Austrian Journal of Statistics
July 2023, Volume 52, 143–162.
http://www.ajs.or.at/
doi:10.17713/ajs.v52i4.1511

Detection and Localization of Faults in a Regional Power Grid

Piotr Kokoszka
Colorado State University

Mantautas Rimkus
Colorado State University

Sanjay Sheshagiri Hosur
University of Wyoming

Dongliang Duan
University of Wyoming

Haonan Wang
Colorado State University

Abstract

The structure of power flows in transmission grids is evolving and is likely to change significantly in the coming years due to the rapid growth of renewable energy generation that introduces randomness and bidirectional power flows. Another transformative aspect is the increasing penetration of various smart-meter technologies. Inexpensive measurement devices can be placed at practically any component of the grid. Using model data, we propose a two-stage procedure for detecting a fault in a regional power grid. In the first stage, a fault is detected in real time. In the second stage, the faulted line is identified with a negligible delay. The approach uses only the voltage modulus measured at buses (nodes of the grid) as the input. Our method does not require prior knowledge of the fault type. The method is fully implemented in R. Pseudo code and complete mathematical formulas are provided.

Keywords: fault detection, fault localization, data-driven approaches.

1. Introduction

Faults in power systems cause excessive currents and can pose safety threats to personnel and property, and even cause major disasters, like widespread fires, as well as disruptions of economic and social activities. Detection of power grid faults is therefore of paramount importance. As the penetration of renewable energy sources, with pronounced stochastic components, increases, traditional fault detection methods can become insufficient. The objective of this work is to propose a statistical methodology for detecting a fault in a regional power grid, with almost no delay, and locating the faulted line. The faulted line is also determined almost immediately. The algorithm we propose takes as inputs the moduli of voltage measured at high frequency at grid buses. Such data are becoming increasing available due to the growing deployment of phasor measurement units (PMUs) able to communicate power transmission measurements in real time and from practically any location where transmission lines connect, start or terminate (generators, transformers, forks, loads etc.). Such nodes of the grid are referred to as buses. The algorithm is developed on a training dataset and evaluated on a test data set that remains inaccessible to us until the algorithm development is complete. This widely adopted approach is designed to ensure that new methodology generalizes well. The novelty of our approach relative to previous research is discussed in greater detail in Section 4. We merely note here that we are not aware of any study that uses a large database of faults and statistically evaluates performance of a fault detection and localization methodology in terms of success rates.

The statistical methodology used in this paper falls into the general field of change point detection and anomaly localization. The field of statistical change point detection is now well-established, and there are consequently many monographs on change point analysis, e.g. Brodsky and Darkhovsky (1993); Csörgő and Horváth (1997); Gustafsson (2000); Chen and Gupta (2011); Basseville, Nikifirov, and Tartakovsky (2012); Brodsky (2017). The methods we consider are known as *sequential*- or *online detection* or *monitoring*. Their objective is to raise an alarm if there is some departure from the desired state. They focus on minimizing the *Expected Detection Delay* (EDD) and maximizing the *Average Run Length* (ARL), i.e., the expected time until a false detection. The last two decades have seen important advances in theoretical understanding of methods for sequential detection of change-points which may occur in one of many channels or sensors. Research in Dragalin, Tartakovsky, and Veeravalli (1999) is concerned risk optimality theory for multichannel log-likelihood ratio tests. There are M channels, and a penalty is imposed if a change in density is detected in an incorrect channel; stopping occurs as soon as a fault is detected in one of the channels. The work in Mei (2005) incorporates a decision center into the optimality considerations, while that in Raghavan and Veeravalli (2010) studies a situation where the change propagates across sensors. Reference Xie and Siegmund (2013) is concerned with mathematical properties of sequential change point detection in an idealized multi-sensor network with a change occurring in a subset of networks. Temporal and spatial independence and normality of observations are assumed. A somewhat similar approach is taken by Zhang, Siegmund, Ji, and Li (2010) who assume independence across i and t and Gaussian observations, and propose a method of dealing with data that have a large cross-sectional dimension p, tens of thousands, and a huge temporal size T, hundred of thousands. They study gene expression data. In our setting, the value of p will be much smaller, and T will also have to be smaller to use a reasonable moving window. There is also very extensive research on detecting anomalies in various types networks networks. To illustrate different flavors of such research, we list, as subjectively selected examples, Huang, Nguyen, Garofalakis, Jordan, Joseph, and Taft (2007), Lévy-Leduc and Roueff (2009), Paschalidis and Smaragdakis (2009), Bartos, Rehak, and Krmicek (2011), Xie (2012), Rassam, Zainala, and Maarof (2013) and Vaughan, Stoev, and Michailidis (2013). The methods developed in these and many other papers are designed to study networks and anomalies different from the power grid anomalies.

Power grid faults are very different from typical data that has motivated the development of existing approaches and the theory that underlies them. They are typically based on the statistical likelihood principle that leads to procedures based on the likelihood ratio. Such procedures enjoy optimality properties, but only under specific assumptions, almost always involving independence and often normal distribution. Power grid data satisfy none of these assumptions. All lines are connected and a fault propagates almost immediately through an affected subgrid. As the figures in Section 2 show, the fault data are very dependent. For the detection of a fault, this is a blessing, but it makes the problem of the localization of a fault very hard (measurements at all buses look similar). Our method is specific to a power grid and has no analog in existing network anomaly detection methods.

The paper is organized as follows. Section 2 introduces the grid data we work with and presents exploratory data analysis that motivates our approaches. In Section 3, we derive our detection and localization algorithms, present them by means of mathematical formulas and pseudocode, and assess their performance. Section 4 is dedicated to the discussion of existing or proposed engineering approaches and a comparison with another method. We conclude with a brief summary and main conclusions in Section 5.

2. Data description

Real power grid fault data are not publicly available mostly due to liability concerns of power companies. Most studies use simulated data. Data used for the development of the statistical methodology proposed in this paper were generated within the framework of the minniWECC , a simulation system designed for the evaluations and testing potential system-wide damping control technologies. The term minniWECC is used to reflect its utilization as a simplified model for the US grid portion managed by Western Electricity Coordinating Council (WECC). The minniWECC represents the overall inter-area modal properties and has enough complexity to reflect the relevant properties of the full western interconnection's bulk power system. This model is geographically consistent, and it matches with properties of the actual system. Results derived on minniWECC are transferable to the real-world grid. The data generated from the minniWECC model have been widely used to validate various power system mode estimation and event detection algorithms which are now adopted for control room applications, see e.g. Follum, Pierre, and Martin (2017); Trudnowski, Kosterev, and Undrill (2013); Byrne, Concepcion, Neely, Wilches-Bernal, Elliott, Lavrova, and Quiroz (2016). There are eight regional power grids covering the US and Canada. Similar regional grids exist in Europe, e.g. the British grid, the Nordic regional grid and the Baltic regional grid that are interconnected with the synchronous grid of Continental Europe.

A simplified one-line diagram of the minniWECC, following Trudnowski *et al.* (2013), is given in Figure 1. The minniWECC version we use consists of 171 lines and 122 buses. Each line $l \in \{1, \ldots, 171\}$ has terminal buses $(i, j) : i, j \in \{1, \ldots, 122\}$, $i \neq j$. We consider only pairs (i, j) for which there is a line l connecting buses i and j.

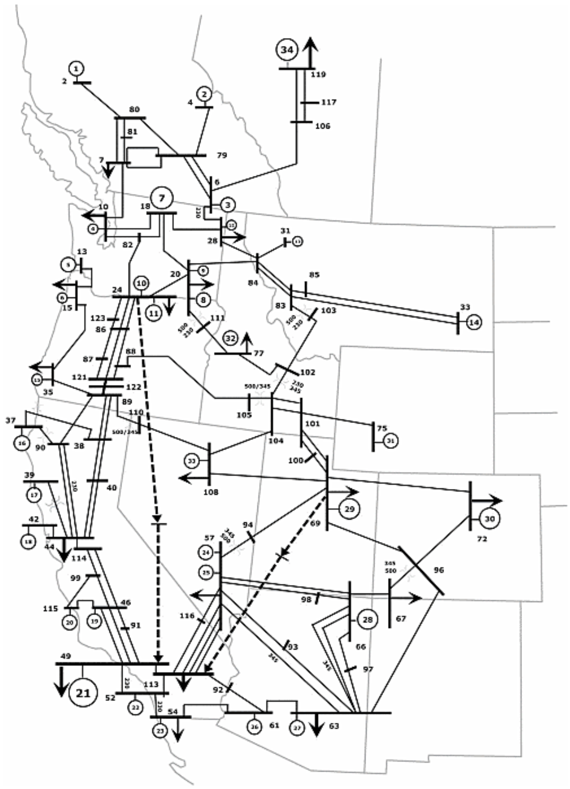

Figure 1: One-line diagram of minniWECC

We simulated a large number of faults within the minniWECC. At the end of minute 10 from the beginning of each simulation, one of four fault types was applied to one of the 171 lines, and the data were recorded up to the end of minute 15. This resulted in 684 different simulations. For each simulation, the faulted line and the type of fault are known. The following fault types were generated:

line to line (LL): two phases of a three phase circuit are short circuited.

line to ground (LG): one conductor comes in contact with the ground or the neutral conductor.

line to line to ground (LLG): any two phases of the power circuit are short circuited to ground or neutral.

three-phase (TP): all three phases of the power circuit are shorted.

