IET Smart Grid

Special Section: Privacy and Security in Smart Grids

IET Journals
The Institution of Engineering and Technology

Survey of machine learning methods for detecting false data injection attacks in power systems

eISSN 2515-2947
Received on 16th January 2020
Revised 22nd June 2020
Accepted on 17th August 2020
E-First on 6th October 2020
doi: 10.1049/iet-stg.2020.0015
www.ietdl.org

Ali Sayghe[1], Yaodan Hu[2], Ioannis Zografopoulos[1], XiaoRui Liu[1], Raj Gautam Dutta[2], Yier Jin[2], Charalambos Konstantinou[1] ✉

[1]*FAMU-FSU College of Engineering, Center for Advanced Power Systems, Florida State University, Tallahassee, FL, USA*
[2]*Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA*
✉ *E-mail: ckonstantinou@ieee.org*

Abstract: Over the last decade, the number of cyber attacks targeting power systems and causing physical and economic damages has increased rapidly. Among them, false data injection attacks (FDIAs) are a class of cyber-attacks against power grid monitoring systems. Adversaries can successfully perform FDIAs to manipulate the power system state estimation (SE) by compromising sensors or modifying system data. SE is an essential process performed by the energy management system towards estimating unknown state variables based on system redundant measurements and network topology. SE routines include bad data detection algorithms to eliminate errors from the acquired measurements, e.g. in case of sensor failures. FDIAs can bypass BDD modules to inject malicious data vectors into a subset of measurements without being detected, and thus manipulate the results of the SE process. To overcome the limitations of traditional residual-based BDD approaches, data-driven solutions based on machine learning algorithms have been widely adopted for detecting malicious manipulation of sensor data due to their fast execution times and accurate results. This study provides a comprehensive review of the most up-to-date machine learning methods for detecting FDIAs against power system SE algorithms.

Nomenclature

m	number of measurements
n	number of state variables
$\boldsymbol{H}$	$m \times n$ Jacobian matrix representing the topology
$\boldsymbol{x}$	$n \times 1$ vector of the state variable
$\boldsymbol{z}$	$m \times 1$ vector of measurements
$\boldsymbol{Z}$	$m \times n$ measurements matrix
$\boldsymbol{e}$	$m \times 1$ vector of measurement errors, s.t. $\boldsymbol{z} = \boldsymbol{Hx} + \boldsymbol{e}$
$\hat{\boldsymbol{x}}$	$n \times 1$ vector of estimated state variables
$\boldsymbol{W}$	$m \times m$ diagonal matrix, s.t. $w_i, i = \sigma_i^{-2}$, where σ_i^2 is the variance of the ith measurement ($1 \leq i \leq m$)
τ	threshold for L_2-norm-based bad data detection
$\boldsymbol{z}_a$	$m \times 1$ measurement vector with bad measurement
$\boldsymbol{a}$	$m \times 1$ attack vector, s.t., $\boldsymbol{z}_a = \boldsymbol{z} + \boldsymbol{a}$
$\boldsymbol{c}$	$n \times 1$ vector of estimation errors s.t. $\boldsymbol{a} = \boldsymbol{Hc}$
V_i, θ_i	voltage magnitude and phase angle at bus i
g_{ij}, b_{ij}	real and imaginary parts of the admittance of the series branch between bus i and bus j

1 Introduction

The first practical power system, developed by Westinghouse Electric company in 1886, changed the landscape of human society [1]. Recently, the integration of information and communication technology into power grid applications has enabled the evolution towards a smart grid architecture. Smart grids, among others, improve the monitoring capabilities of power systems leveraging advanced microprocessor-based components such as phasor measurement units (PMUs) and smart meters. Grid operators can impose controls on electricity generation and consumption, increasing the efficiency and reliability of power systems by utilising the measurements from these components. At the same time, the inclusion of smart sensing and control devices expanded the attack landscape [2]. The increasing network interfaces of smart grid implementations provide entry points for cyber-intruders [3]. In December 2015, a cyberattack on the Ukrainian power grid led to a power outage affecting more than 200,000 customers [4]. One year later, a similar but more complex attack was carried out again in Ukraine [5]. These attack incidents confirm that the vulnerabilities within grid devices and networks could be maliciously exploited (even remotely) with large-scale impacts on the system [6, 7].

It is critical to detect cyber-attacks promptly to increase the security and reliability of the power system. This paper focuses on false data injection attacks (FDIAs), a type of cyberattack that injects false measurements to poison the state estimation (SE) process [8]. Traditional bad data detection (BDD) methods are based on the residuals between the observed and estimated measurements [9–11]; if the residual is larger than a threshold, bad data is suspected to exist. Despite the wide adoption of such methods, it has been demonstrated that FDIAs can bypass BDD algorithms. The concept of FDIAs in power systems was introduced in 2009 [12]. Different techniques have been proposed since then to detect FDIAs including the Kullback Leibler distance method, fast go-decomposition, unscented Kalman filter (UKF), Bayesian formulation, Bayesian framework, generalised likelihood ratio, Markov chains, cosine similarity matching scheme, and diagnostic robust generalised potential [13–26]. However, such techniques often fail to detect FDIAs that fit the same distribution of historical measurements and can only capture attacks that cause abnormal system states [26]. For example, the Kullback Leibler distance method fails to detect FDIAs in system buses where the attacker injects a small measurement error into a specific state. Also, the Bayesian framework and generalised likelihood ratio methods cannot detect FDIAs if the attacker replaces the current meter readings with historical readings that have the same distribution. To address this issue, Majumdar and Pal [16] proposed a technique called diagnostic robust generalised potential. First, the system measurements are separated in leverage and non-leverage sets, and then by employing the diagnostic robust generalised potential method, bad data can be efficiently identified performing residual analysis, even if FDIAs exist in the form of gross errors. However, it is well known that identifying bad leverage points is challenging for such largest normalised residual (LNR) statistical tests [27].

```
Input: Parameters, Measurements, Pseudomeasurements, τ
Output: StateEstimates
while PS.on do
    Topology = PS.build(Parameters, Measurements)
    ObservabilityMatrix = PS.map(Topology,
      Pseudomeasurements)
    while error ≥ τ do
        [error, StateEstimates] =
          PS.stateEstimation(Topology, ObservabilityMatrix)
    end
    return StateEstimates
end
```

Fig. 1 *Algorithm 1: overview of the SE process*

Machine learning algorithms have been widely applied in power grid functions for control and monitoring purposes [28–30]. For example, Zhang *et al.* [28] implemented analysing modules leveraging machine learning algorithms at different levels of the grid network for intrusion detection. Anderson *et al.* [29] proposed a machine-learning algorithm to manage the system loads and sources. Rudin *et al.* [30] suggested using machine learning algorithms to anticipate component failures in power systems. To overcome FDIAs detection limitations, researchers have also developed techniques leveraging machine learning algorithms to efficiently detect such attacks [13, 15, 31–33]. Various types of algorithms have been investigated in the literature including supervised, semi-supervised, unsupervised, and deep learning. Such methods demonstrate better performance in terms of accuracy and adaptability to dynamic and uncertain grid environments [31–34].

In this work, we present a survey of FDIAs detection methods based on machine learning algorithms. The contributions of this paper are as follows:

- We present a comprehensive overview of FDIAs in the power grid including background information for SE, different FDIAs' settings, impacts of FDIAs on power systems, and FDIAs defence methods.
- We provide a survey of FDIAs detection methods based on the machine learning algorithms and describe and their limitations.
- Based on the limitations of the surveyed papers, we identify further research problems to be addressed. By providing such a discussion, we aim to shed light on future directions that utilise machine learning algorithms for FDIAs detection.

The rest of the paper is organised as follows: Section 2 provides the background on power system SE, BDD methods, and FDIAs. Section 3 provides details on different FDIA formulations and their impact on power systems. In Section 4, we present traditional defence strategies against FDIAs. We survey different machine learning methods for attack detection in Section 5. Section 6 discusses the performance of machine learning algorithms in the context of SE while conclusions are presented in Section 7. Common notations used in the paper are listed in Nomenclature.

2 Background

2.1 Power system SE

SE enables system operators (SOs) to optimally manage, plan, and control the power grid. SE is used to assess the system's state, check for anomalous behaviour, and indicate if mitigation strategies are necessary to preserve nominal operation. Depending on the power system level that SE is applied, i.e. transmission level or distribution level, different algorithms, assumptions, and approximations are employed. The differences between the transmission system (TS) and the distribution system (DS) in terms of SE algorithms are discussed in Sections 2.2 and 2.3. Multiple SE algorithms have been proposed aiming to optimise the computational intensive estimation process and enable its real-time calculation [35–45]. Despite the plurality of SE methodologies and their application level, the core components of these analyses are fundamentally similar. An outline of the SE process is presented in Algorithm 1 (see Fig. 1).

2.2 TS modeling for SE

In this case, it is typically assumed that the system is balanced, over determined, i.e. the number of available measurements is more than the number of unknown state variables, and that the system nodes are connected in a mesh topology. These assumptions simplify the analysis, contrary to DSs, which are radially connected, unbalanced, and insufficient measurement points are available (Section 2.3). The inputs to the SE are (i) the power system parameters (e.g. lines, buses, branches, breaker states etc.), (ii) the collected measurements (e.g. voltages, angles, real/reactive power injections, and flows), and (iii) pseudo-measurements (e.g. load forecasts, historical data etc.), which are utilised when insufficient system information is available.

The TS model is composed of a set of buses $\mathcal{N} = \{1, \ldots, n\}$, where $n = |\mathcal{N}|$ is the total number of buses. Furthermore, the states of the system at each bus include the voltage magnitude and the phase angle. We denoted the system states using $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$. Depending on the fidelity of the model, the system measurements can include active and reactive power injections, active and reactive power flows, voltage magnitudes, voltage angles, current magnitudes etc. [9]. Finally, the set of measurements is denoted as $\boldsymbol{z} = (z_1, z_2, \ldots, z_m)^{\mathrm{T}}$, where m is the number of measurements.

The SE inputs are used to build an accurate TS topology and the observability matrix. By inspecting the observability matrix, we can determine which system states are unobservable and derive approximations using the redundancy of the over determined system measurements as well as the pseudo-measurements. The calculated results are passed to the main SE routine, which iterates until an optimal system solution (based on the imposed constraints) is reached. Solving the SE problem can be a time- and resource-consuming procedure. Additionally, SE is sensitive to measurement errors, which can also impact the algorithm's convergence efficacy. Following, we present the AC SE methodology and demonstrate it as a non-linear optimisation problem. Additionally, in Section 2.2.2, we present how, in favour of real-time performance and by partially sacrificing the model's accuracy, we derive a linear (DC) model for the SE problem.

2.2.1 AC state estimation: The AC SE leverages phase angles and voltage magnitudes to construct the system states. Typically, the phase angle at the slack bus is set as the reference, i.e. $\theta_1 = 0$, thus it is not included in the system state vector $\boldsymbol{x}$. With this assumption, we define the power system states as

$$\boldsymbol{x} = (\theta_2, \theta_3, \ldots, \theta_n, V_1, V_2, \ldots, V_n)^{\mathrm{T}} \tag{1}$$

The measurements, $\boldsymbol{z}$, include bus voltages and angles, as well as, the real and reactive power flows and injections. For each bus $i \in \mathcal{N}$, as depicted in Fig. 2, we have

$$P_i = V_i \sum_{j=1}^{n} V_j (g_{ij} \cos\theta_{ij} + b_{ij} \sin\theta_{ij}) \tag{2}$$

$$Q_i = V_i \sum_{j=1}^{n} V_j (g_{ij} \sin\theta_{ij} - b_{ij} \cos\theta_{ij}) \tag{3}$$

where P_i and Q_i are the real and reactive power injections at bus i, respectively. g_{ij} and b_{ij} are the real and imaginary part of the nodal admittance matrix element $\boldsymbol{Y}_{ij}$, and $\theta_{ij} = \theta_i - \theta_j$ is the phase angle difference between buses i and j.

