smart cities

MDPI

Article

Machine Committee Framework for Power Grid Disturbances Analysis Using Synchrophasors Data

Haoran Niu [1], Olufemi A. Omitaomu [1,2,*] and Qing C. Cao [1]

[1] College of Engineering, University of Tennessee, Knoxville, TN 37996, USA; hniu1@vols.utk.edu (H.N.); qcao1@utk.edu (Q.C.C.)
[2] Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA
* Correspondence: omitaomuoa@ornl.gov

Abstract: Events detection is a key challenge in power grid frequency disturbances analysis. Accurate detection of events is crucial for situational awareness of the power system. In this paper, we study the problem of events detection in power grid frequency disturbance analysis using synchrophasors data streams. Current events detection approaches for power grid rely on individual detection algorithm. This study integrates some of the existing detection algorithms using the concept of machine committee to develop improved detection approaches for grid disturbance analysis. Specifically, we propose two algorithms—an Event Detection Machine Committee (EDMC) algorithm and a Change-Point Detection Machine Committee (CPDMC) algorithm. Both algorithms use parallel architecture to fuse detection knowledge of its individual methods to arrive at an overall output. The EDMC algorithm combines five individual event detection methods, while the CPDMC algorithm combines two change-point detection methods. Each method performs the detection task separately. The overall output of each algorithm is then computed using a voting strategy. The proposed algorithms are evaluated using three case studies of actual power grid disturbances. Compared with the individual results of the various detection methods, we found that the EDMC algorithm is a better fit for analyzing synchrophasors data; it improves the detection accuracy; and it is suitable for practical scenarios.

Keywords: frequency disturbance events; situational awareness; phasor measurement units; event detection; anomaly detection; machine committee; smart grid; artificial intelligence

check for updates

Citation: Niu, H.; Omitaomu, O.A.; Cao, Q.C. Machine Committee Framework for Power Grid Disturbances Analysis Using Synchrophasors Data. *Smart Cities* **2021**, *4*, 1–16. https://dx.doi.org/10.3390/smartcities4010001

Received: 19 November 2020
Accepted: 20 December 2020
Published: 22 December 2020

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Power grid disturbances are caused by various events, including line trips, generator trips, and load disconnections, among others [1]. The timely detection of these events are significant to avoid severe consequences including large-scale blackout, which can cost up to \$10 billion in economic losses [2]. In power grid operations, a series of time series data can be obtained through real-time monitoring and recording of the power grid frequency using phasor measurement units (PMUs). The objectives of deploying the PMUs are to [3]: (i) capture slow spontaneous or anomalous oscillatory swings that are poorly damped; (ii) capture frequency transients from sudden losses of generation or load; (iii) capture power system disturbance data to support analyses of the events; and (iv) develop experience in recognizing disturbances as a precursor to the development of emergent states and unconventional transient state control. One application framework can be depicted as shown in Figure 1. Specifically, PMUs are deployed closer to the transmission or distribution lines. The data collected are transmitted to a central data storage, where methods presented in the paper can be applied. In the future, advanced PMU technology may incorporate edge computing capabilities such that the methods presented in this paper are embedded into the PMU devices for real-time event detection;

thus, eliminating the need for transmitting data to a central location before a detection task can be performed.

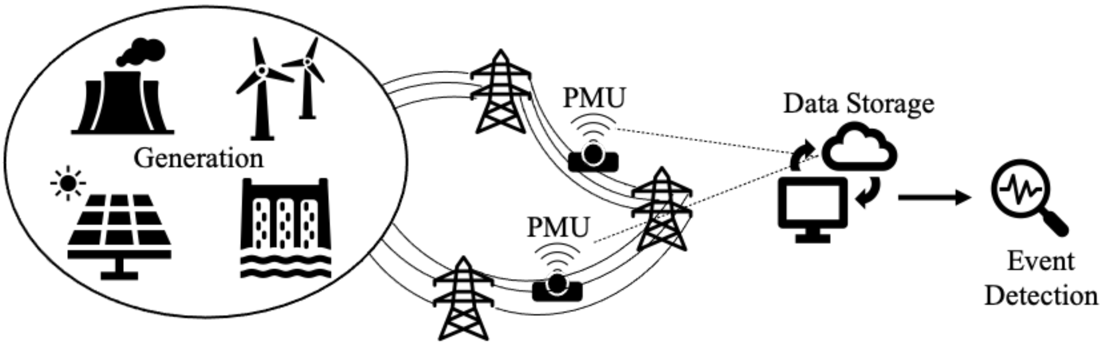


Figure 1. A depiction of how PMUs are used in power systems.

The PMUs collect three types of data—frequency, voltage magnitude, and phase angle. In this paper, we consider only the frequency data. Frequency data is revealing because it provides information about the system changes, namely, generation electromechanical transients, generation demand dynamics, and system operations, such as load shedding, break closing, and capacitor bank switching [4]. By design the power frequency in the United States is 60 Hz (or 50 Hz in other countries). However, the power frequency fluctuates frequently and irregularly throughout the day within an extremely narrow range due to negligible system changes. These variations are due to insignificant perturbations in the system. Consider, for example, the frequency data streams shown in Figure 2. The data is drawn from single-phase PMU that capture response to a generation loss. The resulting system frequency drop is a sharp decline from steady-state frequency of about 60.01 Hz around the time of 17:50:58 to a quasi-steady state frequency of about 59.93 Hz around the time of 17:51:8. The fluctuation in the data before the sharp drop are characterized as normal fluctuations (insignificant perturbations) that should be treated as parts of the steady state region before the drop. The same is true for fluctuations around the quasi-steady state region after the drop in frequency.

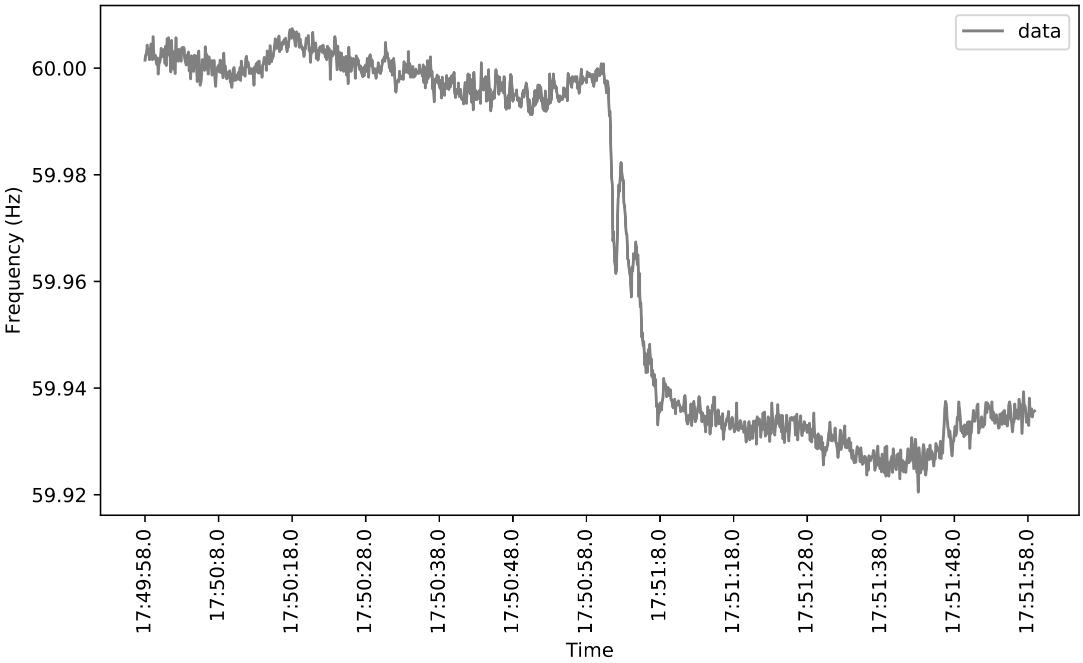


Figure 2. An example of power frequency data streams recorder by a PMU.