The minniWECC data were generated using the Power System Toolbox (PST), Cheung, Chow, and Rogers (2009), which is based on MATLAB. The input data files required to run the simulations were created by Dr. D. Trudnowski. The details of how the simulations are carried out is well documented in the PST manual. The faults were generated using the switching condition matrix, in which the time of fault, time of clearing the fault, total duration of simulation and sampling rate were specified. The values of zero sequence ($z0$) impedance and negative sequence (zn) impedance were also specified in the switching condition matrix. For the generated data the value $z0$ and zn were chosen randomly. These values, in per unit (p.u), are $z0 = 0.17$ and $zn = 0.4$. The elements in the system model are the same for all types of faults. The sequence networks and the fault impedance vary for different types of faults. The equations used for calculating the impedance after different faults, in the simulation are:

three phase fault: $zf = 0$

line to ground fault: $zf = zn * \frac{z0}{zn+z0}$

line to line to ground: $zf = zn + z0$

line to line: $zf = zn$

In the above formulas, zf is the impedence with respect to ground after a fault.

Cases where simulation terminated due to numerical instability, are not included in further analysis. This results in a total of 546 simulations, where 117 simulations are from the TP fault cases, 139 simulations are from LLG cases, 145 simulations are from LL cases, and 145 simulations are from LG cases.

There are 116 out of 122 buses that have at least one line starting from it and can be considered as start buses. The number of possible indexes i in the (i, j) pairs is 116. For 91 out of these 116 i values, there exists only one unique j, i.e. there is only one line starting at i.

The simulated data consist of 120 measurements per second for 900 seconds (15 minutes). They include voltage readings at all buses reported as complex numbers. We consider the modulus of voltage at bus readings. The dimension of the dataset for each fault simulation is 122×108000. A fault is applied in the time interval $(600 - \frac{1}{120}, 600)$ in seconds, which is between observations 72000 and 72001 in each dataset. We set observation 72001 to be an exact fault time and so assume that second 600 is the earliest moment when the fault can be detected. To illustrate, we consider bus 105, which is the start bus of line 96. We gathered magnitude of voltage readings at bus 105 from all available simulations and highlighted simulations where line 96 was faulted. As different periods of the experiments have different variance of data, we represent the data by splitting the records into several time intervals (in seconds): $[0, 600 - \frac{1}{120})$, $[600 - \frac{1}{120}, 600 + \frac{1}{6})$, $[600 + \frac{1}{6}, 900]$. These intervals reflect the data before the fault, just after the fault including one point before the fault, and recovery after the fault. The beginning of the recovery phase is selected based on exploratory analysis. The graphs are shown in Figure 2. Notice that during the period $[0, 600)$ the data between simulation only differ by white noise. Bus 105 shows different responses to faults of different types. TP fault at line 96 has the largest effect on the readings at bus 105. For all fault types, most variability can be seen in the interval $[600, 600\frac{1}{6}]s$ (second panel in Figure 2). Approximately 0.1s after the fault, the recovery of the system begins. Figure 2 shows that one can expect to identify the start bus of the faulted line because it shows a special behavior. On the other hand, this special behavior is more pronounced for some fault types than for other fault types, so the task requires careful consideration. Figure 2 also shows that localization of the faults should be achievable within a second after a fault.

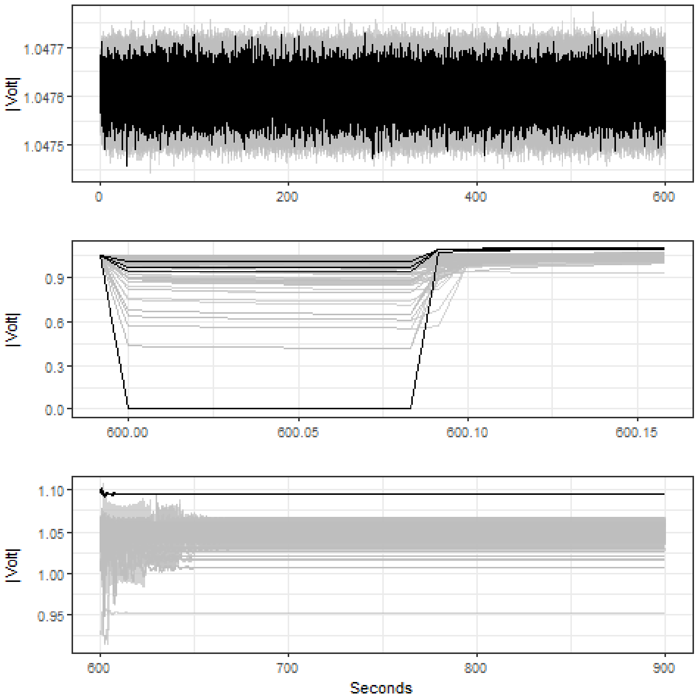

Figure 2: Modulus of voltage readings at bus 105 from all available (546) simulations. Faults at line 96 are highlighted (black) as bus 105 is the start bus of line 96. Readings resulting from faults at other lines (not 96) are plotted in gray. The three graphs represent different time stages of simulations. Notice the different voltage scales and different lengths of time intervals at the different stages.

Figure 3 presents the data in the interval $[600 - \frac{1}{120}, 600 + \frac{1}{6})$ from a different angle. The second panel of Figure 2 shows how a fixed bus “sees” the faults at all lines. Figure 3 shows how all buses see faults at a fixed line within the critical time interval just after a fault. To illustrate the differences in behavior between the buses, we consider faults at line 90. This line connects buses 88 and 121. The response to a fault at a specific bus depends on its relative location relative to the faulted line. For a given fault, the responses at all buses exhibit similar patterns; the size of the response is generally the largest at busses where the faulted line starts and ends, but the pattern is not obvious. Figure 3 shows that an almost instantaneous localization of a faulted line should be possible, but the task is clearly not trivial. The next section is devoted to a systematic study of this problem.

3. Derivation of fault detection and localization algorithms

The goal of the fault detection algorithms is to estimate the time M when the fault occurs and the goal of localization algorithms is to identify the faulted line. Localization is equivalent to finding the pair of buses (i_0, j_0) such that i_0 is the start bus and j_0 is the end bus of the faulted line l_0. Since we have only readings at buses, it is not possible to identify the faulty line in cases where there is more than one line that has the same start and end buses. In such cases, we can only identify the pair (i_0, j_0).

In order to describe our algorithms using mathematical formulas, it is convenient to introduce simple notation. We denote by $\mathbf{A}$ the set of all *pairs* of buses that are connected by at least

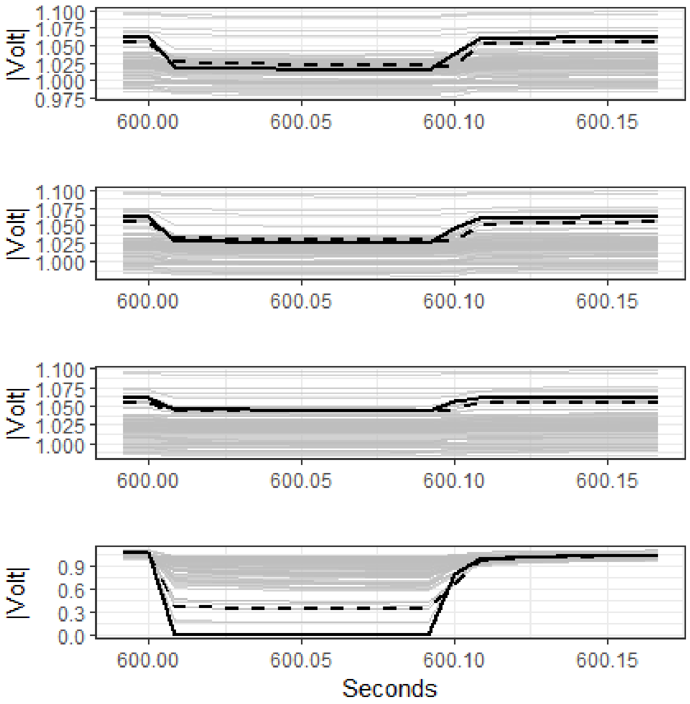

Figure 3: Responses at all buses to four fault types at line 90: LG, LL, LLG, and TP, from top to bottom. Responses at buses 88 and 121, which are connected by line 90, are highlighted in black; bus 88 in solid black, bus 121 in dashed black.

one line, i.e.

$$\begin{aligned}\mathbf{A} = \{(i,j):\ i,j \in 1,\dots,122,\ i \neq j \text{ , there is line} \\ l \in \{1,\dots,171\},\ \text{such that start bus of line } l \text{ is } i \\ \text{and end bus is } j\}. \qquad (1)\end{aligned}$$

Define by $\mathbf{B}$ the set of buses that are start buses of some line, i.e. $\mathbf{B} = \{i:$ there is $j \in \{1,\dots,122\}, (i,j) \in \mathbf{A}\}$. For each $i \in \mathbf{B}$, define set $\mathbf{B}_i$ by $\mathbf{B}_i = \{j : (i,j) \in \mathbf{A}\}$. For each $i \in \mathbf{B}$, $\mathbf{B}_i$ contains all possible end buses for lines that starts from bus i. The goal of a fault localization algorithm is to predict $(i_0, j_0) \in \mathbf{A}$, where $i_0 \in \mathbf{B}$ and $j_0 \in \mathbf{B}_{i_0}$. The prediction is denoted as $(\hat{i_0}, \hat{j_0})$.