We utilise a non-linear function vector $\boldsymbol{h}(\boldsymbol{x}) = (h_1, h_2, \ldots, h_m)^{\mathrm{T}}$ to represent the relationship as presented in (2) and (3). Thus, we obtain the observation model $\boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x}) + \boldsymbol{e}$, where $\boldsymbol{e} = (e_1, e_2, \ldots, e_m)^{\mathrm{T}}$ is the vector of measurement errors [46]. These measurement errors are assumed to be independent of each

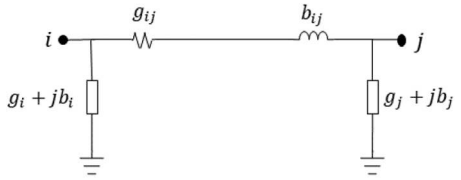

Fig. 2 *Transmission network element between bus i and bus j*

other and follow the Gaussian distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{W})$, in which $\boldsymbol{W}$ is the covariance matrix of the measurement errors

$$\boldsymbol{W} = \text{diag}\left\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_{2n}^2\right\} \tag{4}$$

The weighted least square (WLS) technique is one of the most commonly used methods for SE [9]. In WLS, the estimates are obtained by minimising the sum of the residual squares as illustrated in (5)

$$\begin{aligned}\hat{\boldsymbol{x}} &= \arg\ \min_x \boldsymbol{J}(\boldsymbol{x}) \\ &= \arg\ \min_x (\boldsymbol{z} - \boldsymbol{h}(\boldsymbol{x}))^{\mathrm{T}} \boldsymbol{W}^{-1} (\boldsymbol{z} - \boldsymbol{h}(\boldsymbol{x}))\end{aligned} \tag{5}$$

The optimisation problem presented in (5) can be solved using the iterative normal equation method [44]. At any given point, the solution should satisfy the first-order optimal condition of (6)

$$\boldsymbol{G}_1(\hat{\boldsymbol{x}}) = \frac{\partial J(x)}{\partial x}\bigg|_{x=\hat{x}} = -\boldsymbol{H}^{\mathrm{T}}(\hat{\boldsymbol{x}})\boldsymbol{W}^{-1}\left[\boldsymbol{z} - \boldsymbol{h}(\hat{\boldsymbol{x}})\right] = 0 \tag{6}$$

where $\boldsymbol{H}(\boldsymbol{x}) = \partial h(x)/\partial x$ is the Jacobian matrix (7) derived from the function vector $\boldsymbol{h}(\boldsymbol{x})$, and $\hat{\boldsymbol{x}}$ is the estimated state vector

$$\boldsymbol{H}(\boldsymbol{x}) = \frac{\partial h(x)}{\partial (x)} = \begin{bmatrix} \frac{\partial h_1(x)}{\partial x_1} & \frac{\partial h_1(x)}{\partial x_2} & \cdots & \frac{\partial h_1(x)}{\partial x_n} \\ \frac{\partial h_2(x)}{\partial x_1} & \frac{\partial h_2(x)}{\partial x_2} & \cdots & \frac{\partial h_2(x)}{\partial x_n} \\ \vdots & \vdots & \ddots & \\ \frac{\partial h_m(x)}{\partial x_1} & \frac{\partial h_m(x)}{\partial x_2} & \cdots & \frac{\partial h_m(x)}{\partial x_n} \end{bmatrix} \tag{7}$$

Equation (6) can be iteratively solved using the Newton–Raphson method, and $\hat{\boldsymbol{x}}$ can be approximated with

$$\hat{\boldsymbol{x}}_{v+1} = \boldsymbol{x}_v + ((\boldsymbol{G}_2^{\mathrm{T}}\boldsymbol{G}_2)^{-1}\boldsymbol{G}_2^{\mathrm{T}}\boldsymbol{G}_1)\big|_{x=x_v} \tag{8}$$

where $\boldsymbol{G}_2 = \partial^2 J/\partial x^2$ is the Hessian matrix of $\boldsymbol{J}(\boldsymbol{x})$ and $v \in \mathcal{N}$ is the iteration step.

Alternative methods, such as the maximum likelihood estimation (MLE) can be employed to solve the optimisation problem of (5) [46]. Additionally, orthogonal methods can be utilised to solve the first optimal condition introduced of (6) [44]. However, the rate and convergence accuracy of these heuristic methodologies rely solely on the system observability matrix characteristics (i.e. rank).

2.2.2 DC state estimation: To alleviate the computational burden introduced by non-linear AC SE, the DC SE model (a linear measurement model) is often considered at the sacrifice of accuracy. The DC SE assumes that the line resistance is negligible compared to the corresponding line reactance, and the phase angle difference between neighbouring nodes is small (i.e. zero degrees angle difference). Also, the voltage magnitudes are assumed to be 1 p.u. Thus, dissimilar to the AC SE, in DC SE, the system states are composed only from the phase angles $\boldsymbol{x} = (\theta_2, \theta_3, \ldots, \theta_n)^{\mathrm{T}}$. Moreover, since the reactive power flow between buses is negligible and the reactive power injections at every bus depend on the line susceptance, only active power flows and injections are utilised in DC SE

$$P_i = \sum_{j=1, j\neq i}^{n} b_{ij}(\theta_i - \theta j) \tag{9}$$

Thus, the observation model in DC SE can be formalised as

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e} \tag{10}$$

where $\boldsymbol{H}_{ii} = \sum_{j=1, j\neq i}^{j=n} b_{ij}$ and $\boldsymbol{H}_{ij} = -b_{ij}$. $\boldsymbol{z} = (P_1, \ldots, P_n)^{\mathrm{T}}$ and $\boldsymbol{e} = (e_1, \ldots, e_n)^{\mathrm{T}}$ follow the same assumptions as in the AC SE. Leveraging WLS to solve (10), we obtain the following objective function formulation:

$$\begin{aligned}\hat{\boldsymbol{x}} &= \arg\ \min_x J(\boldsymbol{x}) \\ &= \arg\ \min_x (\boldsymbol{z} - \boldsymbol{H}\boldsymbol{x})^{\mathrm{T}} \boldsymbol{W}^{-1} (\boldsymbol{z} - \boldsymbol{H}\boldsymbol{x})\end{aligned} \tag{11}$$

The solution satisfies the following requirement:

$$\boldsymbol{G}_1(\hat{\boldsymbol{x}}) = \frac{\partial J(x)}{\partial x}\bigg|_{x=\hat{x}} = -\boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}^{-1}\left[\boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}}\right] = 0 \tag{12}$$

which can be simplified to

$$\hat{\boldsymbol{x}} = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{z} \tag{13}$$

2.2.3 Dynamic SE (DSE): SOs have extensively used both AC and DC static SE to monitor TS operation and manage energy generation. However, these SE algorithms rely on bus voltage and angle measurements as well as active and reactive power injections to calculate the system state estimates. The disadvantage of these methods is that the system state approximation – using either non-linear or linear models – depends on low-update rate steady-state measurements, e.g. supervisory control and data acquisition (SCADA) [36]. The current transmission infrastructure advancements with the integration of wind and solar generation require improved estimation algorithms able to capture the dynamic system behaviour [37]. To address the aforementioned AC and DC SE pitfalls, and account for the dynamic and intermittent nature of TS with renewable penetrations, DSE algorithms have been proposed.

The initial implementations of DSE algorithms although they could acutely reflect the system's transient behaviour, they still suffered from the disadvantages of the traditional SE methodologies [39, 47, 48]. For example, to achieve faster convergence rates, non-linear models had to be linearised causing significant approximation errors and Jacobian system matrices had to be recalculated at every iteration step yielding excessive computational overheads [39]. To overcome linearisation errors and computational intensive matrix operations, recent works have opted for improved SE methodologies leveraging Kalman filtering techniques. In [40], an UKF method for DSE is introduced, which overcomes the aforementioned drawbacks and avoids the high-order derivative calculations in favour of real-time performance. Other works have incorporated high data-rate sampling measurements from PMUs to increase the robustness of their estimations [37]. For instance, the authors in [36] showed improvements in estimation accuracy, algorithm convergence, and minimised the estimation complexity. Their algorithm allows leveraging UKF, PMU measurements, as well as a decentralised SE approach, demonstrating a practical implementation for TS DSE.

2.3 DS modelling for SE

In the past, SE was applied exclusively on the transmission level since the distribution level can be simplified to a lumped passive load structure. However, with the deployment of distributed generation, distributed energy resources (DERs), microgrids, electric vehicles, and energy storage systems, the development of comprehensive algorithms that account for bidirectional power flow between transmission and distribution levels is imperative.

The first DS state estimators (DSSEs) are adaptations of the corresponding TS counterparts [38]. However, DS architectures differ significantly from transmission networks.

DSs are radially connected and their interconnections typically present high resistance to reactance (R/X) ratios. On the other hand, TSs are connected in lattice-based formations, aiding redundancy, and their line resistances are negligible. Second, there are fewer measurement points in DSs when compared to TSs, and even when measurements are available, they are usually collected deficiently (every 15 min or even longer). Also, measurements might not be time synchronised and can be inaccurate (due to improper connections or not calibrated meters). Thus, relying on pseudo-measurements for DSSE is a common practice. Furthermore, DSs are constantly changing by integrating distributed generation units, loads, and prosumers, thus DSSE algorithms should be able to account for such characteristics. Finally, TSs are treated as perfectly balanced systems by SE algorithms; such algorithms cannot be applied for DS topologies since they present serious imbalances between phases and require three-phase modelling.

2.3.1 SE for unbalanced system operation: The dynamic behaviour of DSs, furnishing variable power penetrations, and demands at every system bus, generates load flow differences between phases. Thus, for practical DSs, solving the unbalanced three-phase problem is required to perform DSSE. For instance, in [45], the authors solve the DSSE problem utilising unbalanced single-phase and two-phase measurement models. Equations (14) and (15) demonstrate the active and reactive power flows for the three-phase system model at bus i and phase p. In the aforementioned equations, V is the voltage and θ_{ik}^{pm} present the phase angle difference between bus i with phase p and bus k with phase m. g_{ik}^{pm} and b_{ik}^{pm} are the corresponding real and imaginary parts of the admittance matrix representing the conductance and susceptance for each bus, respectively

$$P_i^p = |V_i^p| \sum_{k=1}^{N} \sum_{m \in \{a,b,c\}} |V_k^m| \left(g_{ik}^{pm} \cos\left(\theta_{ik}^{pm}\right) + b_{ik}^{pm} \sin\left(\theta_{ik}^{pm}\right)\right) \quad (14)$$

$$Q_i^p = |V_i^p| \sum_{k=1}^{N} \sum_{m \in \{a,b,c\}} |V_k^m| \left(g_{ik}^{pm} \sin\left(\theta_{ik}^{pm}\right) - b_{ik}^{pm} \cos\left(\theta_{ik}^{pm}\right)\right) \quad (15)$$

Additionally, the line-to-line voltage as well as the real power injection of the two-phase measurement model is demonstrated in (16) and (17). To calculate the mentioned bus voltages and power injections, (i) the three-phase power injection, the phase-to-neutral voltage magnitude, and the magnitude of current injection at the current substation, in addition to, (ii) the two-phase voltage magnitudes and power injections at every distribution centre-tapped transformer are necessary

$$|V_i^{pm}|_{\text{meas}} = \sqrt{|V_i^p|^2 + |V_i^m|^2 - 2|V_i^p||V_i^m| \cos(\theta_i^p - \theta_i^m)} + e_{|V_i^{pm}|} \quad (16)$$

$$P_{i\text{meas}}^{pm} = |V_i^p| \sum_{k=1}^{N} \sum_{n \in \{a,b,c\}} |V_k^n| \left(g_{ik}^{pn} \cos\left(\theta_{ik}^{pn}\right) + b_{ik}^{pn} \sin\left(\theta_{ik}^{pn}\right)\right) - |V_i^m| \sum_{k=1}^{N} \sum_{n \in \{a,b,c\}} |V_k^n| \left(g_{ik}^{pn} \cos\left(\theta_{ik}^{mn}\right) + b_{ik}^{pn} \sin\left(\theta_{ik}^{mn}\right)\right) + e_{P_i^{pm}} \quad (17)$$