As shown in Figure 2, abnormal behaviors in power frequency due to disturbances are directly reflected in the PMUs data. Thus, the increasing deployment of PMUs on the power grid is aiding the understanding of power grid dynamics. Consequently, PMU data have been used for wide area situational awareness [5], disturbance event detection [6], load control [7], line outages [8], and inter-area oscillation analysis [9]. In this paper, we focus on event detection using data-driven approaches. There have been some work, also, done in this area (e.g., [10–12]). However, the proposed framework uses machine committee algorithms to achieve a better detection accuracy. We consider two major detection approaches - event detection and change-point detection.

Event detection in time series refers to finding a point data or a contiguous subsequence in the time series that does not conform to the expected behavior of the system. In power grid, that will mean detecting point data that significantly deviates from the design frequency; therefore, event detection methods will be looking for significant deviations from what constitutes the normal power grid operating frequency. Change-point detection, on the other hand, refers to locating data point in time where there are changes in some aspect of the power frequency distribution. In other words, where the power frequency changes from a somewhat steady state to a somewhat quasi-steady state. We evaluate the proposed approaches using three real-world case studies.

Hence, the main contribution of this paper is four-fold: (1) it presents a machine committee framework for analyzing disturbances in power frequency using PMUs data; (2) it develops a machine committee algorithm that uses five event detection methods to detect anomalous data points in PMUs data; (3) it develops another machine committee algorithm that uses two change-point detection methods to detect phase changes in PMUs data; and (4) it conducts an evaluation of the proposed algorithms using three real-world case studies.

The rest of this paper is organized as follows. Section 2 presents the framework for the machine committee algorithm and discusses the various event detection and change-point detection methods. Section 3 describes the three real-world case studies and presents the results of the evaluation of the proposed algorithms. Section 4 discusses the results and their implications for practical scenarios. Section 5 concludes the paper with a brief summary and discusses plans for future studies.

2. Machine Committee Framework

The proposed Machine Committee framework consists of a group of detection methods, each of which has been widely used in many diverse fields. The framework uses two different machine committee algorithms; one algorithm is based on event detection (ED) methods, while the other algorithm is based on change-point detection (CPD) methods. Specifically, the proposed Event Detection Machine Committee (EDMC) algorithm invokes five basic ED methods to generate detection outputs with different confidence level. The ED methods in the EDMC algorithm perform the same detection task individually and their outputs are combined in a combiner to obtain better event detection performance. On the other hand, the proposed Change-Point Detection Machine Committee (CPDMC) algorithm invokes two basic change-point detection methods to generate detection outputs with different confidence level. The following subsection describes the framework for the EDMC algorithm and the ED methods it uses. There are different ways of combining the individual outputs in the combiner. In this paper, the combiner approach is based on a voting strategy.

2.1. The EDMC Algorithm

Figure 3 shows the framework for the EDMC algorithm. In committee machines, a computational task is solved by using different methods and then combining the detection results of these computations. The idea behind the committee machines is that it generates an aggregated view over a decision of multiple agents which potentially have different weaknesses and advantages. Due to the aggregated vote, these weaknesses are minimized, and the majority vote leads to better results [13].

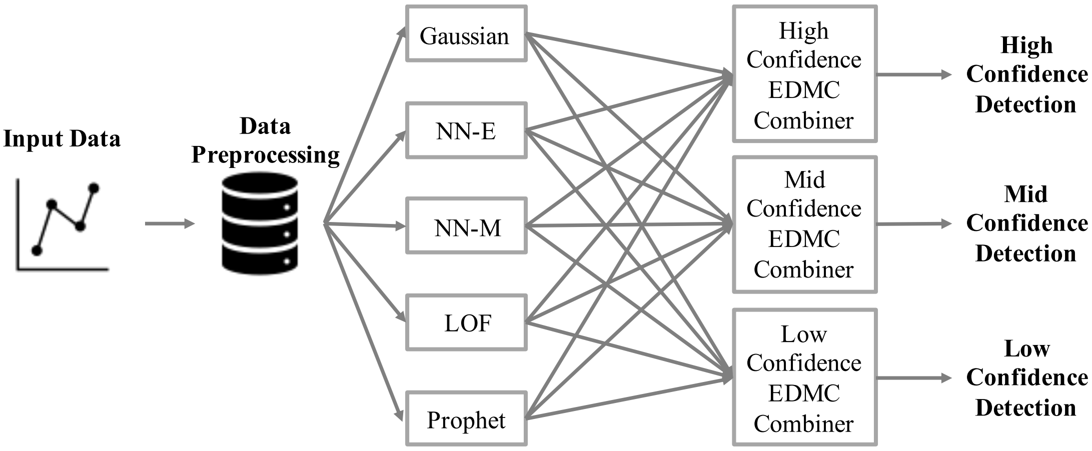

Input Data
Data Preprocessing
Gaussian
NN-E
NN-M
LOF
Prophet
High Confidence EDMC Combiner
Mid Confidence EDMC Combiner
Low Confidence EDMC Combiner
High Confidence Detection
Mid Confidence Detection
Low Confidence Detection

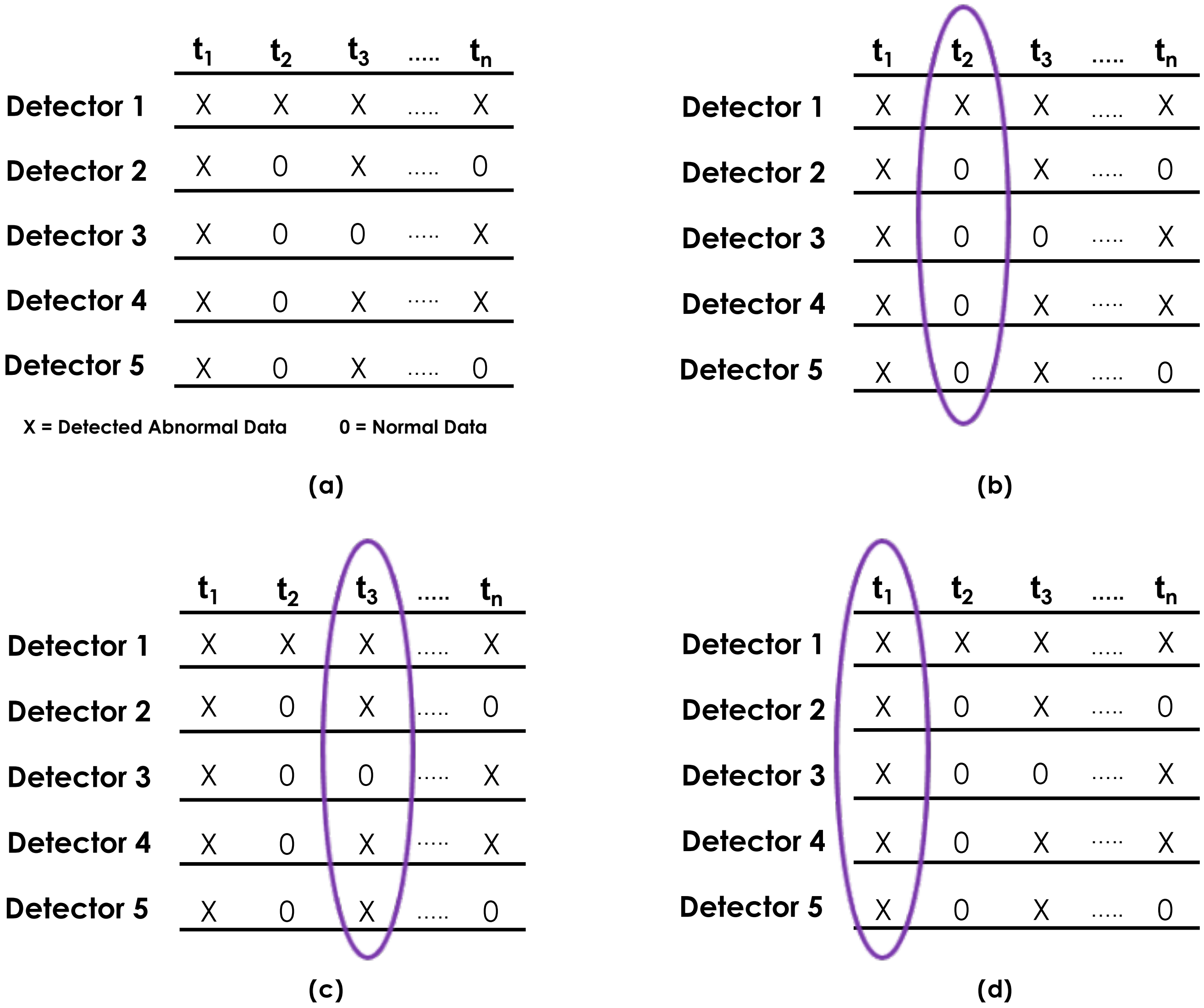
Figure 3. Framework for the EDMC Algorithm.