We model the data as follows. In each simulation, we have a sample of random functions, or fine resolution time series, $x_i(t)$, where $i = 1\dots,N$. In MinniWECC, we have $N = 122$ (the number of buses). The value $x_i(t)$ is the modulus of the voltage at bus i at time t. Define δ as the time difference, in seconds, between two consecutive time points. Then $t = 0, \delta, 2\delta, \dots, T$ is the time domain. In this paper, $\delta = \frac{1}{120}$ and $T = 60 \cdot 15 = 900$ (seconds). Define M as the time in seconds when the fault occurs. In our simulations, a fault occurs at the beginning of second 600, thus $M = 600$s at each simulation run. This value is known to us, but not to any algorithms. The same applies to the location of the faulted line. The proposed procedure for fault detection consists of three stages: 1) detect an event of a fault in the grid in real-time; 2) identify the start bus of the faulted line; 3) identify the end bus of the faulted line.

The evaluation of algorithms on the entire data set might lead to procedures that fit the dataset well, but do not lead to a preferred algorithm that generalizes well. Therefore, in this paper, the data analysis and algorithm development are done on training data, and performance is evaluated using test data. The training dataset has been created by randomly, uniformly sampling 436 simulations (80%) out of 546 available simulations, The remaining 110 simulations are set aside as the test dataset. This is typically called a train-test split approach to algorithm development and evaluation.

3.1. Detection of the time a fault event

The goal is to identify the time point M when the fault occurs. In the proposed monitoring procedure, we consider a moving window, $[t - S_0 - S_1, t]$, where t is the current moment of time, $S_0 \geq \delta$ is the length, in seconds, of the window ending at $t - S_1$, and $S_1 \geq 0$ is the length, in second, of the window ending at t. This is illustrated in Figure 4. The value $S_1 = 0$ corresponds to the case of taking only a single observation (at time t). A moving window ensures that a detection statistic can be calculated in real-time, as it is based on only a limited number of observations, and that it adjusts to the most recent state of the system. A slow evolution of system readings does not indicate a fault of the type we are aiming to detect.

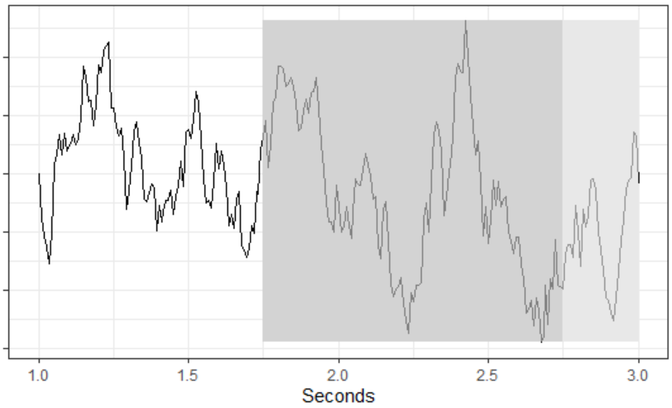

Figure 4: The moving window at $t = 3$ seconds with $S_0 = 1$s and $S_1 = 0.24$s. The light grey area represents the window $(t - S_1, t]$ that is used to evaluate (4). The darker gray area represents the window $[t - S_0 - S_1, t - S_1)$ that is used to evaluate (2) and (3).

To evaluate the state of the system up to time $t - S_1$, we use statistics $\bar{x}_i(t, S_0, S_1)$ and $\mathrm{SD}_i(t, S_0, S_1)$ defined for each bus i as follows:

$$\bar{x}_i(t, S_0, S_1) = \frac{1}{S_0/\delta} \sum_{k=1}^{S_0/\delta} x_i(t - S_1 - k\delta), \tag{2}$$

$$\mathrm{SD}_i(t, S_0, S_1) = \left(\frac{1}{S_0/\delta - 1} \sum_{k=1}^{S_0/\delta} (x_i(t - S_1 - k\delta) - \bar{x}_i(t, S_0, S_1))^2 \right)^{1/2}. \tag{3}$$

These are just the mean and the standard deviation of the magnitude of the voltage at each bus in the moving window $[t - S_0 - S_1, t - S_1)$ in their native resolution. To assess the most recent state of the system, we calculate the averages in the window $[t - S_1, t]$, i.e.

$$m_i(t, S_1) = \frac{1}{1 + S_1/\delta} \sum_{k=0}^{S_1/\delta} x_i(t - k\delta), \tag{4}$$

Notice that $m_i(t, S_1 = 0) = x_i(t)$. For each bus i, we define fault detector $D_i(t, S_0)$ as

$$D_i(t, S_0, S_1) = \frac{|m_i(t, S_1) - \bar{x}_i(t, S_0, S_1)|}{\mathrm{SD}_i(t, S_0, S_1)}. \tag{5}$$

The detector $D_i(t, S_0, S_1)$ is the absolute difference between the means in S_0 second before $t - S_1$ and S_1 seconds before t (including t) normalized by the standard deviation of the observations in the moving window $[t - S_0 - S_1, t - S_1)$.

To check if a fault occurred at time t, we first compute

$$i_{\max} = \arg\max_i D_i(t, S_0, S_1). \tag{6}$$

The index $i_{\max}$ identifies the bus, which is the most "perturbed" at time point t. If the perturbation is large enough, the algorithm should detect the fault. In order to evaluate the size of the perturbation, we introduce a threshold parameter $\tau > 0$. The algorithm detects a fault, if

$$D_{i_{\max}}(t, S_0, S_1) = \max_i D_i(t, S_0, S_1) > \tau. \tag{7}$$

If condition (7) fails, we move on to the next time point and perform the calculations again. The above procedure is summarized in Algorithm 1.

Algorithm 1 Detection of a fault in a grid

```
Input: Tuning parameters $\tau$, $S_0$, and $S_1$, input data $\mathbf{X}$, time resolution $\delta$, and upper time limit $T$;
Output: $t_D$ {The time where the fault is detected}
Initialization $Fault = 0$ and $t = S_0 + S_1$ {Requires at least $S_0 + S_1$ seconds of data before the fault}
while $Fault = 0$ & $t \leq T$ do
    Set $t := t + \delta$
    Find $D_i(t, S_0, S_1)$ for each $i$ (5)
    Find $i_{\max} = \arg\max_i D_i(t, S_0, S_1)$ (6)
    if $D_{i_{\max}}(t, S_0, S_1) > \tau$ then
        Set $t_D := t$ and $Fault = 1$
    end if
end while
```

There are several tuning parameters in Algorithm 1: S_0, S_1 and τ. We want to determine the tuning parameters that balance two criteria: 1) if there is no fault, false alarms should be rare, 2) if there is a fault, it should be detected with a large probability and the delay of detection (time of detection - time of fault) should be small. In traditional online monitoring problems, larger τ ensures that criterion 1) is met, and smaller τ ensures that criterion 2) is met. In power grids, the system is perturbed the most just after the fault happens, and the size of this perturbation is large; an occurrence of a fault is generally easy to spot.

We consider the following quantities:

F_1 - Fraction of simulation in which a fault is detected over the first 10 minutes. This quantity is similar to the size of a test or type I error.

F_2 - Fraction of simulations in which a fault is detected over the whole 15 minutes. This quantity is similar to power under alternative or 1 minus type II error.

D - The first t (in seconds) such that $\max_i D_i(t, S_0, S_1) > \tau$ less 600s, averaged over all simulation where a fault is detected. The quantity D is thus the average time of detection minus the time at each the fault has occurred (600s in our experiments).