Furthermore, when the phase-to-neutral voltage magnitudes and the real power injection measurements are available – assuming ideal centre-tapped and single-phase transformers (i.e. the transformer losses are negligible) – we can acquire the following single-phase measurement equations:

$$|V_i^p|_{\text{meas}} = |V_i^p| + e_{|V_i^p|} \quad (18)$$

$$P_{i\text{meas}}^p = P_i^p + e_{P_i^p} \quad (19)$$

$$Q_{i\text{meas}}^p = Q_i^p + e_{Q_i^p} \quad (20)$$

Performing the Kron reduction method on the initial four-wire matrix, which also includes the line-to-neutral impedances, the simplified (row and column reduced) line impedance can be obtained by using the resistance (R) and reactance (X) of the line. A three-phase (a, b, and c) line impedance matrix between bus i and bus j can be calculated by utilising (21)

$$\boldsymbol{Z}_{\text{Imp}(abc,ij)} = \boldsymbol{R}_{abc,ij} + \text{j}\boldsymbol{X}_{abc,ij} = \begin{bmatrix} \boldsymbol{Z}_{\text{Imp}(aa,ij)}^n, & \boldsymbol{Z}_{\text{Imp}(ab,ij)}^n, & \boldsymbol{Z}_{\text{Imp}(ac,ij)}^n \\ \boldsymbol{Z}_{\text{Imp}(ba,ij)}^n, & \boldsymbol{Z}_{\text{Imp}(bb,ij)}^n, & \boldsymbol{Z}_{\text{Imp}(bc,ij)}^n \\ \boldsymbol{Z}_{\text{Imp}(ca,ij)}^n, & \boldsymbol{Z}_{\text{Imp}(cb,ij)}^n, & \boldsymbol{Z}_{\text{Imp}(cc,ij)}^n \end{bmatrix} \quad (21)$$

This methodology can be applied irrespective of the system modelling being single-phase, two-phase, or three-phase. For example, if we opt for a single-phase model, the corresponding row and column of the other two phases will be zero.

Furthermore, the branch voltages and branch currents modelling is shown in (22) and (23), respectively. The mentioned branch voltage and current modeling allows for direct use in voltage-based or branch current-based SE methods [42]

$$\boldsymbol{V}_{abc,ij} = \begin{bmatrix} V_{ai} \\ V_{bi} \\ V_{ci} \end{bmatrix} - \begin{bmatrix} V_{aj} \\ V_{bj} \\ V_{cj} \end{bmatrix} \quad (22)$$

$$\boldsymbol{I}_{abc,ij} = \begin{bmatrix} I_{a,ij} \\ I_{b,ij} \\ I_{c,ij} \end{bmatrix} \quad (23)$$

Other methods leverage WLS to construct a linear SE model for unbalanced three-phase systems [41]. For this linear approximation, the bus voltages and branch currents as well as the active and reactive power flow measurements are essential for the three-phase unbalanced system model. Furthermore, the SE algorithm requires timely synchronised phasor measurements. We demonstrate the measurement vector, state vector, and the process matrix $\boldsymbol{H}$ in (24)–(26), where the subscripts r and i are the real and imaginary values. The system residuals for unbalanced operation are formulated in (27). All the aforementioned differences in system modelling make DSSE an arduous and computational intensive process limiting its real-time applicability

$$z = z_{\text{r}} + \text{j}z_{\text{i}} \quad (24)$$

$$\boldsymbol{x} = \boldsymbol{x}_{\text{r}} + \text{j}\boldsymbol{x}_j \quad (25)$$

$$\boldsymbol{H} = \boldsymbol{H}_{\text{r}} + \text{j}\boldsymbol{H}_j \quad (26)$$

$$\boldsymbol{r}_{\text{r}} + \text{j}\boldsymbol{r}_{\text{i}} = z_{\text{r}} + \text{j}z_{\text{i}} - (\boldsymbol{H}_{\text{r}} + \text{j}\boldsymbol{H}_{\text{i}})(\boldsymbol{x}_{\text{r}} + \text{j}\boldsymbol{x}_{\text{i}}) \quad (27)$$

2.4 BDD and identification

With bad data injected during the SE, the states might not be accurate, which could lead to wrong decision making and economic losses. Therefore, it is necessary to sanitise the measurements by removing the bad data. Some bad data such as negative voltage magnitudes can be easily removed before the SE process. However, other types of bad data require sophisticated methods in order to be detected, identified, and removed from the true measurement vectors.

2.4.1 Bad data detection: The goal of BDD is to determine whether bad data exist in the measurement vectors [49, 50]. The chi-square test is a statistical method widely used for this process.

Chi-square assumes that the distribution of measurements follows a Gaussian distribution. Thus, the test statistic $\boldsymbol{J}(\hat{\boldsymbol{x}})$ (calculated in (28)) follows chi-square distribution when there exist no bad data [51]:

$$\boldsymbol{J}(\hat{\boldsymbol{x}}) = \sum_{i=1}^{m} \frac{\left(z_i - \boldsymbol{h}_i(\hat{\boldsymbol{x}})\right)^2}{\sigma_i^2} \tag{28}$$

where $\boldsymbol{r}_i = z_i - \boldsymbol{h}_i(\hat{\boldsymbol{x}})$ is known as the residual. If $\boldsymbol{J}(\hat{\boldsymbol{x}})$ is larger than a predetermined threshold, then bad data exist in the measurements.

2.4.2 Bad data identification: The goal of the bad data identification procedure is to determine which set of measurements contains bad data [52, 53]. LNR is one of the most commonly used methods for bad data identification [54]. Similar to the chi-square test method, LNR assumes that the bad measurements have large residuals. The following steps detail the process of identifying bad data using LNR:

i. Calculate the gain matrix $\boldsymbol{G}$ and covariance matrix $\boldsymbol{\Omega}$

$$\boldsymbol{G} = \boldsymbol{H}^{\mathrm{T}} \cdot \boldsymbol{W} \cdot \boldsymbol{H} \tag{29}$$

$$\boldsymbol{\Omega}(\hat{\boldsymbol{x}}) = \boldsymbol{W} - \boldsymbol{H}(\hat{\boldsymbol{x}}) \cdot \boldsymbol{G}^{-1} \cdot \boldsymbol{H}^{T}(\hat{\boldsymbol{x}}) \tag{30}$$

i. Calculate the normalised residuals after solving the estimation problem using the WLS method

$$\boldsymbol{r}_i^n = \frac{|\boldsymbol{r}_i|}{\sqrt{\boldsymbol{\Omega}_i}} \quad i = 1, 2, \ldots, m \tag{31}$$

ii. Find the maximum value $r_{\max}^n$ of r_i^n for $i = 1, 2, \ldots, m$.

iii. Compare the LNR with a pre-determined threshold τ. If $\| r_{\max}^n \| > \tau$, the corresponding measurement is assumed to be bad (modified).

iv. Remove the suspected bad measurement from the measurement set and go to step one.

Although residual-based methods are widely used, it has been demonstrated that they cannot efficiently detect FDIAs [55].

2.5 FDIAs in power systems

FDIAs are a class of cyber-attacks, which can bypass BDD mechanisms, and aim to compromises the data integrity of power system measurements. SOs utilise SE on both the transmission and the distribution level. SE results serve as inputs to other crucial power system services (e.g. optimal power flow, economic dispatch, demand–response, contingency analysis etc.), thus their validity is of paramount importance. Ensuring accurate results requires meticulous line interconnection and topology modelling as well as scalable and dynamic algorithms. Furthermore, efficient SE algorithms that can harness pseudo-measurements (based on historical data or forecasts) and comply with the real-time system operational requirements are crucial. Attackers, either intrusively (e.g. having physical access to a grid asset, which reports measurements) or non-intrusively (e.g. by spoofing a communication channel over which power system measurements are propagated) can maliciously modify and inject false data in the system. Typically, a FDIA is formulated as follows:

$$z_a = z + \boldsymbol{a} \tag{32}$$

where z_a is the tampered measurement vector, z is the true measurement vector, and $\boldsymbol{a}$ is a non-zero attack vector added to the true measurements.

To bypass BDD (i.e. not affect residuals), the attack vector $\boldsymbol{a}$ is constructed as a linear combination of the column vectors of the Jacobian $\boldsymbol{H}$ matrix, that is, $\boldsymbol{a} = \boldsymbol{H}\boldsymbol{c}$, where $\boldsymbol{c}$ is an arbitrary $n \times 1$ non-zero vector. The attack vector is constructed as follows:

$$\begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ a_m \end{bmatrix}_{m \times 1} = c_1 \begin{bmatrix} h_{11} \\ h_{21} \\ \cdot \\ h_{m1} \end{bmatrix} + \cdots + c_n \begin{bmatrix} h_{1n} \\ h_{2n} \\ \cdot \\ h_{mn} \end{bmatrix} \tag{33}$$

$$z_a = \boldsymbol{H}(\boldsymbol{x} + \boldsymbol{c}) \tag{34}$$

and the new estimated state $\hat{\boldsymbol{x}}_a$ is equal to

$$\hat{\boldsymbol{x}}_a = \hat{\boldsymbol{x}} + \boldsymbol{c} \tag{35}$$

The value of $\boldsymbol{c}$ should not exceed the maximum alterable tolerance of any measurement to avoid triggering alarms and draw the grid operator's attention [56]. Following this procedure, z_a produces the same residual as the real measurement vector z, and thus bypasses the residual-based BDD (for the DC SE model)

$$\begin{aligned} \boldsymbol{r}_a &= \| z_a - \boldsymbol{H}\hat{\boldsymbol{x}}_a \| \\ &= \| z + \boldsymbol{a} - \boldsymbol{H}(\hat{\boldsymbol{x}} + \boldsymbol{c}) \| \\ &= \| z + \boldsymbol{a} - \boldsymbol{H}\hat{\boldsymbol{x}} - \boldsymbol{H}\boldsymbol{c} \| \\ &= \| z - \boldsymbol{H}\hat{\boldsymbol{x}} + (\boldsymbol{a} - \boldsymbol{H}\boldsymbol{c}) \| \\ &= \| z - \boldsymbol{H}\hat{\boldsymbol{x}} \| \ = \boldsymbol{r} \end{aligned} \tag{36}$$

In (36), we prove that if $\boldsymbol{a} = \boldsymbol{H}\boldsymbol{c}$, then $\boldsymbol{r}_a = \boldsymbol{r}$, indicating that the attack succeeds without changing the measurement residual or triggering the BDD. FDIA formulation for the AC SE is similar to the DC SE case. The attack bypasses the BDD if $\boldsymbol{a} = \boldsymbol{h}(\hat{\boldsymbol{x}}_a) - \boldsymbol{h}(\hat{\boldsymbol{x}})$, and therefore the residual remains unaltered (37)

$$\begin{aligned} \boldsymbol{r}_a &= \| z_a - \boldsymbol{h}(\hat{\boldsymbol{x}}_a) \| \\ &= \| z + \boldsymbol{a} - \boldsymbol{h}(\hat{\boldsymbol{x}}_a) + \boldsymbol{h}(\hat{\boldsymbol{x}}) - \boldsymbol{h}(\hat{\boldsymbol{x}}) \| \\ &= \| z - \boldsymbol{h}(\hat{\boldsymbol{x}}) + \boldsymbol{a} - \boldsymbol{h}(\hat{\boldsymbol{x}}_a) + \boldsymbol{h}(\hat{\boldsymbol{x}}) \| \ = r \end{aligned} \tag{37}$$

Many researchers provide use cases where the TS SE can be maliciously manipulated if PMU data, remote terminal units' data, or SCADA measurements are compromised, as well as how the corresponding FDIAs can be constructed [12, 57–59]. Owing to the differences between TS and DS modelling and operation, the SE mechanisms can differ significantly as discussed in Sections 2.2 and 2.3. The heavily interconnected DS topology, the number of insufficient measurement points, dynamic and unbalanced DS operation complicate the DSSE process. Attackers can leverage the elaborate DSSE to mount FDIAs and avoid detection. Research works discussing FDIAs which target DS have been reported [60–62]. Detecting and mitigating FDIAs is a field of on-going research. An overview of the state-of-the-art methodologies leveraging machine learning is discussed in Section 5.