Formally, input frequency data in given as $F = [f_1, f_2, \ldots, f_n]$, where n is the number of input data over a time window. The event locations in dataset refer to the data points that do not follow the expected data behavior. The input data will be preprocessed and fed into the next stage, where five ED methods are used. Those methods detect events using the preprocessed data individually with three sets of parameters ($[p_0, p_1, p_2]$) and each method produces detection results in three levels: high confidence, mid confidence, and low confidence. By given the parameter p and input data F, prediction results can be given as $P(p, F) = [r_1, r_2, \ldots, r_n]$, where $r = 0$ if the data is normal, or $r = 1$ if the data is an outlier. Thus, the voting results for an agent is calculated as $R = P(p_0, F) + P(p_1, F) + P(p_2, F)$, where $R[i](i \in [1, n])$ have four potential results: normal data (0), low confidence outlier (1), mid confidence outlier (2), and high confidence outlier (3).

Then, the final detection results are voted from the results of the detection methods. Results with the same confidence level from the five methods are aggregated in this stage. For example, the final high confidence results are voted from the high confident outputs of the five ED methods.

Voting Strategy

An illustration of the voting strategies are represented in Figure 4. We consider five detection methods and n time period as shown in Figure 4a. For each time period, the data classified as anomalies are represented with an X, while normal data are represented with a 0. We can then say that Detector 1 classified all the data points as anomalies, while Detectors 2 and 5 classified data points at t_1 and t_3 as anomalies. Furthermore, Detectors 3 classified data points at t_1 and t_n as anomalies, while Detector 4 classified data points at t_1, t_3, and t_n as anomalies. Using a control number represented as C, we can generate different outputs. If the number of the methods that identify the same data as an anomaly is no less than the control number C, the data is voted as an event. As an illustration, if we set C to 1, we can obtain outputs based on the Union voting strategy as shown in Figure 4b; in this case, time t_2 is the selected output. The output for C equals to 4 is shown in Figure 4c; which means that time t_3 is the selected output. The output for C set to 5 is shown in Figure 4d; that is, time t_1 is the final output.



	t_1	t_2	t_3	.....	t_n
Detector 1	X	X	X	.....	X
Detector 2	X	0	X	.....	0
Detector 3	X	0	0	.....	X
Detector 4	X	0	X	.....	X
Detector 5	X	0	X	.....	0

X = Detected Abnormal Data 0 = Normal Data

(a)

	t_1	t_2	t_3	.....	t_n
Detector 1	X	X	X	.....	X
Detector 2	X	0	X	.....	0
Detector 3	X	0	0	.....	X
Detector 4	X	0	X	.....	X
Detector 5	X	0	X	.....	0

(b)

	t_1	t_2	t_3	.....	t_n
Detector 1	X	X	X	.....	X
Detector 2	X	0	X	.....	0
Detector 3	X	0	0	.....	X
Detector 4	X	0	X	.....	X
Detector 5	X	0	X	.....	0

(c)

	t_1	t_2	t_3	.....	t_n
Detector 1	X	X	X	.....	X
Detector 2	X	0	X	.....	0
Detector 3	X	0	0	.....	X
Detector 4	X	0	X	.....	X
Detector 5	X	0	X	.....	0

(d)

Figure 4. Different voting strategies for the EDMC Combiner. (**a**) Sample data; (**b**) results for Union (C set to 1) strategy; (**c**) results for Voting (C set to 4); and (**d**) results for Intersection (C set to 5) strategy.

2.2. Detection Methods

In this section we describe each of the five ED and two CPD methods implemented in this paper.

2.2.1. Gaussian Anomaly Detection Approach

Gaussian distribution is one common approach for anomaly detection. In this method, data are modeled on a Gaussian distribution and the Cumulative Distribution Probabilities (CDP) of each data points are given by Gaussian distribution function, which is given as:

$$p(x) = p(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} \exp\,(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2})dx, \tag{1}$$

where μ is the mean of the distribution and σ is its standard deviation. A set of thresholds are set to determine the outliers. If the probability of a data point is below or above a particular threshold, the data will be detected as an anomaly. Specifically, the probability of the normal data is located in $[threshold, 1 - threshold]$. Some of the advantages of the Gaussian anomaly detection method include easy interpretation, low calculation time and fair performance. However, it is not an all-rounder; the lack of consideration for the temporal order of data could cause potential information loss.

2.2.2. Nearest Neighbor Approach

The Nearest Neighbor (NN) approach, which is based on a similarity measure, calculates the distance of the k-th nearest neighbor from the data point. The distance depicts the sparseness of neighborhoods of a data. For example, data points with larger nearest neighbor distance typically represent more sparse neighborhoods and more likely they are outliers. We choose three different numbers of nearest neighbors as the parameters of the approach. Then, calculating the mean value of the distances of the data point with the neighbors for each data point. A threshold serves to determine whether a data point is an anomaly or not.

In our method, we use both the Euclidean Distance based NN (NN-E) approach and Mahalanobis Distance based NN (NN-M) approach. The Euclidean distance of points f_1, f_2 in one dimension space, which is the length between the two points, is given as:

$$D^e(f_1, f_2) = \sqrt{(f_1 - f_2)^2}, \tag{2}$$

The Mahalanobis distance is given as:

$$D^m(f) = \sqrt{(f - \mu)^T \cdot \Sigma^{-1} \cdot (f - \mu)}, \tag{3}$$

where μ is the mean of the neighbors value, and Σ is the covariance matrix of the data. The standard Euclidean distance matrix is easy to compute and interpret, but is not always beneficial for distance calculation. The Mahalanobis distance, which takes the correlation of the data into account, may have better performance in some scenarios [14].

2.2.3. Local Outlier Factor Approach

The local outlier factor (LOF), which is based on the local density, represents the degree of being outlier in this approach. By comparing the local density of a data point to the local densities of its neighbors, points with lower density than their neighbors will be claimed as outliers. The local density is also calculated by the distance matrix. Similar to the NN method, three different numbers of nearest neighbors are set as the parameters of the approach. The advantage of this method is that it can capture the outliers that have short distance with their neighbors but have lower local density comparing with that of the neighbors.

2.2.4. Prophet Approach

While the above methods have their advantages, they all didn't consider the time series factor, which may contains periodic changes and trends. Prophet is an efficient time series forecasting tool developed by Facebook's data science team [15].

Prophet uses a decomposed time series model which contains three model components: trend, seasonality, and holidays [16]. They are given as:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t, \tag{4}$$

where $g(t)$ is trend function for non-periodic changes in the time series, $s(t)$ models weekly and yearly periodic changes in the data, $h(t)$ is the holiday function which models the irregular changes in the data, and ϵ_t is a normal distributed error function representing the changes that can't be modeled by previous functions.

2.3. The CPDMC Algorithm

Change-points are characterize as abrupt variations in time series data [17]. Such abrupt changes may represent transitions from one state to another; in power grid frequency data, abrupt changes will represent transition from steady state to quasi-steady state as shown in Figure 2. CPD is the task of finding where those abrupt changes occur in time series data. CPD algorithms are usually classified as offline or online CPD. The framework for the CPDMC algorithm is similar to that of the EDMC algorithm by replacing the five ED methods with two CPD methods - offline CPD and online CPD methods.