In traditional tests, we would like to have $F_1 \approx 0.05$. Since in a power grid, a false alarm may be expensive, we may target a different value of F_1. Even $F_1 = 0$ might be reasonable. This is the case for our simulated data. Once we determined S_0, S_1, τ that give the desired F_1, we compute for them the fraction F_2 and select the values that give the largest F_2. We could then further select the tuning parameters such that D is the smallest possible. This general strategy can be applied to any regional grid, as long as software for simulating faults at the grid of interest is available. Details of its implementation are explained below, cf. the discussion around Tables 1 and 2.

The parameter S_0 determines how much prior data we need to detect a fault. In Algorithm 1, this parameter directly affects how accurate the estimates of the mean and standard deviation are. Thus S_0 should not be too small as a standard deviation estimate using just a few points is not accurate. The parameter S_1 is related to the detection delay, so it should be as small as reasonable. To determine values of S_0, S_1, and τ, we need to examine the behavior of the detector $\max_i D_i(t, S_0, S_1)$ over the first 600 seconds and during the fault. To explain, we begin with two pairs of parameters: $S_0 = 30\delta, S_1 = 0$ and $S_0 = 480\delta, S_1 = 11\delta$. Since the time step δ is equal to (1/120)s, these values correspond, respectively, to pairs of intervals of lengths (0.25s, (1/120)s) and (4s, (1/12)s). Values of the detector are larger for smaller values of S_0 and S_1, as shown in Figure 5. This means that useful values of τ must depend on S_0 and S_1, as illustrated in Figure 5.

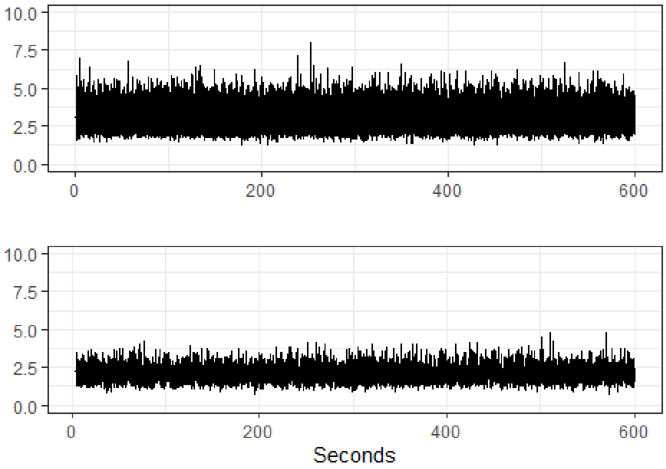

Figure 5: Values of the detector $\max_i D_i(t, S_0, S_1)$ over the first 600 seconds in one of the training simulations (no faults). Upper panel: $S_0 = 30\delta, S_1 = 0$, lower panel: $S_0 = 480\delta, S_1 = 11\delta$.

Table 1: The maximum of the detector $\max_i D_i(t, S_0, S_1)$ over a single simulation in the training sample for $t \leq 600$ (before a fault) for various parameters S_0 and S_1 ($S_1 = 0$ corresponds to a single data point). For fixed S_0 and S_1, the threshold τ should be larger than the values in the table.

	$S_1 = 0$	$S_1 = 4\delta$	$S_1 = 11\delta$	$S_1 = 19\delta$
$S_0 = 15\delta$	9.84	15.16	18.26	19.79
$S_0 = 30\delta$	8.13	10.27	12.75	12.97
$S_0 = 120\delta$	5.97	5.77	6.61	7.40
$S_0 = 240\delta$	5.58	5.24	5.61	5.31
$S_0 = 480\delta$	5.31	4.85	4.78	5.31
$S_0 = 960\delta$	5.17	4.86	4.64	4.21
$S_0 = 1920\delta$	5.22	4.75	4.59	4.16
$S_0 = 7200\delta$	5.37	4.95	4.70	4.19

As every simulation is conducted identically before the fault at second 600, there is no need to examine the behavior of the detector for all simulations. However, after the fault, each simulation exhibits a different behavior as the location and type of the fault differ from one simulation to another. To understand the behavior of the detector in (7) at the time of the fault, one must therefore examine the whole training data set. A fault is detected if the detector exceed a threshold τ, so we must determine the minimum values of the detector at the time of the fault. These are presented in Table 2.

During a fault, the values of $\max_i D_i(t, S_0, S_1)$ given in Table 2 are noticeably larger than the corresponding values given in Table 1. The quantity F_2 (power of detection) can be maximized using a threshold τ that does not exceed the values in Table 2. *The specific choice of S_0 and S_1 should be the one that maximizes the gap between corresponding values between S_0 and S_1 in Tables 1 and 2.* One can see that the values should be $S_0 = 30$ and $S_1 = 0$, even though all other values would be acceptable. Focusing on $S_0 = 30\delta$ and $S_1 = 0$, we see that we can take $\tau = 15$, but many other values will work too. Taking $S_0 = 30\delta, S_1 = 0, \tau = 15$ gives $F_1 = 0$ and $F_2 = 1$ and $D = 0$ on the training data set.

Table 2: The value of the detector $\max_i D_i(t, S_0, S_1)$ at $t = 600$ (time of fault) over all simulations in the training dataset.

	$S_1 = 0$	$S_1 = 4\delta$	$S_1 = 11\delta$	$S_1 = 19\delta$
$S_0 = 15\delta$	288.97	63.21	27.35	7.98
$S_0 = 30\delta$	244.72	36.22	15.18	10.58
$S_0 = 120\delta$	190.05	38.16	15.79	9.04
$S_0 = 240\delta$	133.16	26.57	11.31	6.75
$S_0 = 480\delta$	134.13	26.78	11.49	6.98
$S_0 = 960\delta$	157.24	31.64	13.66	8.29
$S_0 = 1920\delta$	175.12	35.30	15.25	9.26
$S_0 = 7200\delta$	178.11	35.91	15.51	9.44

We summarize the general procedure for selecting the window lengths S_1, S_0 and the threshold τ.

1. Consider several potential values of S_0 and several values of S_1.

2. For each pair (S_0, S_1) in step 1, compute $\tau_{\max,0} = \max_{t \in \mathcal{T}_0} \max_i D_i(t, S_0, S_1)$, where $\mathcal{T}_0$ is the time interval with no faults.

3. For each pair (S_0, S_1) in step 1, compute $\tau_{\min,F} = \min_{t \in \mathcal{T}_F} \max_i D_i(t, S_0, S_1)$, where $\mathcal{T}_F$ is a short time interval containing the time of the faults. (The simulations must be synchronized so that all faults occur at the same time; we used $\mathcal{T}_F$ to be a single point set equal to the time of the fault.)

4. Select the pair (S_0, S_1) maximizing $\tau_{\min,F} - \tau_{\max,0}$.

5. Set $\tau = (\tau_{\min,F} + \tau_{\max,0})/2$.

As this is our final methodology for the detection of the time of the fault (derived from the training dataset), we tested it on the test dataset as well. Using $S_0 = 30, S_1 = 0, \tau = 15$ leads to $F_1 = 0$, $F_2 = 1$, and $D = 0$ on the test dataset. We conclude that there is no room for improvement in terms of performance.

The investigations reported in this section show that it is relatively easy to detect a fault in a regional grid if measurements at *all* buses are available. The parameters of the detector can be chosen in such a way that the detection is instantaneous, there are no false alarms and each fault is detected. One might be concerned that false alarms may occur due to the system evolution, e.g., load shedding or demand re-dispatch. To address this issue with confidence, one would need to simulate a large number of normal changes in the power system. Intuitively, the faults that we have in mind are sudden and large, as demonstrated in the figures shown above. The thresholds in our algorithms are trained for such large faults and they are unlikely to relatively small (in terms of the whole system), normal load changes.

3.2. Identification the faulted line

Once a fault has been detected, we want to determine the line at which it had occurred. This means that we must identify the pair (i_0, j_0) of terminal buses of the faulted line l_0. Our algorithm does it in a sequential manner. We first identify a bus that exhibits the "most anomalous" behavior is a sense that will be quantified. We call such a bus the start bus of the faulted line. Then, we identify the end bus from among those that are connected to the start bus by a single line.

Denote by t_D the time of detection, i.e. the first t such that $\max_i D_i(t, S_0, S_1) > \tau$. A natural choice for the prediction of the start bus of the faulted line would be the bus i_{max} defined in (6) with t replaced by t_D. This approach indeed works reasonably well, but it can be improved. We obtained more accurate identification of the start bus by simply comparing the averages, i.e. by setting

$$\hat{i}_0 = \arg\max_{i \in \mathbf{B}} |m_i(t_D, S_1) - \bar{x}_i(t_D, S_0, S_1)| \tag{8}$$

Restricting the index i to the set $\mathbf{B}$ simply means that we consider only buses which are start buses of some line. If we use the values $S_1 = 0$ and $S_0 = 30\delta$ arising from the investigations reported in Section 3.1, we are looking at the difference between a single observation at the time of detection and the average of observations over the prior 0.25s. Using equation (8) with $S_0 = 30\delta$, and $S_1 = 0$, the start bus i_0 of the faulted line l_0 was identified correctly in 422 out of 436 simulations (422 cases where $\hat{i_0} = i_0$). This gives a 97% success rate.