3 False data injection attack settings and impacts

In this section, we provide a brief overview of how FDIAs can be launched according to the attack knowledge settings (summarised in Table 1) and discuss their potential impacts on power systems (summarised in Table 2).

3.1 FDIAs settings

Typically, system knowledge includes meter measurement data, the Jacobian matrix or system topology, system parameters, and control commands (e.g. switch states). Moreover, to compromise the DS SE, the attacker should know the state estimates to successfully launch a FDIA. Based on the attacker's knowledge, attackers can be classified into two categories: (i) attackers with full system knowledge and (ii) attackers with incomplete or partial system knowledge. Full system knowledge enables the attacker to design FDIAs that will not trigger detection mechanisms. On the other hand, in the case of incomplete or partial information, the attackers may not know the exact system topology (e.g. Jacobian

matrix). Thus, attackers first need to approximate this crucial information (i.e. topology matrix) leveraging meter measurements or historical data, before a stealthy FDIA can be launched.

3.1.1 FDIAs with full system knowledge: The concept of FDIAs in the power grid, originally introduced by Liu *et al.* [12], investigated two different FDIAs scenarios: (i) attacks with limited access to meters and (ii) attacks with limited resources to compromise a large number of meters. In the first scenario, the attacker could only compromise k specific meters due to different security requirements of each meter. For the attack to have considerable impact, the authors assume $k \geq m - n + 1$, where m is the number of measurements and n is the number of states. In the second scenario, the authors assume that there are no protected meters, but the attacker has limited resources and could only compromise a limited number of meters. Owing to resource constraints, the attacker could not compromise more than k meters. In both scenarios, the authors prove that the attacker could systematically and efficiently construct attack vectors, which can modify the SE results without being detected. Both scenarios are experimentally demonstrated on IEEE 9, 27, and 300 bus test cases. The simulation results illustrate the significant impact of FDIAs (e.g. blackouts in large geographic areas).

Sou *et al.* [63] studied how the minimum set of meters – required to compromise the system – can be found. The authors assume that there are no empty measurements, i.e. all the rows of the observation model matrix $\boldsymbol{H}$ are non-zero. In their work, the attacker intends to spoof a specific measurement, e.g. the kth measurement. To avoid detection, the attacker also modifies other measurements according to (34). Thus, the attacker's objective is to minimise the number of compromised meters to reduce the attack cost and detection risk. The sparsest stealthy FDIAs problem formulation is the following:

$$\begin{aligned} \alpha_k = &\min : \| \boldsymbol{Hc} \|_0 \\ &\text{Subject to: } \boldsymbol{H}(k,:)c = 1 \end{aligned} \tag{38}$$

where α_k is defined as the security index of the kth measurement, i.e. the minimum number of measurements required to be compromised for a stealthy FDIA to spoof the kth measurement. Multiplying by a constant c, the attacker can tamper the kth measurement with any value. The security index of the measurements helps the SO to understand the data manipulation patterns and allocate protective resources effectively. To solve the optimisation problem of (38), various methods have been proposed, such as the mixed-integer linear programming (MILP) method and the matching pursuit methodology [63, 64].

3.1.2 FDIAs with partial or incomplete knowledge: In [65], the authors study FDIAs with incomplete transmission line admittance information, i.e. the attacker does not possess an accurate version of matrix $\boldsymbol{H}$. As a result, the attacker does not know the exact values of the transmission line admittance for any part of the power grid topology. However, the attacker could build probability distributions and infer the unknown line admittance with offline and online information. The offline information relies on historical measurements, while the online information is collected by deploying meters or PMUs in the system. The authors compare the impact and detection probability of such attacks against full knowledge FDIAs. The simulation results demonstrate that the attacker could still launch successful FDIAs even with incomplete system information.

Other researchers investigate data-driven approaches to build the Jacobian matrix $\boldsymbol{H}$ and launch FDIAs, referred to as blind FDIAs [67, 68]. In blind FDIAs, no additional knowledge (except system measurements) is required, and the attack is performed utilising the equivalent $\boldsymbol{H}$ matrix constructed in accordance with the acquired measurements. The measurements can be obtained either by direct access to the system or by spoofing the system for a short time period.

Kim *et al.* [67] applied singular value decomposition (SVD) to exploit the subspace of matrix $\boldsymbol{Z}$ and construct the grid topology. $\boldsymbol{Z}$ is constructed using a sample of the system measurements over a period t where the ith row represents the measurements at time i

$$\boldsymbol{Z} = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1m} \\ z_{21} & z_{22} & \cdots & z_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z_{t1} & z_{t2} & \cdots & z_{tm} \end{bmatrix} \tag{39}$$

The covariance matrix of $\boldsymbol{Z}$, $\boldsymbol{\Sigma}_Z$, is computed as follows:

$$\boldsymbol{\Sigma}_Z \triangleq E[(\boldsymbol{Z} - E[\boldsymbol{Z}])(\boldsymbol{Z} - E[\boldsymbol{Z}])^{\mathrm{T}}] = \boldsymbol{H}\boldsymbol{\Sigma}_x\boldsymbol{H}^{\mathrm{T}} + \sigma^2\boldsymbol{I} \tag{40}$$

where $\sigma^2\boldsymbol{I}$ is the covariance matrix of the error vector $\boldsymbol{e}$ ($\boldsymbol{z} = \boldsymbol{Hx} + \boldsymbol{e}$) and $\boldsymbol{\Sigma}_x$ is the covariance matrix of the state vector $\boldsymbol{x}$. The basis matrix of $\boldsymbol{H}\boldsymbol{\Sigma}_x\boldsymbol{H}^{\mathrm{T}}$ is calculated by applying SVD to $\boldsymbol{\Sigma}_Z$, i.e. by finding a unitary matrix $\boldsymbol{U}$, a rectangular diagonal matrix Λ, and a unitary matrix $\boldsymbol{V}$ such that $\boldsymbol{\Sigma}_Z = \boldsymbol{U}\Lambda\boldsymbol{V}^{\mathrm{T}}$. The n columns of the unitary matrix $\boldsymbol{U}$ are equivalent to the eigenvectors of matrix $\boldsymbol{H}\boldsymbol{\Sigma}_x\boldsymbol{H}^{\mathrm{T}}$, which form the basis of the column space of $\boldsymbol{H}\boldsymbol{\Sigma}_x\boldsymbol{H}^{\mathrm{T}}$. Since the column space of $\boldsymbol{H}\boldsymbol{\Sigma}_x\boldsymbol{H}^{\mathrm{T}}$ is equivalent to the column space of $\boldsymbol{H}$, the n columns of $\boldsymbol{U}$ also form a basis of the column space of $\boldsymbol{H}$. Thus, the attacker can construct a potential attack vector $\boldsymbol{a}$ using matrix $\boldsymbol{U}$.

Similarly, Yu and Chin [68] leveraged principal component analysis (PCA) to construct blind FDIAs. PCA is a dimensionality reduction and data transformation method used to reduce a large set of variables to a small set while retaining the critical information of the original set. The authors apply PCA to $\boldsymbol{z}$, which is the measurement vector, and obtain a transformation matrix $\boldsymbol{H}_{\mathrm{pca}}$, as well as, the principal components vector $\tilde{\boldsymbol{x}}$, illustrated in (41)

$$z \simeq \begin{bmatrix} \tilde{\boldsymbol{p}}_1 & \tilde{\boldsymbol{p}}_2 & \cdots & \tilde{\boldsymbol{p}}_n \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{x}}_1 \\ \vdots \\ \tilde{\boldsymbol{x}}_n \end{bmatrix} \equiv \boldsymbol{H}_{\mathrm{pca}}\tilde{\boldsymbol{x}}_{\mathrm{pca}} \tag{41}$$

where $\boldsymbol{z}$ is an $m \times 1$ over determined measurement vector, $\boldsymbol{H}_{\mathrm{pca}}$ is an $m \times n$ matrix with n eigenvectors ($\tilde{\boldsymbol{p}}_i$), and $\tilde{\boldsymbol{x}}_{\mathrm{pca}}$ is the $n \times 1$ principal components vector. The PCA reduced $\boldsymbol{H}_{\mathrm{pca}}$ is leveraged for the construction of the blind FDIA and the formation of the attack measurement vector $\boldsymbol{z}_a$ as described in (42) and (43)

$$\boldsymbol{a}_{\mathrm{pca}} = \boldsymbol{H}_{\mathrm{pca}} \times \boldsymbol{c} \tag{42}$$

$$\boldsymbol{z}_a = \boldsymbol{z} + \boldsymbol{a}_{\mathrm{pca}} \tag{43}$$

Teixeira *et al.* [57] studied stealthy FDIAs in dynamic systems where the $\boldsymbol{H}$ matrix is changing. The attack is constructed following [63]. The authors mathematically prove the possibility of

Table 1 FDIA categories

FDIA categories	References	Examples
attack with full knowledge	[12, 17, 63, 64]	access specific meters, minimise the number of attacked meters
attack with incomplete knowledge	[32, 61, 65–69]	use online and offline data, utilise market price data

Table 2 Impacts of FDIAs on power system

Impacts	References	Examples
power grid operation	[70–72]	LR attack
distributed energy routing process	[57, 73]	energy deceiving attack
affect the operation of the deregulated electricity market	[74, 75]	economic attack

local stealthy FDIAs. If the changes in $\boldsymbol{H}$ do not affect the compromised measurements, the attack vector – constructed using the original $\boldsymbol{H}$ – remains stealthy after any system change. Furthermore, the authors empirically study the impact that the magnitude of an attack vector can introduce on the success rate of the attack. They conduct experiments utilising the IEEE 39 bus test case with energy management system (EMS) software including SE and residual-based BDD. The results validate that even large attack vectors can bypass the detection.

Kekatos *et al.* [66] proposed an algorithm leveraging locational marginal prices (LMPs), which is computed from a network-constrained economic dispatch problem to recover the grid Laplacian with a regularised MLE [76, 77]. Esmalifalak *et al.* [32] proposed an independent component analysis algorithm to estimate the $\boldsymbol{H}$ matrix and system topology by observing the power flow measurements. Similarly, Liu *et al.* [69] showed that the attacker could launch an attack in a local region possessing only local system information.

Deng *et al.* [61] proposed a practical FDIA model against SE in DSs, where the attacker can successfully launch FDIAs with partial system information. The authors illustrate how the attacker could estimate the system states based on a small amount of power flow or power injection measurements. The proposed method reduces the cost of obtaining system states, making FDIAs more realistic against SE on the distribution level. The proposed model is demonstrated on an IEEE test feeder. The results show that attacks can effectively compromise the SE avoiding detection.

3.2 Impacts of FDIAs on power systems

FDIAs can cause significant economic or physical impacts on the power system. In this section, we review the effects of FDIAs and summarise them in Table 2.

3.2.1 Load redistribution (LR) attack: Yuan *et al.* [70] proposed a particular type of FDIA, called LR attack, which targets the security-constrained economic dispatch (SCED) and can potentially affect the power grid operation. The power system uses SCED to reduce the total system operation cost by properly re-dispatching the generation output. Owing to LR attacks, the SCED provides incorrect solutions based on corrupted state estimates and drives the system to infeasible operating states. Moreover, the LR attacks can potentially cause load shedding events immobilising any immediate corrective action [70, 72].

3.2.2 Energy deceiving: Lin *et al.* [73] studied a new variation of FDIAs named energy deceiving attacks, which target the routing process of energy distribution. The authors introduce a distributed energy routing scheme to find the optimal route for energy flow between nodes of the grid. Each node could be either an energy consumer or an energy producer. To distinguish different nodes, a measurement tool is used (e.g. a smart meter). All nodes communicate with each other to share information such as measurements, requests, and demands. The energy deceiving attack is conducted by spoofing the information exchanged between nodes. Malicious energy information or malicious link-state information is injected into the energy request and response messages of the nodes. A successful attack can manipulate the memory of a measurement tool and inject the false demand and supply messages to the grid. The authors analyse the impact of the energy deceiving attack based on the proposed method and conclude that the attack would create imbalances between demand and supply. As a result, the cost of energy distribution can severely increase.