2.3.1. Offline CPD Approach

Offline CPD method considers the entire data set at once, and most appropriate for batch implementation. Thus, the offline CPD-based algorithms look back in time to determine where changes have occurred. Various offline CPD algorithms were developed for different domains [18]. In this paper, we implement the efficient Bayesian offline CPD of which details can be found in [19].

Basically, we assume the data can be partitioned into a number of K segments. The marginal likelihood produced by a single model m for data from time s to t is given as:

$$p(f_{s:t}|m) = \int [\prod_{i=s}^{t} p(f_i|y_{1:i-1}, \theta, m)] p(\theta|m) p(m) d\theta. \tag{5}$$

If the segment include data that are generated from different model types or parameters as it grows, the marginal likelihood will drop which suggests that a change-point and two models should be applied [6].

2.3.2. Online CPD Approach

Online CPD approach, on the other hand, processes data in real-time; that is, as each data point becomes available. The goal is to detect a change point as soon as possible after it occurs, ideally before the next data point arrives [17].

Adams and Mackay present a Bayesian CPD for online inference in their work [20]. By generating an accurate prediction of the next data in the sequence, they used a causal predictive filtering rather than segmentation methods of offline CPD. Intuitively, the predictive probability of next unseen data based on the existing data is calculated. If the next data has a large margin with the prediction, it will be claimed as a change-point data. The predictive probability is calculated by the marginal predictive distribution, which is given as:

$$p(f_{t+1}|f1 : t) = \sum_{r_t} p(f_{t+1}|r_t, f_t^{(r)}) p(r_t|f_{1:t}), \tag{6}$$

where r_t is the given run length, which is the time steps since the last change-point data.

3. Evaluation

In this section, an evaluation of the proposed EDMC and CPDMC algorithms are presented using three real-world case studies. We start with the enumeration of the parameters used for the experiments; then, description of the synchrophasor data used for the evaluation; and presentation of the results of the evaluation.

3.1. Parameters for the Methods

Table 1 shows the parameters used for each methods in the experiments. For each method, three sets of parameters were selected. For the EDMC Combiner, C is set to 2 for the voting strategy; while for the CPDMC Combiner, C is set to 1 for the final CP probabilities.

The threshold for the Gaussian approach is user defined and it can be regarded as the user p-value. In this paper, we chose [0.01, 0.05, 0.1] as the three thresholds. Assuming that the data follow normal distribution, then the CDP of outliers is less than 0.1 or larger than 0.9. Consequently, the CDP for high confidence outliers is less than 0.01 and larger than 0.99; the CDP for mid confidence outliers is in [0.01, 0.05] or in [0.95, 0.99] range; and the CDP for low confidence outliers is in [0.05, 0.1] or [0.90, 0.95] range.

For the NN method, the detection thresholds are 0.1, 0.2, 0.3. In this case, the threshold is the percentage of the number of the nearest neighbors used in the method. Through a trial and error method, we found that when the threshold is more than 0.3, the results have fewer differences. By choosing the threshold as 0.1, 0.2 and 0.3, the method shows different performances, which is suitable for using a voting system.

The reason for choosing the parameter used for LOF is very similar to that of NN. In order to obtain different performances of the LOF method, the parameter used are 0.3, 0.5, and 0.7 based on empirical experiments.

For the Prophet method, the internal width is set to 0.99 so that the full boundary of method could be used. Furthermore, we used different range of historical data for estimating the trend in the data. Specifically, 10%, 30%, and 50% of the data were used to obtain different performances.

The parameters for the CPD approaches are based on heuristic. We observed that the probability for the normal data is no more than 0.1 and the probability for high confidence CP is always larger than 0.5. Then we used 0.25 to distinguish between the low confidence and mid-confidence results. This subsection describes these CPD methods.

Table 1. Parameters for methods used in experiments.

Detection Methods	Parameters	Value	Description
Gaussian	*thresholds*	0.01, 0.05, 0.1	The probability of normal data locates in $[thresholds, 1 - thresholds]$.
NN_E	*n_neighbors*	0.1, 0.2, 0.3	*n_neighbors* is the percentage of the number of the nearest neighbors used in the methods.
NN_M	*n_neighbors*	0.1, 0.2, 0.3	
LOF	*n_neighbors*	0.3, 0.5, 0.7	
Prophet	*internal_width* *anomaly_range*	0.99 0.1, 0.3, 0.5	*internal_width* is the width of the uncertainty intervals provided for the forecast. *anomaly_range* is the proportion of history in which trend anomaly will be estimated.
CPD	*thresholds*	0.1, 0.25, 0.5	The three thresholds split data into non-CP, low-confidence CP, mid-confidence CP and high-confidence CP.

3.2. Synchrophasor Data

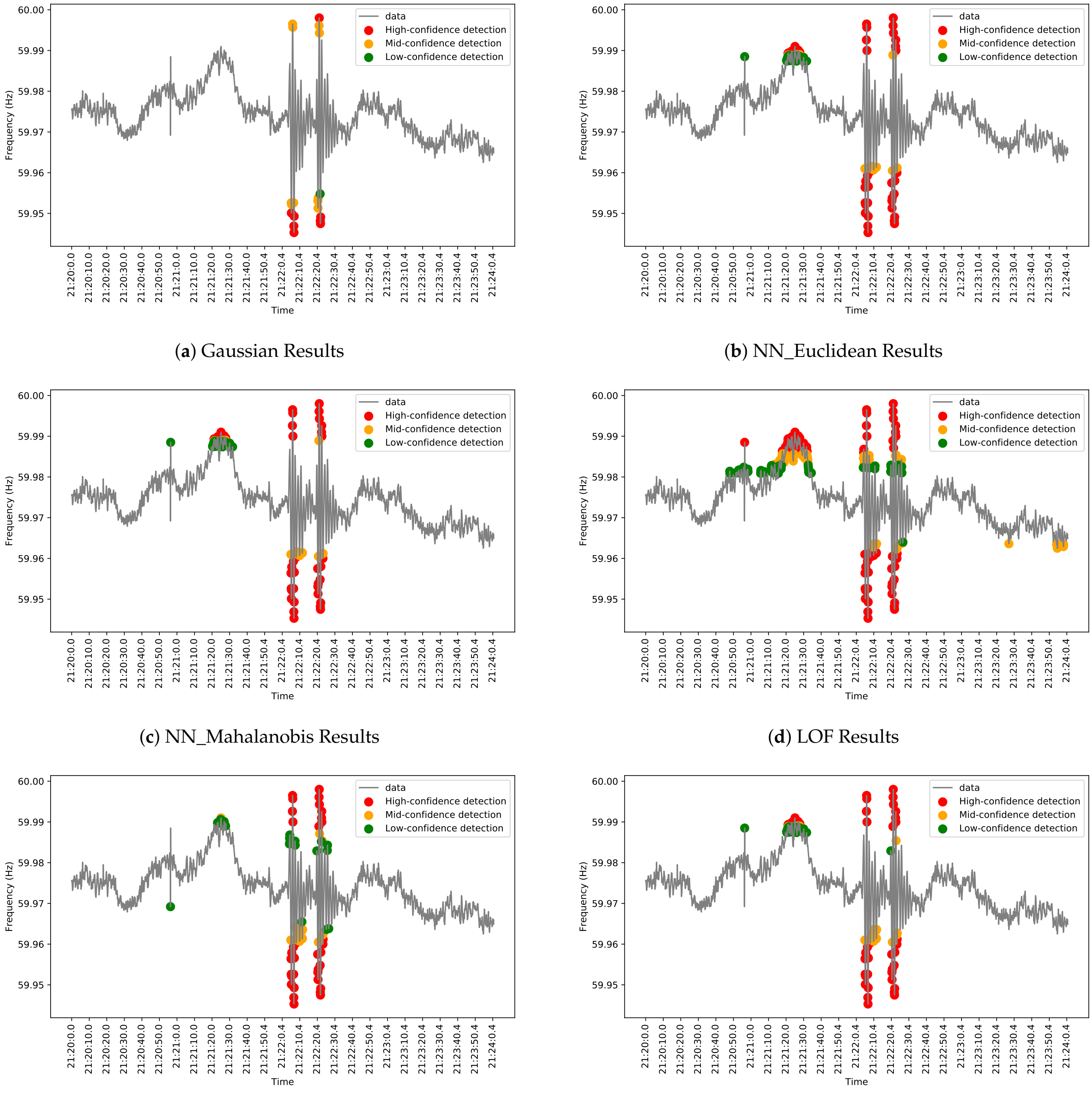
The single-phase synchrophasor data, which contain time stamp and frequency value (10 or more measurements per second), is collected by thousands of PMUs that are deployed on the power grid in the USA [9]. The high volume, velocity, and variety of PMU measurement data make it possible to take advantage of artificial intelligence techniques in applications such as short-time events and faults detection.