Additional analysis shows that the 14 incorrectly identified instances ($\hat{i_0} \neq i_0$) relate to only three buses: 121, 122, and 89. The potential reason for failure in these cases could be that the magnitudes of the voltage at buses 121 and 89 correlate perfectly before the fault (correlation is equal exactly to 1). This could be due to a simulation setup. Our proposed methodology tends to favor the wrong bus. For example, in cases where bus 89 is the start bus of the faulted line, relation (8) tends to select Bus 122. More complete results are given in Table 3.

Table 3: Prediction of the start bus of the faulted line over all training simulations for buses 89, 121 and 122

		Predicted		
		Bus 89	Bus 121	Bus 122
Actual	Bus 89	4	0	10
	Bus 121	3	0	0
	Bus 122	1	0	0

Once the start bus of the faulted line l_0 is predicted as $\hat{i}_0$, we need to predict the end bus of the faulted line. We denote this prediction by $\hat{j}_0$. Notice that once we have $\hat{i}_0$, the prediction $\hat{j}_0$ must be from $\mathbf{B}_{\hat{i}_0}$. Otherwise, the l_0 prediction would not make sense as $\hat{i}_0$ and $\hat{j}_0$ would not define a line. This greatly narrows the potential options for $\hat{j}_0$. If there is only one line originating at $\hat{i}_0$, then $\hat{j}_0$ is its end bus. However there are cases, where multiple lines originate at $\hat{i}_0$. Thus, if there are indexes j_a and j_b, such that $j_a \neq j_b$ and $j_a, j_b \in \mathbf{B}_{\hat{i}_0}$, an approach is needed to choose $\hat{j}_0$ from $\mathbf{B}_{\hat{i}_0}$. To find a prediction of the end bus of the faulted line, we propose and test two different procedures. The main difference between them is the time between the fault detection and the time when the prediction of j_0 is made. One may expect that an additional time delay in the identification of the end bus may increase the accuracy of the predictions, as the algorithm can utilize more information.

Notice that if the start bus prediction $\hat{i}_0$ is incorrect, the prediction of l_0 will be automatically incorrect. Thus for the end bus prediction, we only consider simulations, where i_0 was identified correctly. Using equation (8), we identified the start bus correctly in 422 out of 436 training simulations. Also, in 222 out of 422 simulations with correctly identified $\hat{i}_0$, $\mathbf{B}_{\hat{i}_0}$ has only one element, so the determination of $\hat{j}_0$ is trivial. The following algorithms are designed to identify the j_0 in the 200 training simulations, where the start bus was predicted correctly and there is more than one element in $\mathbf{B}_{\hat{i}_0}$.

In the first approach, j_0 is predicted at time t_D. This algorithm predicts the end bus of the faulted line similarly to equation (8), where the potential end buses must be in $\mathbf{B}_{\hat{i}_0}$, i.e.

$$\hat{j}_0 = \arg\max_{j \in \mathbf{B}_{\hat{i}_0}} |m_j(t_D, S_1) - \bar{x}_j(t_D, S_0, S_1)| \tag{9}$$

This algorithm thus predicts (i_0, j_0) at time t_D, the time of the fault detection. Using equation (9) with $S_0 = 30\delta$ and $S_1 = 0$, the end bus was correctly identified in 71 out of 200 simulations. This gives only 35% accuracy.

To describe the second approach, we need to introduce additional notation. We define the recovery process of bus i as

$$D_i^*(t, t_D, S_0, S_1) = m_i(t, S_1) - \bar{x}_i(t_D, S_0, S_1), \quad t > t_D. \tag{10}$$

Compared to (9), in (10) we use the interval ending at t_D to evaluate the state of the system before the fault, but push the current time forward after the fault. To illustrate, we show in Figure 6 the process (10) over the period of 0.25s after the fault (with $S_0 = 30\delta$, $S_1 = 0$). We do so for three training simulations. Notice that for each simulation, $D_i^*(t, t_D, S_0, S_1)$ undergoes a visible change approximately after 0.1s. This remains true for the remaining training simulations with the magnitude of change varying from simulation to simulation.

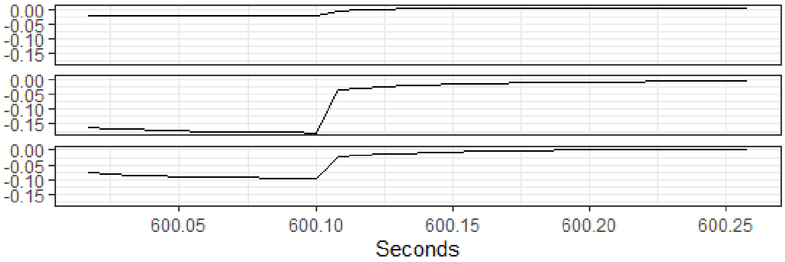

Figure 6: The recovery process (10) for three different training simulations ($S_0 = 30\delta$, $S_1 = 0$)

To determine the moment when recovery begins, we compute the absolute value of relative change and test the condition

$$\left| \frac{D_{\hat{i}_0}^*(t, t_D, S_0, S_1) - D_{\hat{i}_0}^*(t - \delta, t_D, S_0, S_1)}{D_{\hat{i}_0}^*(t - \delta, t_D, S_0, S_1)} \right| > \tau_1. \tag{11}$$

Examples illustrating the behavior of the left-hand side of (11) are shown in Figure 7. The time of the recovery, t_R, is defined as the smallest $t > t_D$ such that condition (11) is met. Its computation requires a selection of the value of τ_1. This threshold must be chosen sufficiently small to ensure that condition (11) is met for all simulations in the training dataset over a reasonably short time interval after a fault has been detected. Evaluating all potential $\tau_1 \in (0.01, 1)$ in increments of 0.01, shows that we need $\tau_1 \leq 0.17$ if we require $t_R \in (t_D + \delta, ct_D + 1)$ (the δ refers to (1/120)s and the 1 to 1s). The value $\tau_1 = 0.1$ is one of several values that maximize the performance of the algorithm we now describe on the training dataset, and this is the value we use in the following.

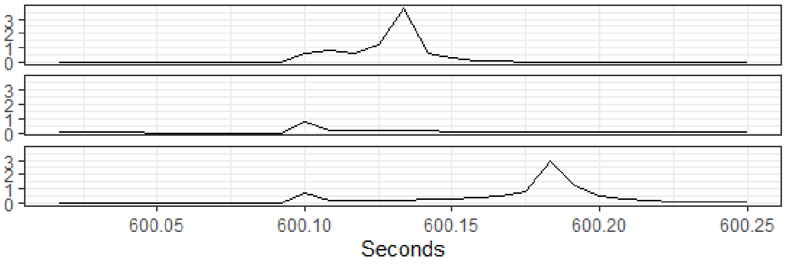

Figure 7: The left-hand side of (11) for three predicted start buses ($S_0 = 30\delta$, $S_1 = 0$)

We now describe an algorithm that identifies j_0 at time t_R. For each bus $j \in \mathbf{B}_{\hat{i}_0}$, we consider the recovery process $D_j^*(t, t_D, S_0, S_1)$ defined by (10). Examples illustrating its behavior are shown in Figure 8. Approximately 0.1s after the fault, the buses in $\mathbf{B}_{\hat{i}_0}$ exhibit differing behaviors. The recovery process of the end bus of the faulted line shows somewhat different behavior compared to other buses in $\mathbf{B}_{\hat{i}_0}$. It generally undergoes the most pronounced change

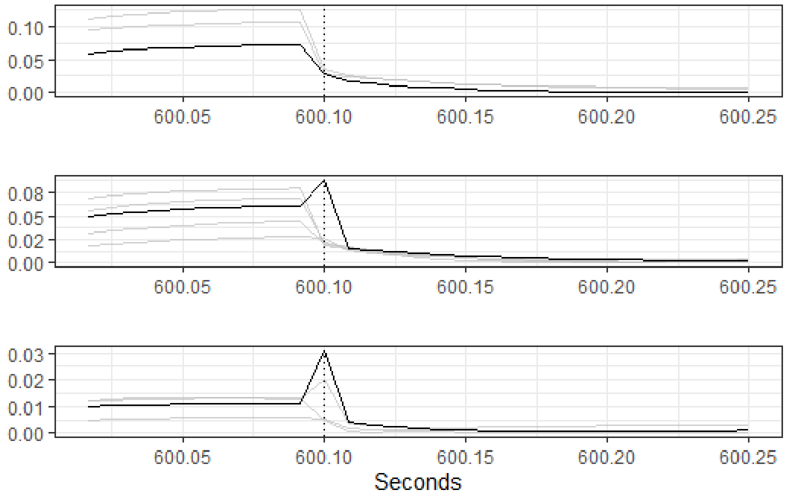

Figure 8: The absolute value of recovery process (10) for end buses in $\mathbf{B}_{\hat{i}_0}$ for three different training simulations ($S_0 = 30\delta$, $S_1 = 0$). In each of the three panels, the absolute value of recovery process corresponding to the correct end bus is highlighted (black). The vertical dotted line indicates the time t_R where $\tau_1 = 0.1$ is used.