Table 3 Defense strategies against FDIAs

Methods	References	Limitations
protecting minimum sets of meters	[17, 55, 78–80]	protected only measurements that are trusted
using PMUs	[81, 82]	vulnerable to GPS spoofing attack
defences based on game theory	[83–86]	rationality of the agents and modelling challenges
defences based on cryptographic methods	[87–89]	not practical for large systems with a limited budget
topology defence method	[90]	it is possible for the attacker to learn and guess the new configuration
proactive approaches to mitigate FDIAs	[91]	computationally intensive

3.2.3 Economic attack: In terms of impacts on economic operations, Xie *et al.* [74] demonstrated how FDIAs affect the energy market. Real-time market prices are determined using ex-post LMP values, which in turn rely on the actual SCADA measurements to calculate their final settlement prices. Thus, if an attacker can manipulate the system measurement data, the results of the SE, and consequently, the electric energy price can be affected. The authors use a linear form of optimal power flow (OPF), DC OPF, to calculate the LMPs and formulate the attack as a convex optimisation problem. There are two cases applied to the IEEE 14 bus system, one for a single congested line and the other for three congested lines. The study illustrates that FDIAs can manipulate the nodal price of the ex-post market and can also bring financial profits to attackers. The authors also explore, in a later study [75], more realistic attack scenarios assuming threat models in which the attackers can only manipulate a limited number of sensors.

4 Defences against FDIAs

In this section, we discuss existing countermeasure approaches against FDIAs. Table 3 lists different detection methods and their limitations.

Liu *et al.* [12] showed that if the attacker knows the system matrix $\boldsymbol{H}$ and can compromise $k \geq m - n + 1$ meters, then she/he can effectively inject the malicious vector to the measurement vector z without being detected. Therefore, it is crucial to identify and protect a set of meter measurements. Bobba *et al.* [17] highlighted the requirement to identify and protect a set of measurements to prevent FDIAs. Both studies leverage a brute force approach to identify the set of measurements that require protection. Dan *et al.* [55] proposed a greedy algorithm to find the minimum set of measurements essential to be protected. Owing to the large number of meters in the power system and the limited protection budget, the authors consider protecting a subset of meters ρ to increase the security level of the system. Subsequently, the authors consider two objective functions:

i. Maximise the minimum attack cost:

$$\rho^m = \max_{\rho} \min_{k} \alpha_k \quad \text{Subject to}: \varrho(\rho) \leq \pi \tag{44}$$

where $\varrho(\rho)$ is the protection cost, π is the budget, and α_k is defined as the security index of the kth measurement (38).

ii. Maximise the average attack cost

$$\rho^m = \max_{\rho} \frac{1}{m} \sum_{k \in M} \alpha_k \tag{45}$$

To minimise the protection cost, Bi and Zhang [79] framed the protection problem as a variant of the Steiner tree problem in a graph. Given an undirected graph with non-negative edges and a set of vertices, which represent transmission lines and buses in the power network, the Steiner tree problem entails finding a tree with minimum weights, which contains all the vertices [92]. To select the minimum set of meters to be protected, they propose two algorithms: a Steiner vertex enumeration algorithm and MILP. The proposed algorithms significantly reduced the computational complexity and are able to find the minimum set of meters

necessary to be protected. The shortcomings of protecting a minimum set of meters are two-fold (i) possible decrease of redundancy and (ii) occasional lack of security.

Kim and Poor [82] proposed another approach to protect the minimum set of measurements. The authors suggest installing PMUs in the critical substations of electric power systems. PMU is a global positioning system (GPS)-based measurement device that directly measures synchronised voltages and phase angles. The GPS connected to the PMU devices time-stamps the measurements, thereby preventing the measurements from being compromised by attackers. Even though installing PMUs is a powerful solution to prevent FDIAs, it is costly to deploy PMUs on a large scale. The cost of a large-scale PMU deployment has led to additional research on the optimal placement of PMUs in power systems. To reduce the number of PMUs used in a system, Chen and Abur [81] developed a placement algorithm to find out locations for PMU installations. In addition to cost concerns, PMUs are vulnerable to GPS spoofing attacks, which could invalidate the PMU data time-stamps (by faking the GPS signal) and compromise the reliability of all the synchrophasor data [93].

In terms of approaches that utilise game theory concepts, Wei *et al.* [83] proposed a stochastic-based approach for the protection of the power system from coordinated attacks. Coordinated FDIAs manipulate power system measurements – by emulating the real behaviour of the system – and thus remain undetectable. The authors' design an optimal load shedding algorithm to assess the effects of coordinated attacks, e.g. where and how many loads to be shed under successful attacks. The effect of the attack is then used in a resource allocation stochastic game to model the interactions between a malicious attacker and a defender. The authors prove the effectiveness of the proposed approach in protecting the power system from FDIAs. However, the game theory model is not scalable, nor realistic, since it models the interaction between defenders and attackers as a series of causal events [94].

Sun *et al.* [87] proposed an encryption-based method leveraging a dynamic secret to protect wireless communications. The method encrypts the measurement data by using the aforementioned secrets, which are dynamically generated at the sender's side to protect the security and privacy of the power data. To create the encryption key, instead of using the transmitted data, which are vulnerable to eavesdropping attacks, the authors utilise a packet re-transmission communication protocol. The re-transmission protocol employs steganography to encrypt the measurement data rather than send them in plaintext. Although encryption methods can protect measurements against FDIAs, they introduce computation overheads and increase communication latency, which may become impractical for large and densely interconnected systems with limited edge-computing resources. To address computation overheads induced by encrypted communications (for the measurement exchange), the authors in [95] proposed a lightweight hardware-based security primitive, which leverages real-time battery entropy for ephemeral key generation and secure authentication between power system assets.

Shahid *et al.* [90] proposed a new topology defence model to protect the power system from stealthy blind FDIAs. The authors exploit the concept of dummy measurement values in the power network to detect stealthy attacks in the network. In their model, meters in the smart grid send two different sets of measurements to EMS, which include dummy and real measurements. The dummy measurements rely on the real measurements and are assigned by operators at the control centre. The dummy measurements are only known to the SO. Thus, the SO could quickly detect any attack in the system by comparing all the received measurements against the dummy measurements. However, the mentioned defence model can protect the system only for a limited duration since the attacker can eventually learn or guess the configuration.

Li *et al.* [91] introduced a proactive approach to mitigate FDIAs (PAMA) in smart grids. PAMA can protect the crucial system information such as the original measurement data, system configurations, and grid connections from leakage or even theft. The proposed method focuses on the proactive prevention and mitigation of FDIAs before an attack is conducted. To enhance system robustness against FDIAs, the authors design a distributed computing model that integrates the Paillier cryptosystem to encrypt all system information (including the original measurement data, system configurations, and grid connections). However, PAMA is computationally intensive and challenging to model.

Table 4 Summary of machine learning methods to detect FDIAs

Type of machine learning method	Algorithms	References
supervised learning	SVM and KNN	[31, 33] [97, 98]
semi-supervised learning	semi-SVM	[31, 99, 100]
unsupervised learning	FCM	[101–103]
deep learning	MLP, RNN, DBN	[99, 103–105] [98, 106–109]

5 Machine learning for FDIAs detection

Machine learning is a form of artificial intelligence that enables computers to learn and improve without being explicitly programmed [96]. Different machine learning algorithms have been proposed by researchers to enable FDIAs detection. The existing use of machine learning algorithms can be categorised as shown in Table 4. In this section, we first discuss the metrics used for the performance evaluation of machine learning-based FDIAs detection algorithms. Leveraging these metrics, we review the existing literature and compare their performance.

5.1 Performance metrics

A multitude of metrics has been adopted to evaluate the performance of the detection methods. Accuracy, precision, recall, F1 score, and receiver operating characteristic (ROC) curve are among the most common metrics. With the true label of a measurement and its predicted label, the output of a detection model can be divided into true positive (TP): indicating a correct positive prediction, true negative (TN): a correct negative prediction, false positive (FP): an incorrect positive prediction, and false negative (FN): an incorrect negative prediction [110].

Accuracy is the ratio of the number of correct predictions to the number of total predictions

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{46}$$

Accuracy is meaningful when the measurement data is balanced (when the number of positive and negative measurement samples are equal). To evaluate the performance of the detection model with imbalanced data, precision, recall, and F1 score are often considered. Precision (also called positive predictive value) describes the capability of a model to identify an attack's overall true positive predictions [111]. It is represented as the ratio of the correct positive predictions to the number of samples labelled as positive

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{47}$$

The recall (also called sensitivity) gives the model the capability to identify all attacks [111]. Recall is described as the ratio of the number of correct positive predictions to the number of positive samples

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{48}$$

From (47) and (48), it can be observed that precision and recall are closely related. For a given model, a decrease in FP (precision) leads to an increase in FN (recall), and vice versa. To achieve an optimal trade-off between precision and recall, the F1 score is used to combine these two metrics. To avoid being heavily impacted by

extreme values of precision or recall, the F1 score is designed as the harmonic average of precision and recall, as shown below

$$\text{F1 score} = \frac{2 \times (\text{Recall} \times \text{Precision})}{\text{Recall} + \text{Precision}} \tag{49}$$

For a given classifier, recall and precision may vary a lot with different sets of measurements, e.g. balanced data and unbalanced data, making it hard to evaluate the performance of a classifier. To have a stable representation of the classifier performance, the ROC curve is often used. In a continuous binary classifier, the output is a continuous variable ranging from 0 to 1. Thus, a threshold is leveraged to divide the outputs into positive and negative. Different thresholds result in different TP rates (TPR, equals to recall) and FP rates (FPR = FP/(FP + TN)). ROC curve plots TPR (y-axis) against FPR (x-axis) under different thresholds. The closer the ROC curve is to the upper left corner or coordinate (0, 1) (the larger the area under the curve), the better the performance. ROC illustrates how well the detection method distinguishes between the attacked and the secured measurements.

5.2 Supervised learning algorithm

Machine learning algorithms can be classified into supervised learning, semi-supervised learning, and unsupervised learning. In supervised learning, inputs and desired outputs are provided to the machine to construct a function that maps the input to the desired output.

Detecting FDIAs is considered a supervised binary classification problem. The objective of the binary classifier is to decide whether the given data $\boldsymbol{s}$ with m features is either $\boldsymbol{z}$, a normal measurement vector (negative class) or $\boldsymbol{z}_a = \boldsymbol{z} + \boldsymbol{a}$, an attacked measurement vector (positive class) [33]. The output class labels are

$$y = \begin{cases} +1 & \text{for } \boldsymbol{a} \neq 0 \\ -1 & \text{for } \boldsymbol{a} = 0 \end{cases} \tag{50}$$

where $\boldsymbol{a}$ is the attack vector.

The common used supervised learning algorithms are perceptrons [112], support vector machines (SVMs) [113], k-nearest neighbours (KNNs) [114], and logistic regression [115]. In perceptrons, a weight vector $\boldsymbol{w} \in \mathscr{R}^{M_{\text{Tr}}}$ is trained such that the output label, y_i, of a sample s_i is predicted by the following classification function:

$$f(s_i) = \text{sign}(\boldsymbol{w} \cdot s_i) = \begin{cases} 1 & \text{for } \boldsymbol{w} \cdot s_i \geq 0 (a \neq 0) \\ -1 & \text{for } \boldsymbol{w} \cdot s_i < 0 (a = 0) \end{cases} \tag{51}$$

During the training phase, the weight vector is updated for each training sample as $\boldsymbol{w}(i+1) = \boldsymbol{w}(i) + \Delta \boldsymbol{w}$, where $\Delta \boldsymbol{w} = \gamma(y_i - f(s_i))s_i$ and γ is the learning rate. From the classification function, we can see that the convergence of the perceptron algorithm can be guaranteed when the samples are linearly separable. Therefore, it is suitable for FDIAs detection only when a hyper plane can separate the measurements.