For the evaluation of the proposed framework, we consider the following three case studies of real-world disturbances to the power grid.

3.2.1. Case Study 1

In this case study, an event occurred during a large severe storm system on the Eastern Interconnection in the USA on 4 April 2011 (Case study 1 Youtube animation video: https://www.youtube.com/watch?v=KmK2VMG57gw). We used the data collected from the PMU deployed at the Florida State University for this evaluation.

Figure 5 presents the data and results of the EDMC methods and algorithm. The results are grouped into three classes: high-confidence events (depicted using red color), mid-confidence events (orange color), and low-confidence events (green color). Figure 5a–e show the results of the individual event detection methods; while, Figure 5f shows the EDMC results; while Figure 6 shows the CPDMC results.



(**a**) Gaussian Results

(**b**) NN_Euclidean Results

(**c**) NN_Mahalanobis Results

(**d**) LOF Results

(**e**) Prophet Results

(**f**) EDMC Results

Figure 5. EDMC-related results for study case 1.

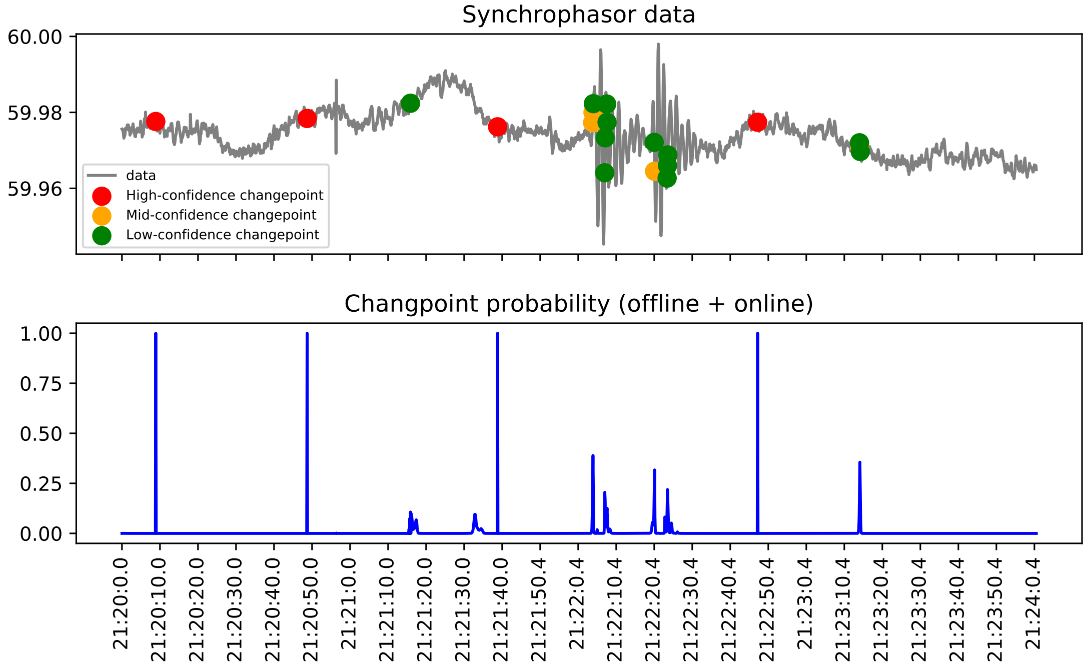
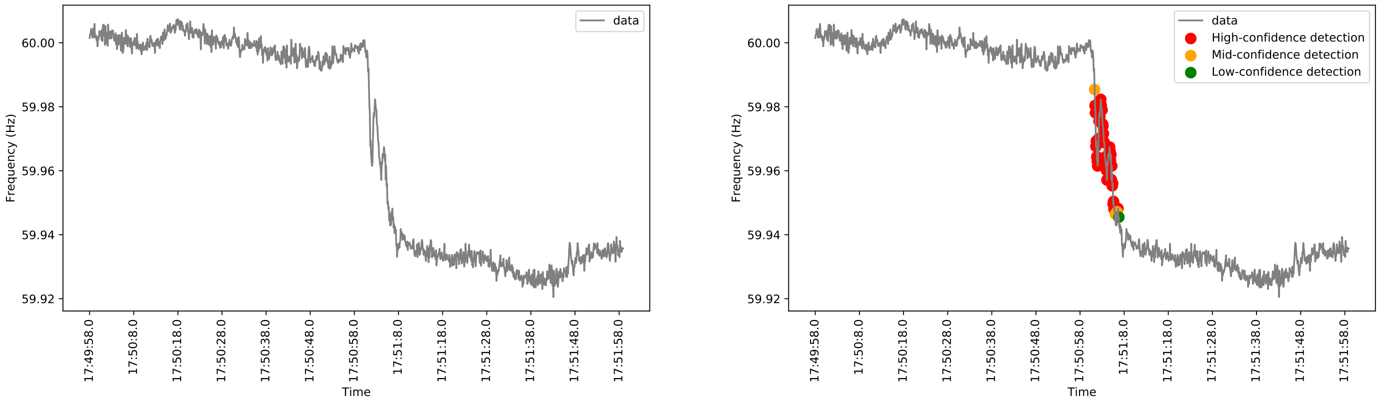


Figure 6. CPDMC results for study case 1.