between 0.09s and 0.1s after the fault. To distinguish the end point of the faulted line from other end points, we use differences in absolute values of recovery process formula, i.e.

$$|D_j^*(t, t_D, S_0, S_1)| - |D_j^*(t-\delta, t_D, S_0, S_1)|, \quad t > t_D, \tag{12}$$

Examples illustrating the behavior of differences (12) are shown in Figure 9. Notice that the value of (12) for the end bus of the faulted line 0.1s after the fault is the largest compared to other buses in $\mathbf{B}_{\hat{i}_0}$. Our algorithm thus predicts j_0 as

$$\hat{j}_0 = \arg\max_{j \in \mathbf{B}_{\hat{i}_0}} (|D_j^*(t, t_D, S_0, S_1)| - |D_j^*(t-\delta, t_D, S_0, S_1)|). \tag{13}$$

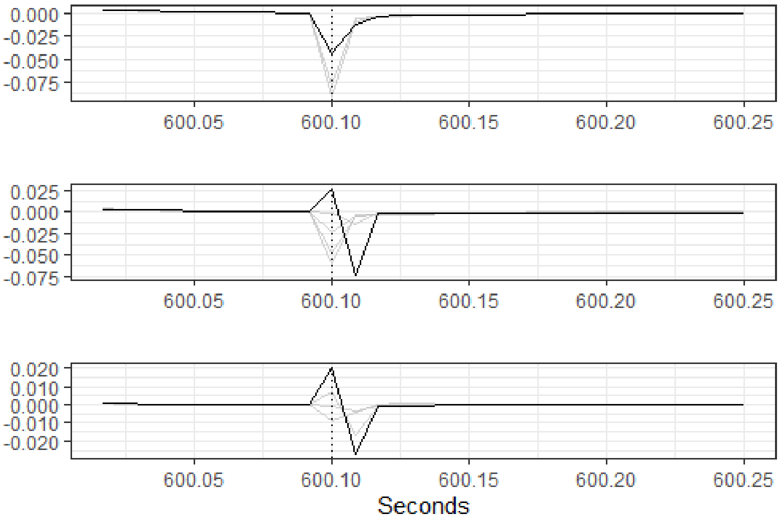

Figure 9: Differences in absolute value of recovery process (12) for end buses in $\mathbf{B}_{\hat{i}_0}$ for three different training simulations ($S_1 = 0$). The line for the correct end bus is highlighted (black). The vertical dotted line indicates time t_R where $\tau_1 = 0.1$ is used.

The above algorithm correctly identified the end bus in 179 out of 200 applicable simulations, 89% accuracy. Using (9), we only got 35% accuracy. We see that by delaying the identification of the end bus by 1s at most (in training simulations such value did not exceed 0.11s), we increased the performance by 54% on the training simulations. We thus recommend the algorithm based on (11). If we include the identification of the bus of the faulted line, we conclude that our methodology correctly identifies the faulted line in 401 simulations out of 436 training simulations. To summarize, our final methodology is defined by relations (8) and (13) with $S_0 = 30\delta$, $S_1 = 0$ and $\tau_1 = 0.1$. The procedure is summarized in Algorithm 2.

On the test dataset, our methodology correctly identifies the faulted line in 92% of simulations (102 simulations out of 110). The start bus of the faulted line was identified correctly in 95% validation simulations (104 cases out of 110). In simulations where the start bus i_0 was identified correctly and where was more than one eligible choice for j_0, the end bus was correctly identified in 97% of instances (56 simulations out of 58).

Algorithm 2 (i_0, j_0) prediction using time points t_D and t_R

```
Input: Tuning parameters $\tau_1$, $S_0$, and $S_1$, input data $\mathbf{X}$, time resolution $\delta$, network description data $\mathbf{Y}$, the fault moment $t_D$
Output: $\hat{i}_0, \hat{j}_0$
Find $\hat{i}_0$ using (8)
if $|\mathbf{B}_{\hat{i}_0}| = 1$ then
    Set $\hat{j}_0 = j_0$, where $j_0 \in \mathbf{B}_{\hat{i}_0}$
else
    Find $t_R$ using (11)
    Find $\hat{j}_0$ using (13)
end if
```

4. Discussion of other approaches

Faults in power grids have been the subject of intense research. In general, existing fault detection schemes in power systems can be categorized into the following three approaches: quantitative model-based, qualitative model-based, and data-driven approaches Jiang, Chuang, Wang, Hung, Wang, Lee, and Hsiao (2011). Quantitative model-based approaches and qualitative model-based approaches could achieve good detection performance in simulations. However, when implemented in real-world applications, they are quite sensitive to the noise in the voltage and current measurements. Moreover, they assume that the system models are accurately given and their detection performance would be greatly affected by the inaccuracy in the system models. Recently, data-driven approaches begin to receive considerable attentions from the researchers due to the following reasons: 1) various intelligent electronic devices (IEDs) have been widely adopted and installed in the power grid Moghaddass and Wang (2017); Ghosal and Conti (2019); Chen, Hill, and Wang (2020), which collect large amount of different data at many nodes across the entire grid; and 2) compared to the traditional model-based approaches, the data-driven approaches are more resilient against measurement errors and system model inaccuracy. Meanwhile, they have more flexibility in their implementations and adapt better to the variations in system components and/or topology Chen, Jiang, Chen, and Yi (2018); Chen, Yi, Jiang, Zhang, and Chen (2019); Tripathi and De (2018); Yin, Li, Gao, and Kaynak (2014). Following Zhou, Wang, Srivastava, Wu, and Banerjee (2019), we note that event detection was studied using moving averages, Chow, Chakrabortty, Arcak, Bhargava, and Salazar (2007), principal components, Xie, Chen, and Kumar (2014), geographical visualization, Kaci, Kamwa, Dessaint, and Guillon (2014), wavelets, Kim, Chun, Yoon, Lee, and Shin (2017), dynamic programming, Cui, Wang, Tan, Florita, and Zhang (2019) and energy similarities, Yadav, Pradhan, and Kamwa (2019). Past work has considered fault classification, Nguyen, Barella, Wallace, Zhao, and Liang (2015), cascading events, Rafferty, Liu, Laverty, and McLoone (2016), and cyber events, Pan, Morris, and Adhikari (2015), Giani, Bitar, Garcia, McQueen, Khargonekar, and Poolla (2013), Liao and Chakrabortty (2019). The above list of citations is only meant to illustrate the scope of the research and is not exhaustive.

Several papers are closely related to our research. Before discussing them, we note that the chief and novel feature of our work is that we consider a large grid (122 buses) that mimics a real regional grid and a large number of faults. Existing research focuses on small test grids and methods are often illustrated on a single fault. Our work thus has important practical and statistical dimensions. In certain aspects, our approach is less sophisticated than many other approaches, but simplicity may be of advantage in practical applications. The complex structure of existing methods makes them difficult to implement on our grid data. Moreover, their descriptions generally omit the details of the implementation and no publicly available code is available. The papers discussed below make profound contributions, but the approaches they propose are difficult to implement using the information they provide. Another novel aspect of our approach is that it can identify the faulted line with high probability, not just an affected bus (which may have many connecting lines) or its neighborhood.

We now discuss selected papers to justify the points made above. Using the magnitude of the voltage, Gholami, Srivastava, and Panday (2019) develop advanced fault detection methods based on multiple detectors and likelihood computed from an ensemble model. The approach of Hannon, Deka, Jin, Vuffray, and Lokhov (2019) requires more data than our approach - current, frequency, voltages, and the application is concerned with one bus. The focus of Ardakanian, Yuan, Dobbe, von Meier, Low, and Tomlin (2017) is the estimation of the admittance matrix. A change in this matrix and its localization provide information on the timing and the localization of a fault. The algorithm is evaluated on a 13 bus grid and one fault. In Pandey, Srivastava, and Amidan (2020) the authors provide a method to find the bus (PMU) and the subgraph where the fault occurred. This is similar to finding the start bus of the faulted line in our approach. Similarly, Li, Deka, Chertkov, and Wang (2019) develop methodology to detect the bus that is close (in a small neighborhood) to the faulted line.