In SVMs, a hyper plane is constructed to separate two different classes. The hyper plane can be represented by a weight vector $\boldsymbol{w}$, and a bias value b. The decision boundaries for the linear separable data can be formulated as two parallel hyper planes using (52)

$$\begin{cases} \boldsymbol{w}^{\text{T}} s_i + b = +1, & \text{if } y_i = +1 \\ \boldsymbol{w}^{\text{T}} s_i + b = -1, & \text{if } y_i = -1 \end{cases} \tag{52}$$

In (52), each line represents a support vector, as shown in Fig. 3. Margin D is the separation area between the two support vectors and can be computed as

$$D = \frac{2}{\boldsymbol{w}^2} \tag{53}$$

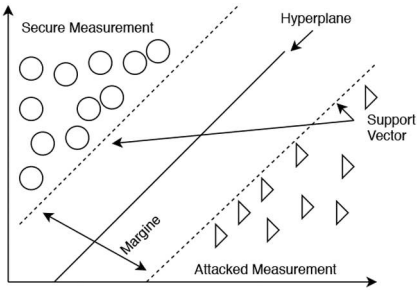

Fig. 3 *SVM concept*

The hyper planes can be determined by solving

$$\begin{gathered} \min_{\boldsymbol{w}, \xi, b} \ \| w \|_2^2 + \zeta \sum_{i=1}^{M^{\text{Tr}}} \xi_i \\ \text{Subject to:} \quad y_i(\boldsymbol{w}^{\text{T}} \cdot s_i + b) - 1 + \xi_i \geq 0 \\ \xi_i \geq 0 \ \forall i = 1, 2, 3, \ldots, \boldsymbol{M}_{\text{Tr}} \end{gathered} \tag{54}$$

where ζ is the adjustable regularisation parameter, ξ_i is the slack variable for the non-linear separable training set, and $\boldsymbol{M}_{\text{Tr}}$ is the feature vector.

KNN is another supervised learning algorithm that assigns labels to an unlabelled sample according to its KNNs. The Euclidean distance is used to determine the similarity between a given labelled sample, s_i, and an unlabelled sample, s'_i. The set of KNNs for a given measurement sample can be determined using the Euclidean distance as follows [116]:

$$\begin{gathered} \| s'_i - s_{i(1)} \|_2 \leq \| s'_i - s_{i(2)} \|_2 \leq \ldots \leq \| s'_i - s_{i(\boldsymbol{M}_{\text{Tr}})} \|_2, \\ \aleph(s'_i) = \{ s_{i(1)}, s_{i(2)}, \ldots, s_{i(k)} \} \end{gathered} \tag{55}$$

Majority voting is one of the most commonly used methods for assigning labels from the set of KNNs of s_i. KNN is easy to implement but it fails to work when the size of the data sample is smaller than the dimension of the feature vector [117].

The logistic regression algorithm assumes that the distribution of the label y_i of data s_i follows the following logistic function [118]:

$$P(y_i | s_i) = \frac{1}{1 + \exp(-y_i(\boldsymbol{w} \cdot s_i + b))} \tag{56}$$

The weight vector $\boldsymbol{w}$ is estimated by maximising the following cost function:

$$\boldsymbol{J}(\boldsymbol{w}) = -\frac{1}{\boldsymbol{M}_{\text{Tr}}} \sum_{i=1}^{\boldsymbol{M}_{\text{Tr}}} \log(1 + \exp(-y_i(\boldsymbol{w} \cdot s_i + b))) \tag{57}$$

A brief comparison of various machine learning methods is presented in [31]. The paper is one of the first research works to utilise supervised learning algorithms for FDIAs detection. The authors used a hierarchical network in which the measurements are grouped as clusters, and each cluster is regarded as a sample s_i. The false data is directly injected into the measurements before the measurements are grouped into clusters. The detection method is based on the observations made in [18]. According to Liu *et al.* [18], the distance between samples determines the attack vector

$$\| s_i - s_j \|_2 = \begin{cases} \| z_i - z_j + \boldsymbol{a}_i - \boldsymbol{a}_j \|_2, & \text{if } \boldsymbol{a}_i, \boldsymbol{a}_j \neq 0 \\ \| z_i - z_j + \boldsymbol{a}_i \|_2, & \text{if } \boldsymbol{a}_i \neq 0, \boldsymbol{a}_j = 0 \\ \| z_i - z_j \|_2, & \text{if } \boldsymbol{a}_i, \boldsymbol{a}_j = 0 \end{cases} \tag{58}$$

Therefore, by looking into the distance between two samples, it is possible to detect a FDIA. In their experiment, the performance of different machine learning algorithms is evaluated against FDIAs with different sparsity k/m (the ratio of measurements that the

attacker has access to). Accuracy, precision, and recall are used as performance metrics. The results proved that the machine learning algorithms perform better than any other algorithm (e.g. state vector estimation approach) in detecting FDIAs. Although SVMs achieved the highest prediction accuracy, they also present some limitations, such as the selection of the kernel and sensitivity to the sparsity of the system. KNN is very sensitive to system size and performed better for the small-sized systems. Despite conducting plenty of experiments, Ozay *et al.* [31] did not evaluate the performance of detection algorithms for stealthy FDIAs. Furthermore, only the sparsity of injected data was considered. The magnitude of the injected data could potentially impact as well as the operation of the system and the performance of the detection methods. Last, the lack of attack data can result in imbalanced data samples during the detector training process affecting its classification accuracy.

Considering the limitations of [31], a similar work is conducted in [33]. Both works utilise closely related system models. Two assumptions are taken into consideration when the attack vectors are created in the adversary model: (i) that the injected value a_i is greater than the noise level and (ii) that the mean of the attack vector $\boldsymbol{a}_i$ is larger than the variance of the attack vector. Attack vectors with different sparsity and variance (The variance reflects the magnitude of disturbances caused by false data.) are tested in their experiments. To solve the imbalanced data problem, they propose the extended nearest neighbour (ENN) algorithm. For each class, ENN measures the average ratio of the nearest neighbours belonging to the same class. Instead of using majority voting, the label of a sample was predicted by finding the class which presents the greatest ratio variability with the sample labelled in that class. The performance of SVMs, KNN, and ENN is then experimentally evaluated. Accuracy and F1 scores are used as the performance metrics. SVMs outperformed KNN and ENN in most of the test cases. A critical range of sparsity was observed in which the accuracy and F1 score increased significantly. However, this is reasonable since the distance increases $\| s_i - s_j \|$ when the sparsity increases, which leads to more distinct classes. The experiment was conducted on the IEEE 30 bus system. The detection performance of the algorithms in larger systems was not demonstrated.

Esmalifalak *et al.* [97] proposed a distributed SVM algorithm. Each substation owned a training set and stealthy FDIAs, which could bypass BDD methods based on their corresponding residuals. Before training, PCA is applied to the training set to reduce the feature dimension. To avoid a huge volume of data exchange, each substation is trained using a local classifier, and only the locally optimised weight vectors are exchanged. Their optimisation problem, (54), is provided below

$$\begin{aligned} \min_{w_k, \xi_k, b_k} \quad & \| w_k \|_2^2 + \zeta \sum_{k=1}^{n} \sum_{i=1}^{\mathrm{Tr}} \xi_{ki} \\ \text{Subject to:} \quad & y_{ki}(w_k^{\mathrm{Tr}} \cdot s_{ki} + b_k) - 1 + \xi_{ki} \geq 0 \\ & \xi_{ki} \geq 0 \ \forall i = 1, 2, 3, \ldots, m; k = 1, \ldots, n \end{aligned} \tag{59}$$

where n is the number of substations and w_k is the local optimisation parameter. The alternating direction method of multipliers is used to solve this distributed optimisation problem. Experiments are performed on the IEEE 118 bus system. The authors empirically verify the convergence of distributed SVM classifiers to centralised SVMs with different numbers of substations.

To recapitulate, supervised learning methods have achieved superior performance in comparison with traditional residual-based BDD methods. Among the aforementioned algorithms, SVMs have demonstrated to achieve the highest accuracy. According to Esmalifalak *et al*. [97], the curse of the dimensionality problem can be solved by leveraging PCA, which significantly enhanced the efficiency of machine learning algorithms. Nevertheless, most of the attack data are often generated randomly in the experiments while a sophisticated adversary would deliberately choose attack vectors considering the system dynamics. The performance of the proposed methods against such sophisticated FDIAs still remains unknown. Moreover, most of the prior works conducted simulation experiments. Thus, the efficiency of existing methods, if applied to real power system deployments, cannot be guaranteed.

5.3 Semi-supervised learning

In semi-supervised learning, the majority of the given data is unlabelled. Although semi-supervised learning algorithms are the least common learning approaches applied for detection of FDIAs, we still introduce them in this survey work for completeness. An example of a semi-supervised learning algorithm is a semi-supervised SVM (S_3VM). S_3VM assumes that samples with different labels are clustered into different groups and that the diameter of each cluster is small enough to avoid sub-clusters [119]. The objective function of S_3VM is defined as

$$\min_{\boldsymbol{w}, b} \ \zeta \left[\sum_{i=1}^{\boldsymbol{M}_{\mathrm{Tr}}} L^{\mathrm{Tr}}(s_i, y_i) + \sum_{i=1}^{\boldsymbol{M}_{\mathrm{Ts}}} L^{\mathrm{Ts}}(s'_i) \right] + \| w \|^2 \tag{60}$$

where $y = w^{\mathrm{T}} s_i + b$ and ζ is the regularisation parameter, L^{Tr} and L^{Ts} are the loss function of the training and test samples, respectively.

Foroutan and Salmasi [99] investigated FDIA detection methods by using the S_3VM based on Gaussian mixture distributions. According to Filho [120], a finite mixture distribution model, defined as a convex combination of two or more probability density functions, is capable of approximating any arbitrary distributions due to its flexibility in modelling complex data. The authors assume that all FDIAs have the same amount of energy or $\boldsymbol{c}$ vector, where $\boldsymbol{a} = \boldsymbol{Hc}$, and that they have the same mean squared error. In the adversary model, the attack vectors are designed based on the minimum energy residual attack and sparsest attack, introduced in [78, 121], respectively. In the training phase, a positive data set, i.e. a data set with attacked measurements, was used to build the Gaussian mixture model. Then, a mixture of data sets consisting of both positive and negative labels (attacked and normal measurements) determines the threshold. In the evaluation phase, the unlabelled data set used for testing and F1 score evaluates the performance of the results. PCA was applied to the data set to overcome measurement dimensionality issues. The authors demonstrate the performance of the proposed detection method on the IEEE 118 bus power system. To generate diversified datasets, different topological networks are constructed using Monte–Carlo simulations. The performance of the proposed detection method depends on the selection of a proper threshold. A high-threshold value reduces recall while a low-threshold value lowers precision. The impact of the detection algorithms is illustrated with a ROC curve. Although the proposed model demonstrates a high F1 score compared with other machine learning algorithms (e.g. SVMs and perceptrons), it performs well only when the attacked measurements and the real measurements lie in distinct regions of the feature space, i.e. the attacked data can be effortlessly isolated.

Another detection method based on the S_3VM algorithm is proposed in [31]. The input samples are integrated into the cost function forming the following optimisation problem:

$$\min \ \| W \|_2^2 + \zeta_1 \sum_{i=1}^{M_{\mathrm{Tr}}} L^{\mathrm{Tr}}(S_i, y_i) + \zeta_2 \sum_{i=1}^{M_{\mathrm{Te}}} L^{\mathrm{Te}}(S'_i) \tag{61}$$

where ζ_1 and ζ_2 are the cost parameters, L^{Tr} and L^{Te} are the loss functions for the training and testing samples. In the simulation, the authors use default values for the parameters as suggested in [100]. The experiments are conducted on IEEE 9, 57, and 118 bus systems, and the measurement matrix is generated using Matlab's Matpower toolbox. Compared with supervised learning algorithms, S_3VM demonstrated improved robustness against data sparsity despite the fact that S_3VM still remains sensitive to unbalanced data samples.

