement Units (PMUs) in modern power systems. However, amidst this sea of data, there often exists inherent structures or patterns that we are interested in capturing. These patterns can often be characterized by a low intrinsic rank, denoted as r, which signifies the number of dominant components essential for describing the dataset effectively.
To address the computational burden associated with traditional SVD, particularly for large datasets, a randomized approach can offer significant efficiency gains. In this approach, instead of directly computing the SVD of the entire dataset X, we employ a two-step process that leverages random projections to capture the essential information efficiently.
Step 1: A random projection matrix $P \in \mathbb{R}^{m \times r}$ is generated. This matrix is used to project the dataset X onto a lower-dimensional space. The resulting matrix Z is obtained by multiplying X with P. To ensure numerical stability and efficient computation, the matrix Z is then decomposed into an orthogonal matrix Q and an upper triangular matrix R, using the QR decomposition. Remarkably, this step not only yields an orthonormal basis for the projected

space Z but also for the original dataset X, facilitating subsequent computations.
Step 2: The dataset X is projected onto the orthonormal basis Q obtained from the previous step. This projection results in a new dataset Y, which captures the essential information of X in a reduced-dimensional space. Subsequently, the Singular Value Decomposition is performed on Y, yielding matrices U_Y, Σ, and V, which are analogous to those obtained from the SVD of X. Notably, the matrix U_X, which represents the left singular vectors of X, can be efficiently reconstructed as the product of Q and U_Y.
By adopting this randomized approach, we can effectively reduce the computational complexity associated with computing the SVD of large datasets while preserving the essential structure captured by the dominant singular components. This method offers a scalable and efficient alternative for analyzing massive datasets generated by PMUs, enabling expedited insights into power system dynamics and facilitating real-time monitoring and control applications. We can do this by researching on SVD and its optimized variants.

9 Conclusion

We implemented a new data compression method based on the singular value analysis of PMU data streams. The compression method includes a dimensionality reduction technique and a progressive partitioning algorithm. The dimensionality reduction algorithm uses the PMU uncertainties to establish a threshold criterion upon which the validity of SVD modes is verified. The progressive partitioning algorithm splits the synchrophasor data into partitions with coherent dimensionalities. These partitions are separately compressed to improve both the accuracy and CR under different system conditions. The proposed compression method is applicable to all synchrophasor data types, i.e., magnitude, phase angle, and even complex phasor. The performance of the proposed method is evaluated using simulation PMU data involving two-three types of events and disturbances. The results confirm that the proposed compression method is highly reliable and provides high CR and accuracy. In the upcoming PS lab, we will work with randomized SVD and try to improve this work further.

Bibliography

[1] R. Pourramezan, R. Hassani, H. Karimi, M. Paolone and J. Mahseredjian, "A Real-Time Synchrophasor Data Compression Method Using Singular Value Decomposition," in IEEE Transactions on Smart Grid, vol. 13, no. 1, pp. 564-575, Jan. 2022, doi: 10.1109/TSG.2021.3114585.

[2] F. Zhang, X. Wang, Y. Yan, J. He, W. Gao and G. Chen, "A Synchrophasor Data Compression Technique With Iteration-Enhanced Phasor Principal Component Analysis," in IEEE Transactions on Smart Grid, vol. 12, no. 3, pp. 2365-2377, May 2021, doi: 10.1109/TSG.2020.3046666.

[3] Youtube lectures by Steven Brunton on Singular Value Decomposition

[4] Data Driven Science and Engineering (book) by Steven Brunton

[5] Gregory Gunderson's blog on Randomized SVD