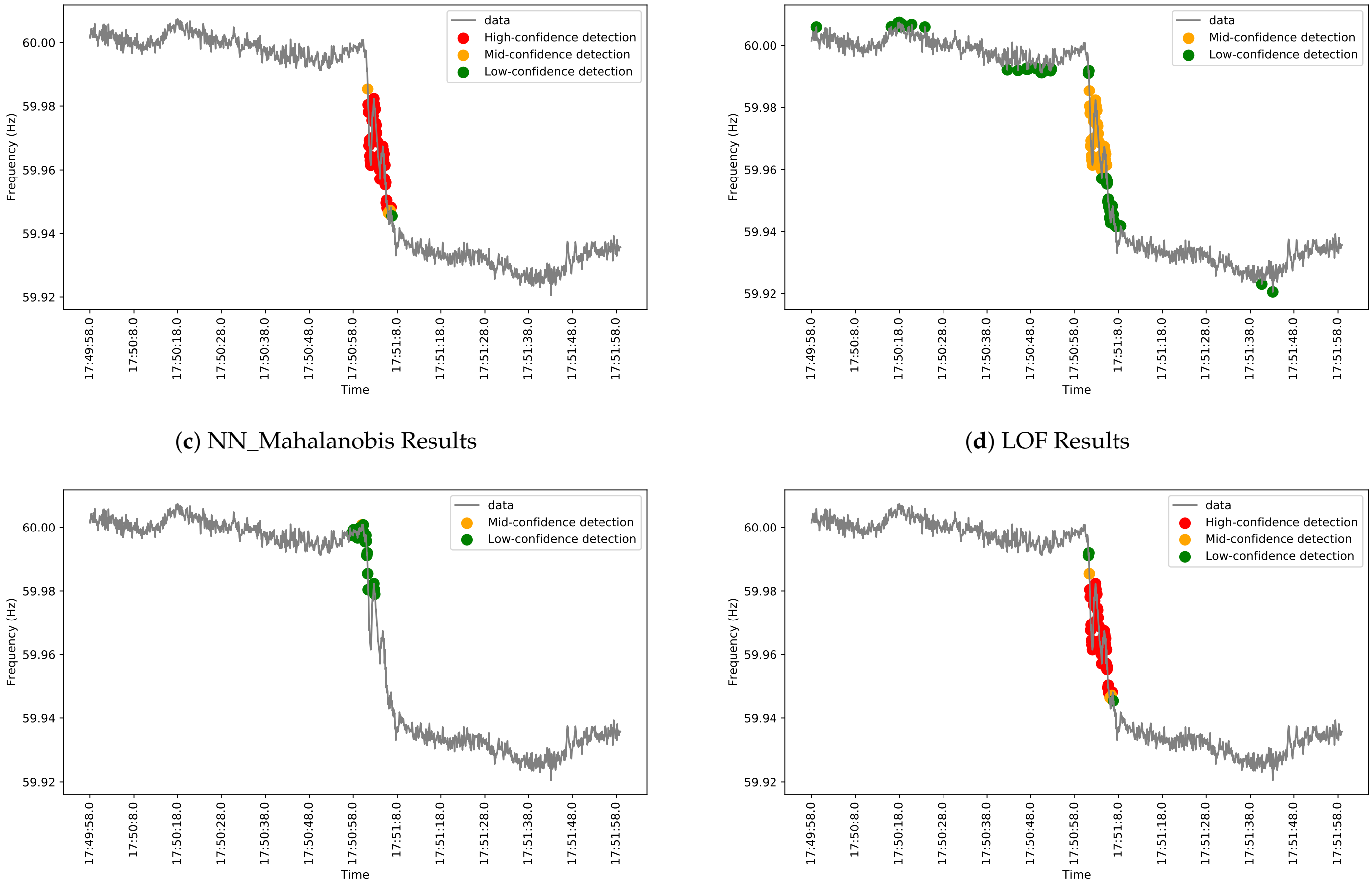
3.2.2. Case Study 2

In the second case study, an event occurred when a 1600 MW generator trip caused by the East Coast Earthquake on 23 August 2011 (Case study 2 Youtube animation video: https://www.youtube.com/watch?v=XUN_h-k8kBg). The data is collected from the PMU deployed at Atlantic City, New Jersey. Figure 7 shows the results for EDMC algorithm using this data. From the figure, we can see there is a frequency drop starting around 17:50:58 because of the generator trip. Figure 8 shows the CPDMC results.



(**a**) Gaussian Results (**b**) NN_Euclidean Results

Figure 7. *Cont.*



(**c**) NN_Mahalanobis Results

(**d**) LOF Results

(**e**) Prophet Results

(**f**) EDMC Results

Figure 7. EDMC-related results for study case 2.

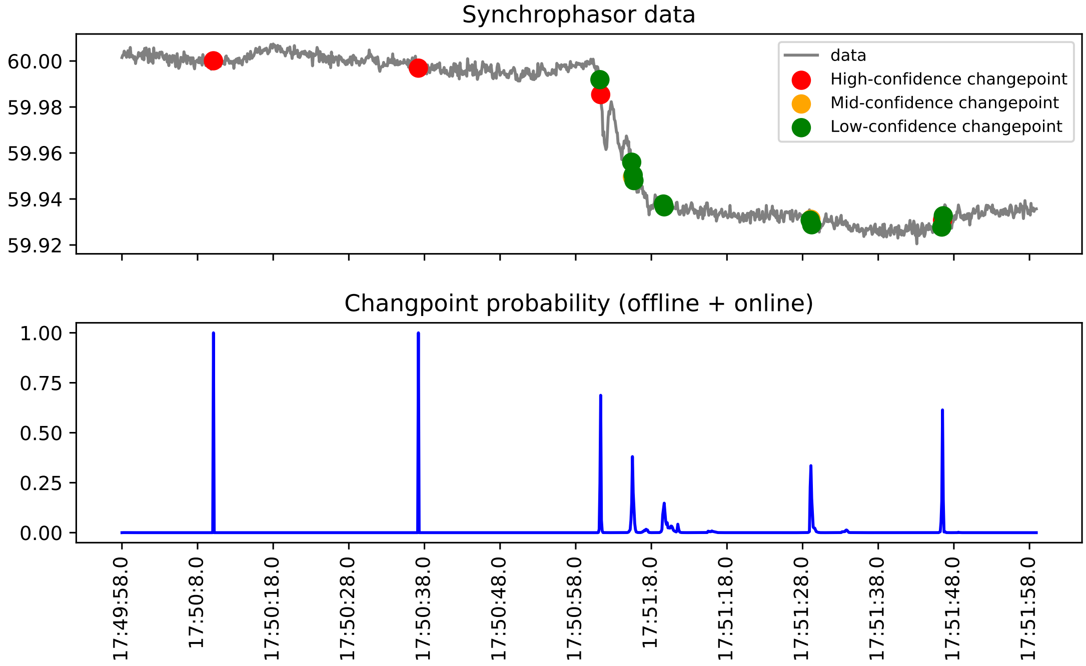
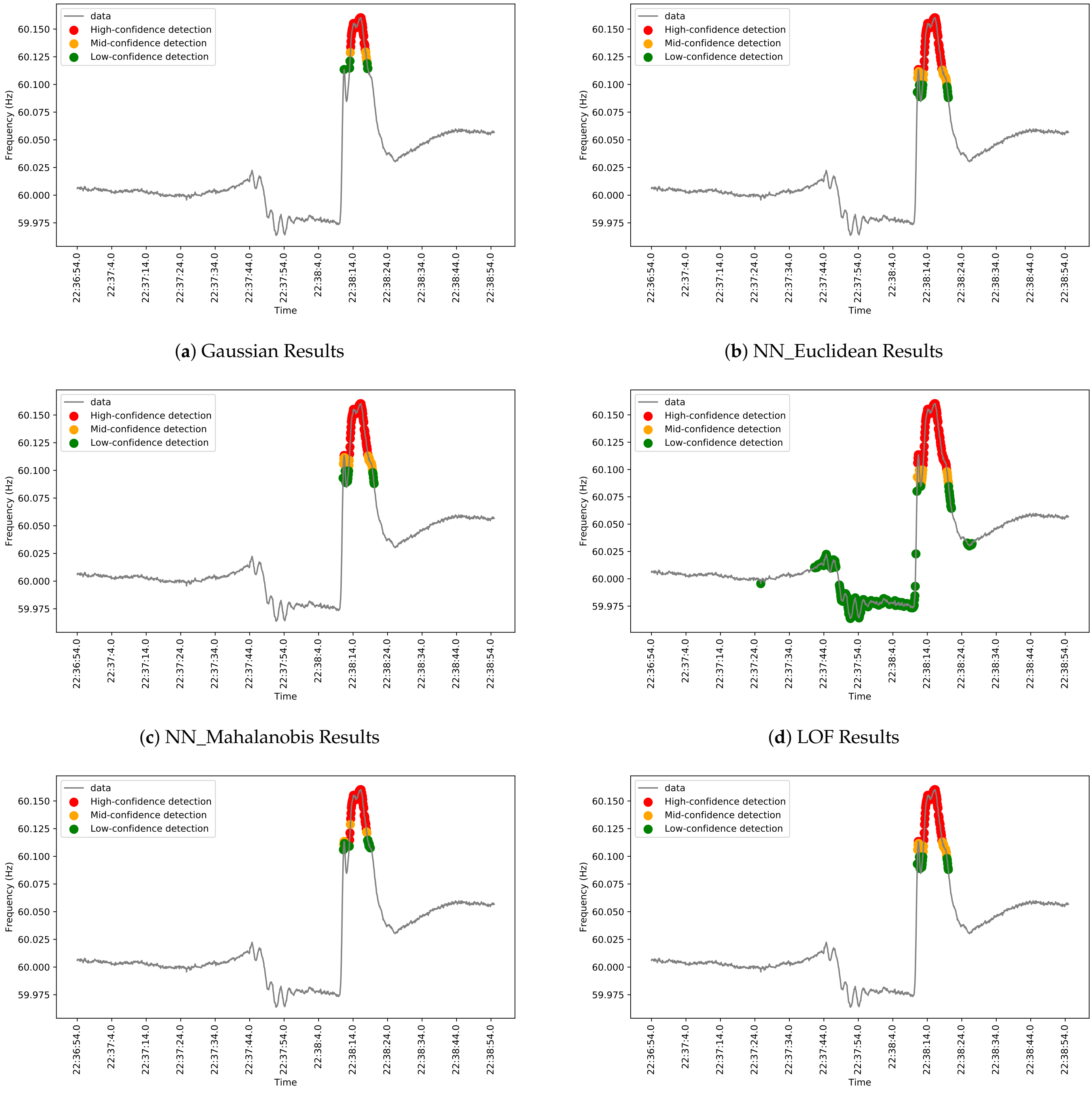


Figure 8. CPDMC results for study case 2.