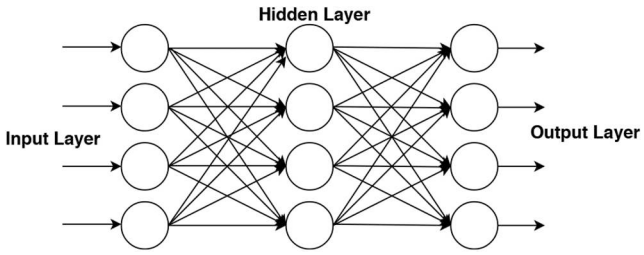

Fig. 4 *MLP concept*

5.4 Unsupervised learning

Unsupervised learning algorithms group the unlabelled samples based on the similarities and differences between samples, without any prior training. Clustering is the most popular unsupervised learning method where the measurement samples are grouped based on the distance between samples in the feature space. Different distance metrics can be chosen (e.g. Euclidean distance).

k-means (also called hard *c*-mean) is an example of a clustering method, which divides data into k groups. *k*-means iteratively assigns each data point to one of the k groups, whose centroid has the minimum distance to the data point in the feature space. Each centroid in a cluster is a collection of features, which define a group. The centroids are updated at each round. Several techniques are used to validate the *k*-value including cross-validation and other information criteria. Fuzzy *c*-means (FCM) clustering is another type of *k*-means clustering, which assigns data points to two or more clusters [101]. Each point belongs to a cluster based on a corresponding probability value, rather than having a binary value as is the case of *k*-means clustering. In FCM, the clustering problem can be solved by minimising the following equation:

$$J = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m} \parallel x_i - C_j \parallel^2, \quad 1 \le m < \infty \tag{62}$$

where N is the number of data points, $C = 2$ is the number of the clusters (cluster of attacks and cluster of normal measurements), x_i is the ith dimensional measured data, and C_j is the centre of the jth cluster, which is determined using

$$C_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}} \tag{63}$$

where u_{ij} is the degree of membership of the ith measurement. The updated membership u_{ij} computed by the following equation:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}((\parallel x_i - C_j \parallel)/(\parallel x_i - C_k \parallel))^{2/(m-1)}} \tag{64}$$

Mohammadpourfard *et al.* [103] presented a visualisation based on the unsupervised anomaly detection method and FCM clustering to detect and locate FDIAs. The authors also propose a localisation method that helps in identifying the attack after topology reconfigurations and the integration of different resources. When FDIAs occur, the probability distributions of system states deviate significantly from normal states, hence enabling FDIA detection. First, the authors normalise the data, and then various statistical measures are applied to characterise the probability distribution of each state vector. PCA is applied to the new feature set to reduce the dimensionality of data and to visualise them in a two-dimensional space where the grid operators can determine whether an attack has occurred or not (using patterns of normal and abnormal data). FCM is used to detect outliers and locate the FDIAs. Load data from the New York Independent System Operator are used for the simulations. FDIAs data are generated on the IEEE 9 and 14 bus system with the assumption that the adversary decreases or increases a specific state variable by at least 6% of its original value. The proposed method is applied to two different FDIAs scenarios: detecting FDIAs with and without topology changes. Compared to supervised learning algorithms such as SVMs and KNN, the proposed model achieves higher detection accuracy [103].

Yang *et al.* [102] proposed three different anomaly detection approaches to detect FDIAs: (i) local outlier factor, (ii) isolation forest, and (iii) robust covariance estimation. The local outlier factor is a density-based anomaly detection method that measures the local standard deviation of any given data point from its neighbours by comparing their local density [122]. Isolation forest is an outlier detection technique based on decision trees that does not employ any distance or density measure and can handle large, high-dimensional datasets. Robust covariance estimation is another anomaly detection method based on the elliptic envelope fitting method, which assumes that the given data is a Gaussian distribution and defines the shape of the data. An IEEE 14 bus system case is used to evaluate the mentioned detection approaches. Attack vectors generated with Gaussian distributed non-zero elements have the same mean and variance as the original measurement set. The authors use PCA to reduce the data dimension from 41 to 2, to reduce noise, and simplify the detection problem. All proposed methods achieve high accuracy for FDIAs detection. However, these three detection methods achieve high detection rates only when the contamination rate is known and small [102].

5.5 Deep neural network

Deep learning algorithms mimic the human brain structure, functions, and are one of the fastest developing artificial intelligence technologies. Although deep learning algorithms require time and large amounts of data for their training stage, they have been applied for FDIAs detection achieving high-accuracy rates [123].

Multilayer perceptrons (MLPs), also called feed-forward neural networks, are deep learning models where information flows in only one direction, i.e. from the input through the hidden layers to the output, as shown in Fig. 4 [124]. They consist of an input layer, which receives the input signals, one or more hidden layers to construct the approximation function, and an output layer that predicts the final decision based on the input and approximation function.

Multiple studies where MLPs have been applied to detect FDIAs have been reported in the literature [103–105]. In these works, the FDIA detection problem is formulated as a supervised classification problem. In MLP-assisted binary classification, a linear combination of an input weight vector produces a single output, as shown in the following equation:

$$y = \varphi\left(\sum_{i=1}^{n} w_i \boldsymbol{s}_i + b\right) \tag{65}$$

where y is the estimated output of the activation function, w is the weight, $\boldsymbol{s}$ is the input vector, b is the bias, and φ is the non-linear activation function. The activation function is an essential feature of the MLP architectures. It decides whether a neuron should be fired or not by calculating the weighted sum of inputs and adding a corresponding bias to it. Sigmoid, tanh, and rectified linear unit (RELU) are examples of activation functions. RELU is the most widely used function because it is fast and less computationally expensive. MLPs use back-propagation training algorithms and the weights are updated using gradient descent to minimise the error function.

Ashrafuzzaman *et al.* [125] proposed different MLP structures for the detection of FDIAs in an AC static SE system topology. The paper assumes that partial knowledge of the system, including the $\boldsymbol{H}$ matrix and other parameters, is known to the attacker. A standard IEEE 14 bus system is used to conduct the simulation. The Matpower toolbox is used to generate the measurement vector z, which contains 122 measurement features (40 active and reactive power flows, 14 power injections, and 27 voltage measurements). The authors train the MLP using stochastic gradient descent (an optimisation technique for the network parameters update) and tanh as the activation function. Four models with different network

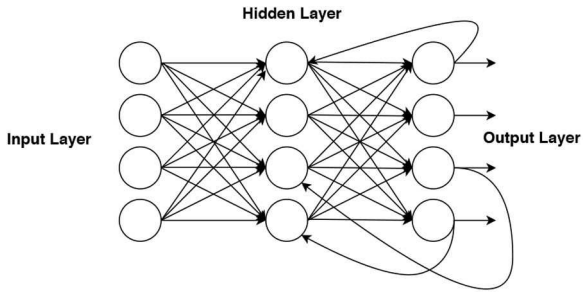

Fig. 5 *RNN concept*

architectures are utilised for the detection. The first model consists of one hidden layer with 100 neurons, and the second model consists of three hidden layers with 150 neurons. For the third and fourth models, the authors use the first and second models with a regularisation value of 0.0001. Regularisation is a technique used to reduce or prevent over fitting of a neural network. The models' detection performance is compared with other machine learning algorithms. Accuracy, precision, recall, and F1 score are used to evaluate the MLP detection. The results of the four discussed models are similar with an accuracy of around 98%.

Similarly, Foroutan and Salmasi [99] applied MLPs to detect FDIAs and compared them with common machine learning detection models. The network consists of an input layer, one hidden layer, and an output layer. Tanh is used as an activation function. Although MLP produced higher detection accuracy than the other algorithms, their training process is slow. Ganjkhani *et al.* [104], on the other hand, introduced a novel MLP algorithm leveraging a non-linear autoregressive exogenous (NARX) configuration, which takes into account the high correlation between power system measurements as well as the state variables. The NARX configuration is used for time series prediction and can predict step-ahead values of the states by factoring measurement values and historical data as input variables. In the experiment, NARX is constructed with an input and a hidden layer with different numbers of neurons and sigmoid linear activation functions. The historical data contained 6048 measurement vectors and state variables. The detection model is trained using 70% of the historical data and 30% for the testing and validation.

A recurrent neural network (RNN) is a sophisticated deep learning algorithm that uses internal memory or feedback loops, as shown in Fig. 5. Unlike MLPs, RNNs use the information from past events for their predictions. A long short-term memory unit can be added to a standard RNN to solve the problem of vanishing gradient descent and store information for an extended period [126]. RNNs formulate the FDIAs detection problem as a sequence of prediction. Results from previous time steps are used for the prediction of the current output rendering RNNs efficient in detecting manipulated measurements.

Ayad *et al.* [98] utilised RNNs as a sequence classification algorithm for detecting FDIAs in DC SE. Back-propagation through time, an extensive type of back-propagation, is applied to train the algorithm. The authors run the training algorithm multiple times to produce the optimal set of parameters that achieve the least error. Then, the optimal parameters are applied to the network for the prediction of the test data classes. Since the output ranged from 0 to 1, a threshold is set to determine the output class as either 1 or 0 (1 is compromised, and 0 is normal). The IEEE 30 bus test case is used for the experiments with 112 measurement vectors. The proposed model obtains outstanding detection results with an accuracy rate of 99%.

James *et al.* [107] proposed an RNN architecture for detecting FDIAs in AC SE setups. The discrete wavelet transform (DWT) algorithm is used for the RNN model. The main goal of DWT is to extract the hidden time-frequency domain characteristics and features at every specific time. The proposed model is able to leverage dynamic temporal and spatial features for attack detection. The authors detect FDIAs in AC SE with complete and incomplete system knowledge. For the incomplete knowledge case, the attacker only knew a few selected phase angles, power flows, and power injections for selected buses. The remaining buses information is generated using the algebraic sum of the connecting buses. The RNN-based detection model is constructed with two types of neuron layers: gated recurrent unit and fully-connected dense layers. To tune the network hyper parameters, a dropout approach is used. In dropout, the outputs of some layers are discarded according to predefined probabilities. Dropout solves the over fitting problem, eminent in extensive training datasets, and increases the accuracy for newly added test data. The proposed detection method's performance is assessed on the IEEE 118 bus and 300 bus test cases. Over 200k samples are generated to train the detection model, and which achieves a high detection rate of 93% [107]. However, the main challenge of this RNN type is to optimally tune the network hyper parameters.

Deep belief networks (DBNs) consist of multiple layers of stochastic and latent variables [127]. The latent variables are generally binary variables. DBNs are compositions of simple, unsupervised networks such as restricted Boltzmann machines or autoencoders [127]. The authors in [128] utilised autoencoder networks for the detection of FDIA leveraging temporal and spatial sensor data correlations. He *et al.* [108] proposed a DBN and state vector estimator (SVE) for real-time detection of FDIAs. The proposed model utilises an extended DBN called conditional DBN which extracts temporal features in high-dimensions. SVE calculates the ℓ_2-norm of the measurement residuals and compares them with a given threshold as follows [108]:

$$\begin{cases} \boldsymbol{r} = \| \hat{\boldsymbol{z}} - \boldsymbol{H}\hat{\boldsymbol{x}} \|_2 > \tau, & \text{Attack alarm} \\ \boldsymbol{r} = \| \hat{\boldsymbol{z}} - \boldsymbol{H}\hat{\boldsymbol{x}} \|_2 \leq \tau, & \text{No attack alarm} \end{cases} \tag{66}$$

The authors design the model based on the assumption that the topology of the power system does not change significantly within a small time-frame. For the simulation, the IEEE 118 bus test case is used to simulate four different attack scenarios. ROC curve is used to evaluate the detection scheme. Then, the detection results with a different number of attacked measurement k are compared with other detection algorithms such as MLPs and SVMs. The proposed model achieves the highest detection accuracy. However, training a DBN is extremely computationally expensive, since this process can take up to weeks even if specialised hardware exploiting graphics processing unit acceleration is used [129].