To provide some idea on the advance our method makes relative to existing statistical approaches to change point detection, we implemented the standard moving window CUSUM approach which is described in many textbooks, see e.g. Brodsky and Darkhovsky (1993). We consider a moving window, $(t - K\delta, t]$. The lag K is similar to $(S_0 + S_1)/\delta$ and corresponds to the number of points in a window. The algorithm is the same for each t, so it is easier to think about the beginning and the end of the window. We compute the averages before and after the potential change point (fault): $\hat{\mu}_i(k) = \frac{1}{k}\sum_{l=1}^{k} x_i(t - K + l\delta)$; $\tilde{\mu}_i(k) = \frac{1}{K-k}\sum_{l=k+1}^{K} x_i(t - K + l\delta)$, where $k \in \{1, \dots, K-1\}$. Next, we compute the normalized difference of these averages,

$$P_i(t,k) = \frac{k(K-k)}{K}[\hat{\mu}_i(k) - \tilde{\mu}_i(k)]$$

and the detector for a single bus

$$D_i(t,K) = \frac{1}{K-1}\sum_{k=1}^{K-1} P_i(t,k)^2.$$

The detector for the whole grid is

$$D(t,K) = \frac{1}{N}\sum_{i=1}^{N} D_i(t,K). \tag{14}$$

We signal a fault if $D(t,K) > \tau$. The value of τ would be determined by a procedure we employed in Section 3, but we can illustrate more convincingly that our method works much better by the examination of Figure 10 which shows that the detector (14) reacts to a fault much slower than our detector.

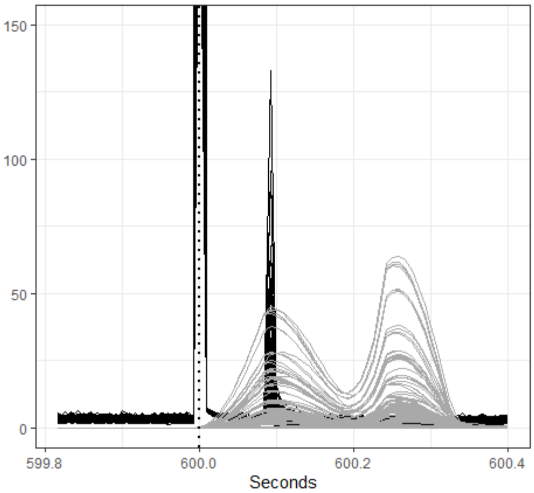

Figure 10: Comparison of the proposed method and the traditional moving window CUSUM approach. Curves in black show our detector calculated using (7) with $S_0 = 30\delta$ and $S_1 = 0$. Curves in gray are calculated using (14) with $K = 30$. The vertical dashed line represents the time of the fault. Both detectors have two spikes. For our proposed detector the first spike occurs almost simultaneously with the fault and is so high that it crosses the upper boundary of the display. The second, not relevant, spike of our detector coincides with the first spike of the traditional detector.

Intuitively, the detector based on equation (7) works better because instead of calculating statistics for each point of a moving window, we only calculate them at the windows $[t - S_0 - S_1, t - S_1)$ and $[t - S_1; t]$. This also make our method much faster. We introduce the normalization with the SD, as this helps to standardize the signals from each bus and tune up the τ properly. This is useful because the SDs at different buses are not necessarily the same. The identification of an appropriate threshold τ is more difficult in case of detector (14), but we do not discuss the details because our proposed detector works much better with a clear cut threshold.

5. Summary

We have proposed a two-stage procedure for fault monitoring in a regional power grid. In the first stage, a fault is detected. In the second stage, the faulted line is identified. Our methods are fully data-driven and require only knowing the start and end buses of each line. Our procedure assumes that a fault occurred on a line, which corresponds to most practical scenarios. If a fault occurs on a bus, the identification of the faulted bus can follow the algorithm for finding the start bus i_0 of the faulted line. While some faults are easier to detect than others, our method does not require prior knowledge of the fault type. The approach uses only the voltage modulus measured at buses as the input. The general framework can be extended to different inputs, e.g. current or power. The chief contribution of this work is to develop a general statistical approach to anomaly detection and localization suited to regional power grids. These grids can be viewed as networks or graphs, but methods developed for such data structures in different context are not applicable due to a very special form of power transmission networks and anomalies that can occur in them. Our approach can be adapted and fine-tuned to work with somewhat different regional grids, but the general paradigm could be followed.

We emphasize that the approach we presented assumes that PMUs are placed at all buses. As pointed out by a reviewer, so far, power grids in the real world are only partially covered by PMUs. While a complete coverage may be expected in the future, it is important to extend the method to grids with incomplete PMU coverage. Partial insights in this direction are reported in Rimkus, Kokoszka, Prabakar, and Wang (2022) who focus on fault detection (no localization) in a smaller distribution grid.

It may be of interest to compare our method to other approaches that may be developed in the future. To facilitate such comparisons, we placed our commented code at https://github.com/MantautasRimk The code is written in R R Core Team (2022). We are not aware of an existing method that can be readily applied to detect and localize a fault in a regional power grid to which our method can be compared. Several sound and promising approaches exist that could potentially be generalized to a regional grid, but they are not implemented in publicly available software.

Acknowledgements

The authors are grateful to Dr. Daniel Trudnowski of Montana Tech for his permission to use the minniWECC model. This research was partially supported by the United States National Science Foundation grants DMS–1923142, EECS–1828066 and OAC–1923983.

References

Ardakanian O, Yuan Y, Dobbe R, von Meier A, Low SH, Tomlin CJ (2017). "Event Detection and Localization in Distribution Grids with Phasor Measurement Units." *2017 IEEE Power & Energy Society General Meeting*, **2017**. doi:10.1109/PESGM.2017.8273895. URL http://arxiv.org/abs/1611.04653.

Bartos K, Rehak M, Krmicek V (2011). "Optimizing Flow Sampling for Network Anomaly Detection." In *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1304–1309. doi:10.1109/IWCMC.2011.5982728.

Basseville M, Nikifirov IV, Tartakovsky A (2012). *Sequential Analysis: Hypothesis Testing and Change–Point Detection*. Chapman & Hall/CRC. doi:10.1201/b17279.

Brodsky BE (2017). *Change-Point Analysis in Nonstationary Stochastic Models*. CRC Press. doi:10.1201/9781315367989.

Brodsky BE, Darkhovsky BS (1993). *Nonparametric Methods in Change–Point Problems*. Kluwer. doi:10.1007/978-94-015-8163-9.

Byrne RH, Concepcion RJ, Neely J, Wilches-Bernal F, Elliott RT, Lavrova O, Quiroz JE (2016). "Small Signal Stability of the Western North American Power Grid With High Penetrations of Renewable Generation." In *Proc. of 2016 IEEE 43rd Photovoltaic Specialists Conference (PVSC)*, pp. 1784–1789. Portland, OR. doi:10.1109/PVSC.2016.7749930.

Chen H, Jiang B, Chen W, Yi H (2018). "Data-Driven Detection and Diagnosis of Incipient Faults in Electrical Drives of High-Speed Trains." *IEEE Transactions on Industrial Electronics*, **66**(6), 4716–4725. doi:10.1109/TIE.2018.2863191.

Chen H, Yi H, Jiang B, Zhang K, Chen Z (2019). "Data-Driven Detection of Hot Spots in Photovoltaic Energy Systems." *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, **49**(8), 1731–1738. doi:10.1109/TSMC.2019.2896922.

Chen J, Gupta AK (2011). *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. Birkhäuser. doi:10.1007/978-0-8176-4801-5.

Chen T, Hill DJ, Wang C (2020). "Distributed Fast Fault Diagnosis for Multimachine Power Systems via Deterministic Learning." *IEEE Transactions on Industrial Electronics*, **67**(5), 4152–4162. doi:10.1109/TIE.2019.2917367.

Cheung KW, Chow J, Rogers G (2009). "Power System Toolbox, v 3.0." *Technical report*, Rensselaer Polytechnic Institute and Cherry Tree Scientific Software,. doi:10.1109/59.207380.

Chow JH, Chakrabortty A, Arcak M, Bhargava B, Salazar A (2007). "Synchronized Phasor Data Based Energy Function Analysis of Dominant Power Transfer Paths in Large Power Systems." *IEEE Transactions on Power Systems*, **22**(2), 727–734. doi:10.1109/TPWRS.2007.895162.

Csörgő M, Horváth L (1997). *Limit Theorems in Change-Point Analysis*. Wiley.

Cui M, Wang J, Tan J, Florita AR, Zhang Y (2019). "A Novel Event Detection Method Using PMU Data With High Precision." *IEEE Transactions on Power Systems*, **34**(1), 454–466. doi:10.1109/TPWRS.2018.2859323.

Dragalin VP, Tartakovsky AG, Veeravalli VV (1999). "Multihypothesis Sequential Probability Ratio Test–Part I: Asymptotic Optimality." *IEEE Transactions on Information Theory*, **45**, 2448–2461. doi:10.1109/18.796383.