3.2.3. Case Study 3

The third case study shows an event that occurred when a 500-kV line connecting Arizona with San Diego tripped following a capacitor switch-out (Case study 3 Youtube animation video: https://www.youtube.com/watch?v=YsksUyeLu2Y). Approximately 1.4 million people were affected. The PMU that deployed at Arizona State University captured the event, which generated a peak frequency between the time 22:38:10 and 22:38:24 and the results for the EDMC algorithm are shown in Figure 9; while, the results of the CPDMC results are shown in Figure 10.



(**a**) Gaussian Results

(**b**) NN_Euclidean Results

(**c**) NN_Mahalanobis Results

(**d**) LOF Results

(**e**) Prophet Results

(**f**) EDMC Results

Figure 9. EDMC-related results for study case 3.

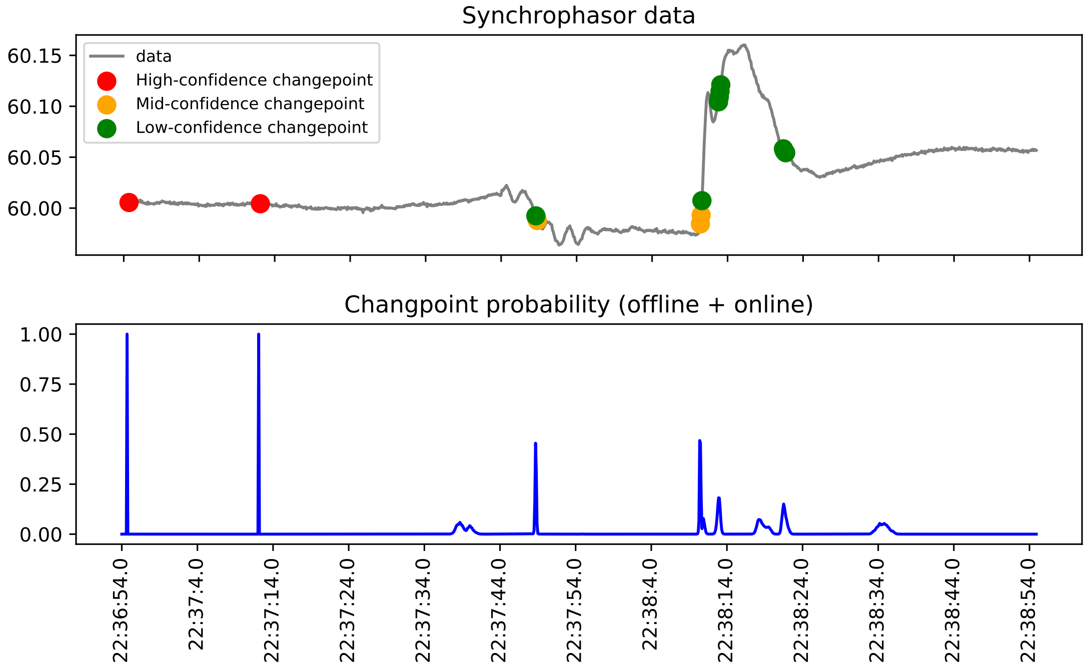


Figure 10. CPDMC results for study case 3.

3.3. Performance

We use True Positive Rate (TPR) and False Positive Rate (FPR) for performance comparison for EDMC and individual approaches of EDMC. The TPR and FPR are given as:

$$TPR = TP/(TP + FN), \tag{7}$$

$$FPR = FP/(FP + TN), \tag{8}$$

where TP, FN, FP, TN represent the number of true positive, false negative, false positive, and true negative respectively. The TPRs and FPRs of each approach for the three case studies are presented in Table 2. The best value in each case is identified in red color; while, the worst value is identified in blue color. In Case Study 1, LOF has the best TPR; but it also has the worst FPR. Guassian, NN_E, and NN_M have zero FPR but their TPRs are low. EDMC has a promising TPR with a better FPR. In Case Study 2, EDMC performed the best for both TPR and FPR. In Case Study 3, LOF has the best TPR and the worst FPR which is similar to Case Study 1 results. The FPR of the other basic approaches are not good compared with EDMC. Overall, the performance of EDMC is a good trade off between TPR and FPR.

Table 2. TPR and FPR in Case Studies.

Case Study	Criteria	Gaussian	NN_E	NN_F	LOF	Propeht	EDMC
Case Study 1	TPR	15.22%	52.90%	52.90%	92.75%	52.89%	86.23%
	FPR	0	0	0	8.86%	0.22%	0.13%
Case Study 2	TPR	0	86.44%	86.44%	83.05%	32.21%	89.83%
	FPR	0	0	0	2.35%	1.74%	0
Case Study 3	TPR	63.33%	69.17%	69.17%	91.67%	50.83%	72.50%
	FPR	0	0	0	25.96%	0	0

4. Discussions

The early detection of abnormal patterns in frequency data that may be indicative of disruptive disturbance could help to prevent large-scale outages or limit their impact. In this section, we discuss the results and their implications for large-scale outages.

4.1. Case Study 1

Looking at the data for case study 1, we notice several smaller fluctuations before and after the major fluctuations between 21:22:0.4 and 21:22:30.4. Those smaller fluctuations probably suggest the emergence of the impending major fluctuations. The major characteristics of this event is the significant fluctuations in the frequency signal around the event time, which may suggest the presence of oscillation in the system. However, from the results in Figure 5, we notice that the Gaussian method failed to detect any of these smaller fluctuations; thus, unable to provide any early warnings before the major fluctuations. Furthermore, the other methods detected the smaller fluctuations to a different degree. For example, the LOF method seems to detect a lot more data points than the other methods; the Prophet method, on the other hand, detects only the peak data points. The NN_Euclidean and NN_Mahalanobis methods have similar results but identified more data as outliers. All the results shown in Figure 5a–e suggest that the use of a single method may be providing limiting information. The results shown in Figure 5f minimize the weaknesses of these methods and maximize their strengths. Overall, the EDMC algorithm detects extremely higher and lower data points with respect to the other data points in the data. Unlike in the results for the LOF method, most of the detected data points are of mid to high confidence.

In addition to the ED methods, the results of the CPD methods are shown in Figure 6. Looking at these results, we can see that the CPD methods are not a good fit for analyzing this data set. The CPD methods detected change-point around the event region with low- to mid-confidence, and detected change points due to regular fluctuations in the data with high-confidence.

4.2. Case Study 2

For this case study, we would like to see the methods detect data points between 17:50:58.0 and 17:51:8.0. The major recognizable characteristics of this event is the huge drop in frequency, which is generally attributed to a major drop in load. Interestingly, the Gaussian method, of which results are shown in Figure 7a, had the worst performance and didn't detected any event. However, NN_Euclidean and NN_Mahalanobis detected the most data in the process of generator trip with high-confidence, which is shown in Figure 7b,c. The LOF method, on the other hand, detect those same data points with mid-confidence; but, it also detected several several other data points with low-confidence. The Prophet method detected data points around the start of the drop in frequency but with low-confidence, but it didn't detected any data points with high-confidence and mid-confidence. Again, we can see the variability in the results of the different methods. The EDMC results, which are shown in Figure 7f, seem to combine the best of all these previous results. It detected the start of the frequency drop with low-confidence; then, detected the data point after that with mid-confidence; and the data points after that with high-confidence. The results suggest a logical progression in the confidence of the detection. Furthermore, at the end of the frequency drop, the level of confidence changed back to mid- and low-confidence, which suggest the start of the quasi-steady state. Again, the use of a machine committee approach eliminates false alarms as seen using the LOF method, late detection as seen using the NN-related methods, and/or limited detection as seen using the Prophet method.

When we compare these results to the results of the CPD methods shown in Figure 8, we can see that the onset of the frequency drop is detected with high-confidence, which is similar to the EDMC results. However, there are some false change-point locations with higher probability at the start of the data series. Therefore, we can say that the ED methods are a better fit for this case study.

4.3. Case Study 3

In case study 3, the objective is to detect higher power grid frequency between 22:38:14.0 and 22:38:24.0. For this event, we noticed an increase in frequency that could suggest an instability in the system; thus, a characteristic feature of a blackout. In Figure 9a–e, we show the results of each of the ED methods. The Prophet method detected the least number of data points, but most of them are with high-confidence. The NN_Euclidean, NN_Mahalanobis, and Gaussian methods have similar results while the LOF method detected more data points are possible events; however, most of them are with low-confidence. In actual sense, all the data points detected with low-confidence are false alarms. Overall, the LOF method seems to generate some false alarms for each case study in this paper. The results for the EDMC method shown in Figure 9f are similar to that of NN_Euclidean and NN_Mahalanobis methods. In these three methods, the early detected data points are detected with low-confidence, the next round of data points are detected with mid-confidence, and the higher value data points are detected with high-confidence. These three methods have no false alarms.

In comparison to the CPD methods shown in Figure 10, the ED methods are, again, a better fit for event detection in this case study. For the CPD methods, the probability of a change point around the higher frequency value is less than 0.5; hence, detected with low confidence. In addition, data point locations detected as change-point with high probability of more than 0.5 can be regarded as false alarms.

The EDMC algorithm, which is an ensemble-based event detection algorithm for synchrophasors data, combines five different event detection methods and automatically combines their outputs using the voting strategy. As shown in our three case studies, the EDMC algorithm is more stable and less sensitive to the change of data pattern, and achieve comprehensive and reasonable results. The results show that the EDMC algorithm performs better than each of the five detection methods separately by detecting irregular frequency patterns in synchrophasors data streams. In addition, the EDMC algorithm performs better than the CPDMC algorithm for power grid disturbance analysis.

5. Conclusions

This paper proposes a machine committee framework for power grid disturbance analysis using synchrophasors data. The framework consists of two algorithms—EDMC and CPDMC. Each algorithm is an ensemble-based algorithm that combines different detection methods and automatically combines their outputs using the voting strategy. The EDMC algorithm combines five ED methods; while, the CPDMC algorithm combines two CPD methods. The algorithms were tested using three real-world data sets. From the results of the evaluation, we can conclude that the EDMC algorithm is a better fit for analyzing power grid disturbances recorded by synchrophasors. The CPDMC algorithm generated a lot of false alarms and the probability of detection is very low for event regions.

Our conclusion is limited to the three event cases evaluated in this paper. In the future, additional studies will include more cases with diverse disturbance events. In addition, future studies will include longer time series to understand the effect of the length of the time series on the performance of the algorithms.

Author Contributions: H.N. co-developed the algorithms, implemented the algorithms, and contributed to the manuscript. O.A.O. designed the study, co-developed the algorithms, and contributed to the manuscript. Q.C.C. supervised the algorithms implementation and contributed to the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

Acknowledgments: This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, W.; Yin, H.; Chen, C.; Till, A.; Yao, W.; Deng, X.; Liu, Y. Frequency Disturbance Event Detection Based on Synchrophasors and Deep Learning. *IEEE Trans. Smart Grid* **2020**, *11*, 3593–3605. [CrossRef]
2. *The Economic Impacts of the August 2003 Blackout*; Electricity Consumers Resource Council: Washington, DC, USA, 2004.
3. Jia, Y.; Gardner, R.M.; Xia, T.; Liu, Y. Synchronized Phasor Measurement in Smart Grid Situational Awareness. In *Smart Power Grids 2011*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 565–591.
4. Zhang, Y.; Markham, P.; Xia, T.; Chen, L.; Ye, Y.; Wu, Z.; Yuan, Z.; Wang, L.; Bank, J.; Burgett, J.; et al. Wide-area frequency monitoring network (FNET) architecture and applications. *IEEE Trans. Smart Grid* **2010**, *1*, 159–167. [CrossRef]
5. Bank, J.N.; Omitaomu, O.A.; Fernandez, S.J.; Liu, Y. Visualization and classification of power system frequency data streams. In Proceedings of the 2009 IEEE International Conference on Data Mining Workshops, Miami, FL, USA, 6 December 2009; pp. 650–655.
6. Xia, T.; Zhang, H.; Gardner, R.; Bank, J.; Dong, J.; Zuo, J.; Liu, Y.; Beard, L.; Hirsch, P.; Zhang, G.; et al. Wide-area frequency based event location estimation. In Proceedings of the 2007 IEEE Power Engineering Society General Meeting, Tampa, FL, USA, 24–28 June 2007; pp. 1–7.
7. Yao, W.; You, S.; Wang, W.; Deng, X.; Li, Y.; Zhan, L.; Liu, Y. A fast load control system based on mobile distribution-level phasor measurement unit. *IEEE Trans. Smart Grid* **2019**, *11*, 895–904. [CrossRef]
8. Tate, J.E.; Overbye, T.J. Line outage detection using phasor angle measurements. *IEEE Trans. Power Syst.* **2008**, *23*, 1644–1652. [CrossRef]
9. Bank, J.N.; Omitaomu, O.A.; Fernandez, S.J.; Liu, Y. Extraction and visualization of power system interarea oscillatory modes. In Proceedings of the IEEE PES General Meeting, Providence, RI, USA, 25–29 July 2010; pp. 1–7.
10. Xie, L.; Chen, Y.; Kumar, P. Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis. *IEEE Trans. Power Syst.* **2014**, *29*, 2784–2794. [CrossRef]
11. Liu, Y.; You, S.; Yao, W.; Cui, Y.; Wu, L.; Zhou, D.; Zhao, J.; Liu, H.; Liu, Y. A distribution level wide area monitoring system for the electric power grid–FNET/GridEye. *IEEE Access* **2017**, *5*, 2329–2338. [CrossRef]
12. Song, Y.; Wang, W.; Zhang, Z.; Qi, H.; Liu, Y. Multiple event detection and recognition for large-scale power systems through cluster-based sparse coding. *IEEE Trans. Power Syst.* **2017**, *32*, 4199–4210. [CrossRef]
13. Nagy, Z. *Artificial Intelligence and Machine Learning Fundamentals: Develop Real-World Applications Powered by the Latest AI Advances*; Packt Publishing Ltd.: Birmingham, UK, 2018.
14. Dokas, P.; Ertoz, L.; Kumar, V.; Lazarevic, A.; Srivastava, J.; Tan, P.N. Data mining for network intrusion detection. In Proceedings of the NSF Workshop on Next Generation Data Mining, Baltimore, MD, USA, 1–3 November 2002; pp. 21–30.
15. Taylor, S.J.; Letham, B. Forecasting at scale. *Am. Stat.* **2018**, *72*, 37–45. [CrossRef]
16. Harvey, A.C.; Peters, S. Estimation procedures for structural time series models. *J. Forecast.* **1990**, *9*, 89–108. [CrossRef]
17. Aminikhanghahi, S.; Cook, D.J. A survey of methods for time series change point detection. *Knowl. Inf. Syst.* **2017**, *51*, 339–367. [CrossRef] [PubMed]
18. Truong, C.; Oudre, L.; Vayatis, N. Selective review of offline change point detection methods. *Signal Process.* **2020**, *167*, 107299. [CrossRef]
19. Fearnhead, P. Exact and efficient Bayesian inference for multiple changepoint problems. *Stat. Comput.* **2006**, *16*, 203–213. [CrossRef]
20. Adams, R.P.; MacKay, D.J. Bayesian online changepoint detection. *arXiv* **2007**, arXiv:0710.3742.