Follum J, Pierre JW, Martin R (2017). "Simultaneous Estimation of Electromechanical Modes and Forced Oscillations." *IEEE Transactions on Power Systems*, **32**(5), 3958–3967. doi:10.1109/TPWRS.2016.2633227.

Gholami A, Srivastava A, Panday S (2019). "Data-Driven Failure Diagnosis in Transmission Protection System With Multiple Events and Data Anomalies." *Journal of Modern Power Systems and Clean Energy*, **7**, 767–778. doi:10.1007/s40565-019-0541-6.

Ghosal A, Conti M (2019). "Key Management Systems for Smart Grid Advanced Metering Infrastructure: A Survey." *IEEE Communications Surveys & Tutorials*, **21**(3), 2831–2848. doi:10.1109/COMST.2019.2907650.

Giani A, Bitar E, Garcia M, McQueen M, Khargonekar P, Poolla K (2013). "Smart Grid Data Integrity Attacks." *IEEE Transactions on Smart Grid*, **4**(3), 1244–1253. doi:10.1109/TSG.2013.2245155.

Gustafsson F (2000). *Adaptive Filtering and Change Detection*. Wiley. doi:10.1002/0470841613.

Hannon C, Deka D, Jin D, Vuffray M, Lokhov AY (2019). "Real-time Anomaly Detection and Classification in Streaming PMU Data." doi:10.48550/ARXIV.1911.06316.

Huang L, Nguyen X, Garofalakis M, Jordan M, Joseph A, Taft N (2007). "In–Network PCA and Anomaly Detection." In *Advances in Neural Information Processing Systems*, pp. 617–624. MIT Press.

Jiang JA, Chuang CL, Wang YC, Hung CH, Wang JY, Lee CH, Hsiao YT (2011). "A Hybrid Framework for Fault Detection, Classification, and Location, Part I: Concept, Structure, and Methodology." *IEEE Transactions on Power Delivery*, **26**(3), 1988–1998. doi:10.1109/TPWRD.2011.2141157.

Kaci A, Kamwa I, Dessaint L, Guillon S (2014). "Synchrophasor Data Baselining and Mining for Online Monitoring of Dynamic Security Limits." *IEEE Transactions on Power Systems*, **29**(6), 2681–2695. doi:10.1109/TPWRS.2014.2312418.

Kim D, Chun TY, Yoon S, Lee G, Shin Y (2017). "Wavelet-Based Event Detection Method Using PMU Data." *IEEE Transactions on Smart Grid*, **8**(3), 1154–1162. doi:10.1109/TSG.2015.2478421.

Lévy-Leduc C, Roueff F (2009). "Detection and Localization of Change-Points In High-Dimensional Network Traffic Data." *The Annals of Applied Statistics*, **3**, 637–662. doi:10.1214/08-AOAS232.

Li W, Deka D, Chertkov M, Wang M (2019). "Real-time Faulted Line Localization and PMU Placement in Power Systems through Convolutional Neural Networks." doi:10.48550/arXiv.1810.05247.

Liao M, Chakrabortty A (2019). "Optimization Algorithms for Catching Data Manipulators in Power System Estimation Loops." *IEEE Transactions on Control Systems Technology*, **27**(3), 1203–1218. doi:10.1109/TCST.2018.2805294.

Mei Y (2005). "Information Bounds and Quickest Change Detection in Decentralized Decision Systems." *IEEE Transactions on Information Theory*, **51**, 2669–2681. doi:10.1109/TIT.2005.850159.

Moghaddass R, Wang J (2017). "A Hierarchical Framework for Smart Grid Anomaly Detection Using Large-Scale Smart Meter Data." *IEEE Transactions on Smart Grid*, **9**(6), 5820–5830. doi:10.1109/TSG.2017.2697440.

Nguyen D, Barella R, Wallace SA, Zhao X, Liang X (2015). "Smart Grid Line Event Classification Using Supervised Learning Over Pmu Data Streams." In *2015 Sixth International Green and Sustainable Computing Conference (IGSC)*, pp. 1–8. doi:10.1109/IGCC.2015.7393695.

Pan S, Morris T, Adhikari U (2015). "Classification of Disturbances and Cyber-Attacks in Power Systems Using Heterogeneous Time-Synchronized Data." *IEEE Transactions on Industrial Informatics*, **11**(3), 650–662. doi:10.1109/TII.2015.2420951.

Pandey S, Srivastava A, Amidan B (2020). "A Real Time Event Detection, Classification and Localization using Synchrophasor Data." *IEEE Transactions on Power Systems*, **35**, 4421–4431. doi:10.1109/TPWRS.2020.2986019.

Paschalidis IC, Smaragdakis G (2009). "Spatio–Temporal Network Anomaly Detection By Assessing Deviations of Empirical Measures." *IEEE/ACM Transactions on Networking*, **17**, 685–697. doi:10.1109/TNET.2008.2001468.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rafferty M, Liu X, Laverty DM, McLoone S (2016). "Real-Time Multiple Event Detection and Classification Using Moving Window PCA." *IEEE Transactions on Smart Grid*, **7**(5), 2537–2548. doi:10.1109/TSG.2016.2559444.

Raghavan V, Veeravalli V (2010). "Quickest Change Detection of a Markov Process Across a Sensor Array." *IEEE transactions on information theory*, **56**, 1961–1981. doi:10.1109/TIT.2010.2040869.

Rassam MA, Zainala A, Maarof M (2013). "An Efficient Distributed Anomaly Detection Model for Wireless Sensor Networks." In *Proc. Conf. Parallel and Distributed Computing and Systems (AASRI Procedia)*, pp. 9–14. doi:10.1016/j.aasri.2013.10.052.

Rimkus M, Kokoszka P, Prabakar K, Wang H (2022). "Toward Statistical Real-Time Power Fault Detection." *Technical report*, Colorado State University.

Tripathi S, De S (2018). "Dynamic Prediction of Powerline Frequency for Wide Area Monitoring and Control." *IEEE Transactions on Industrial Informatics*, **14**(7), 2837–2846. doi:10.1109/TII.2017.2777148.

Trudnowski D, Kosterev D, Undrill J (2013). "PDCI Damping Control Analysis for the Western North American Power System." In *Proceedings of 2013 IEEE Power & Energy Society General Meeting*, pp. 1–5. Vancouver, BC. doi:10.1109/PESMG.2013.6672718.

Vaughan J, Stoev S, Michailidis G (2013). "Network–Wide Statistical Modeling, Prediction and Monitoring Of Computer Traffic." *Technometrics*, **55**, 79–93. doi:10.1080/00401706.2012.723959.

Xie L, Chen Y, Kumar P (2014). "Dimensionality Reduction of Synchrophasor Data for Early Event Detection: Linearized Analysis." *IEEE Transactions on Power Systems*, **29**(6), 2784–2794. doi:10.1109/TPWRS.2014.2316476.

Xie Y (2012). *Statistical Signal Detection with Multi-Sensor and Sparsity*. Ph.D. thesis, Stanford University. doi:10.1.1.684.6036.

Xie Y, Siegmund D (2013). "Sequential Multisensor Change-Point Detection." *Annals of Statistics*, **41**, 670–692. doi:10.1214/13-AOS1094.

Yadav R, Pradhan AK, Kamwa I (2019). "Real-Time Multiple Event Detection and Classification in Power System Using Signal Energy Transformations." *IEEE Transactions on Industrial Informatics*, **15**(3), 1521–1531. doi:10.1109/TII.2018.2855428.

Yin S, Li X, Gao H, Kaynak O (2014). "Data-Based Techniques Focused on Modern Industry: An Overview." *IEEE Transactions on Industrial Electronics*, **62**(1), 657–667. doi:10.1109/TIE.2014.2308133.

Zhang N, Siegmund D, Ji H, Li J (2010). "Detecting Simultaneous Changepoints In Multiple Sequences." *Biometrika*, **97**, 631–645. doi:10.1093/biomet/asq025.

Zhou M, Wang Y, Srivastava AK, Wu Y, Banerjee P (2019). "Ensemble-Based Algorithm for Synchrophasor Data Anomaly Detection." *IEEE Transactions on Smart Grid*, **10**(3), 2979–2988. doi:10.1109/TSG.2018.2816027.

Affiliation:

Piotr Kokoszka
Statistics Department
Colorado State University
1877 Campus Delivery, Fort Collins, Colorado 80523-1877
E-mail: Piotr.Kokoszka@colostate.edu
URL: https://www.stat.colostate.edu/~piotr/

Austrian Journal of Statistics http://www.ajs.or.at/
published by the Austrian Society of Statistics http://www.osg.or.at/
Volume 52 *Submitted:* 2022-03-25
July 2023 *Accepted:* 2022-08-08