Wei *et al.* [109] proposed a different DBN-based model, where the detection process can be divided into three parts: (i) the data pre-processing stage, (ii) the training stage, and (iii) the testing stage. During the pre-processing data stage, measurement data including the attacked measurements are extracted using different IEEE standard nodes. The training process of the DBN is divided into the pre-training stage and reverse-trimming stage. In the pre-training stage, the authors use an unsupervised greedy learning algorithm from the bottom layer to the upper layer to extract the measurement features, train every layer, and share the measurement features with every layer. The Restricted Boltzmann Machine (RBM) is trained layer-by-layer and tuned using back-propagation to minimise prediction errors. After the training process, part of the measurement data is used to test and validate the performance of the detection model. The simulation results show that the DBN-based detection achieves high accuracy in detecting FDIAs (98%).

6 Discussion on the detection performance of machine learning algorithms

This section discusses the performance of the machine learning-based FDIA detection algorithms presented in Section 5. A noteworthy advantage of such detection algorithms is that they do not assume exact knowledge of the power system model nor its corresponding parameters, thus any induced uncertainties, e.g. measurement noise, topology changes, power flow perturbations etc. do not affect the algorithm's detection efficacy.

The majority of FDIA detection research focuses on the transmission level and studies that examine detection algorithms involving automatic generation control and wind generation have been reported [101, 130, 131]. On the other hand, studies that examine FDIAs detection for DSs are also essential and a direction

of on-going research [61]. Our investigation suggests that FDIAs studies can be broadly classified under two major categories, (i) random FDIAs, where the attacker aims to inject falsified attack vectors and compromise the SE algorithm by modifying any measurement vector that can be attained and (ii) targeted FDIAs, in which the attacker objective is to inject specific errors into the SE algorithms by maliciously modifying distinct measurement vectors. Apart from the aforementioned FDIAs types, studies involving stealthy FDIA detection have also been proposed [97].

The detection accuracy of the machine learning FDIA algorithms yields significantly different results depending on the setup used to evaluate the algorithm's performance. For example, some studies – to characterise the detection performance – examine algorithms under hundreds of different FDIA scenarios and varying power system topologies, while others report results based on very limited datasets. Notably, the algorithms presented in [31, 103, 107] are thoroughly tested on multiple power system architectures, such as IEEE 14, 30, and 118 bus systems, contrary to the algorithms in [57, 99, 102], which utilise only one IEEE system model during the performance analysis. Additionally, some papers consider basic FDIAs while others evaluate detection performance against stealthy FDIAs, which significantly skews the algorithm efficacy [97, 125]. Finally, a number of researchers develop their custom metrics to assess the proposed detection algorithms or do not provide any quantitative results whatsoever. For all the aforementioned reasons, providing an overarching algorithm comparison or declaring an optimal detection algorithm for every case is infeasible, since detection performance is contingent upon a multitude of reasons (e.g. TS or DS, stealthy or basic FDIAs, size of the system under test etc.) and comprehensive results are not available in the literature.

SVMs are consistently more effective in detecting FDIAs in power systems with reported detection rates ranging from 85 to 99% [31, 33, 97]. Contrary to supervised learning detection methods, SVMs do not require exhaustive training and big data sets which increase computational complexity and training duration [132]. On the other hand, SVMs performance can degrade significantly if the kernel selection process is not properly conducted or when we deal with sparse systems [133, 134]. Another drawback of SVMs, which has been recently reported and can effectively lower their detection accuracy, is that their susceptibility to adversarial examples [135]. Adversarial examples are carefully crafted inputs intentionally designed to falsify machine learning algorithms [136]. For instance, label flipped attacks are a form of adversarial example, which targets SVMs and affect their detection competency against FDIAs in power systems [135, 137].

Apart from SVMs, deep learning algorithms have been proposed for the detection of different types of FDIAs (e.g. stealthy or basic) and in different power system topologies (i.e. TS or DS). Contrary to SVMs, deep learning methodologies require large amounts of training data and their detection efficiency is heavily affected by the dimension of the training dataset. Multiple works report detection rates between 90 and 99% when abundance of training data is available for the deep learning detectors [98, 99, 103–107]. Despite the impressive results that deep learning algorithms exhibit, their training process requires an excessive amount of time, has high-computational costs and demands specialised equipment, in addition to big datasets. For instance, the authors in [108] reported that more than $3k$ measurement samples are essential in order for their deep learning algorithm to achieve detection rates of 98%.

Previous works prove that machine learning algorithms including supervised learning, SVMs, and deep learning method are able to effectively and in real-time detect FDIAs in power systems [108, 122]. The main pitfall of machine learning approaches is that they require extensive data sets and historical data including attack scenarios to train the detectors [138], which causes all the aforementioned disadvantages [31]. Besides the exploitation of resources, e.g. memory, storage space, specialised hardware etc. overfitting is another vulnerability that machine learning algorithms suffer from. By overfitting a machine learning algorithm we end up with a detector that is able to perform exceptionally well for specific datasets but cannot generalise this performance for all possible test cases, thus even selecting a proper training set becomes challenging [139]. Adversarial examples can also compromise machine learning-based algorithms. Limited research works to address this issue [135, 137, 140, 141], thus developing robust machine learning detectors against adversarial examples is imperative and one of our future directions.

7 Conclusions and future directions

Improving the cybersecurity of cyber-physical energy systems is vital for the efficient and resilient operation of the power grid. FDIAs can elicit severe physical and economic impacts on power systems. Researchers have thoroughly investigated FDIAs and have proposed algorithms to detect these data integrity attacks. Among these algorithms, machine learning-based methodologies are gaining attention due to their superior detection performance.

In this paper, we provide a comprehensive review of various FDIA detection methods leveraging machine learning algorithms. The goal of this survey is to compare different machine learning FDIA detectors employed in power systems. Our investigation concludes that supervised learning and deep learning methods achieve the highest detection rates. Our future work will explore how machine learning-based FDIA detectors perform in DS which incorporate DERs (e.g. microgrids) and what modifications are essential. Also, we aim to develop detection algorithms leveraging generative adversarial networks to further improve FDIAs detection performance against stealthy and more sophisticated attacks.

8 References

[1] Kline, R.R., Lassman, T.C.: 'Competing research traditions in American industry: uncertain alliances between engineering and science at Westinghouse Electric, 1886–1935', *Enterp. Soc.*, 2005, **6**, (4), pp. 601–645

[2] McLaughlin, S., Konstantinou, C., Wang, X., *et al.*: 'The cybersecurity landscape in industrial control systems', *Proc. IEEE*, 2016, **104**, (5), pp. 1039–1057

[3] Hahn, A., Govindarasu, M.: 'Cyber attack exposure evaluation framework for the smart grid', *IEEE Trans. Smart Grid*, 2011, **2**, (4), pp. 835–843

[4] Assante, M.J.: 'Confirmation of a coordinated attack on the Ukrainian power grid'. Available at https://ics.sans.org/blog/2016/01/09/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid, accessed July 2019

[5] BBC: 'Ukraine power cut 'was cyber-attack''. Available at https://www.bbc.com/news/technology-38573074, accessed July 2019

[6] Liu, X., Konstantinou, C.: 'Reinforcement learning for cyber-physical security assessment of power systems'. 2019 IEEE Milan PowerTech, Milano, Italy, 2019, pp. 1–6

[7] Keliris, A., Konstantinou, C., Sazos, M., *et al.*: 'Open source intelligence for energy sector cyberattacks', in '*Critical infrastructure security and resilience*' (Springer, Germany, 2019), pp. 261–281

[8] Shweppe, J., Rom, D.: 'Power system static state estimation: part I, II, and III'. Power Industry Computer Conf., Denver, CO, USA, 1969

[9] Bandak, C.E.: 'Power systems state estimation', IEEE, 2014

[10] Monticelli, A., Garcia, A.: 'Reliable bad data processing for real-time state estimation', *IEEE Trans. Power Appar. Syst.*, 1983, **PAS-102**, (5), pp. 1126–1139

[11] Zakerian, A., Maleki, A., Mohammadnian, Y., *et al.*: 'Bad data detection in state estimation using decision tree technique'. 2017 Iranian Conf. on Electrical Engineering (ICEE), Tehran, Iran, 2017, pp. 1037–1042

[12] Liu, Y., Ning, P., Reiter, M.K.: 'False data injection attacks against state estimation in electric power grids', *ACM Trans. Inf. Syst. Secur.*, 2011, **14**, (1), p. 13

[13] Musleh, A.S., Chen, G., Dong, Z.Y.: 'A survey on the detection algorithms for false data injection attacks in smart grids', *IEEE Trans. Smart Grid*, 2020, **11**, (3), pp. 2218–2234

[14] Zhang, M., Shen, C., He, N., *et al.*: 'False data injection attacks against smart gird state estimation: construction, detection and defense', *Sci. China Technol. Sci.*, 2019, **62**, pp. 2077–2087

[15] Aoufi, S., Derhab, A., Guerroumi, M.: 'Survey of false data injection in smart power grid: attacks, countermeasures and challenges', *J. Inf. Secur. Appl.*, 2020, **54**, p. 102518

[16] Majumdar, A., Pal, B.C.: 'Bad data detection in the context of leverage point attacks in modern power networks', *IEEE Trans. Smart Grid*, 2016, **9**, (3), pp. 2042–2054

[17] Bobba, R.B., Rogers, K.M., Wang, Q., *et al.*: 'Detecting false data injection attacks on dc state estimation'. Preprints of the First Workshop on Secure Control Systems, CPSWEEK, Stockholm, Sweden, 2010, vol. 2010

[18] Liu, L., Esmalifalak, M., Ding, Q., *et al.*: 'Detecting false data injection attacks on power grid by sparse optimization', *IEEE Trans. Smart Grid*, 2014, **5**, (2), pp. 612–621

[19] Chaojun, G., Jirutitijaroen, P., Motani, M.: 'Detecting false data injection attacks in ac state estimation', *IEEE Trans. Smart Grid*, 2015, **6**, (5), pp. 2476–2483

[20] Li, B., Ding, T., Huang, C., *et al.*: 'Detecting false data injection attacks against power system state estimation with fast godecomposition (godec) approach', *IEEE Trans. Ind. Inf.*, 2019, **15**, (5), pp. 2892–2904

[21] Živković, N., Sarić, A.T.: 'Detection of false data injection attacks using unscented Kalman filter', *J. Mod. Power Syst. Clean Energy*, 2018, **6**, (5), pp. 847–859

[22] Kosut, O., Jia, L., Thomas, R.J., *et al.*: 'Limiting false data attacks on power system state estimation'. 2010 44th Annual Conf. on Information Sciences and Systems (CISS), Princeton, NJ, USA, 2010, pp. 1–6

[23] Kosut, O., Jia, L., Thomas, R.J., *et al.*: 'Malicious data attacks on smart grid state estimation: attack strategies and countermeasures'. 2010 First IEEE Int. Conf. on Smart Grid Communications, Gaithersburg, MD, USA, 2010, pp. 220–225

[24] Li, S., Ylmaz, Y., Wang, X.: 'Quickest detection of false data injection attack in wide-area smart grids', *IEEE Trans. Smart Grid*, 2015, **6**, (6), pp. 2725–2735

[25] Rawat, D.B., Bajracharya, C.: 'Detection of false data injection attacks in smart grid communication systems', *IEEE Signal Process. Lett.*, 2015, **22**, (10), pp. 1652–1656

[26] Singh, S.K., Khanna, K., Bose, R., *et al.*: 'Joint-transformation-based detection of false data injection attacks in smart grid', *IEEE Trans. Ind. Inf.*, 2018, **14**, (1), pp. 89–97

[27] Zhao, J., Mili, L.: 'Vulnerability of the largest normalized residual statistical test to leverage points', *IEEE Trans. Power Syst.*, 2018, **33**, (4), pp. 4643–4646

[28] Zhang, Y., Wang, L., Sun, W., *et al.*: 'Distributed intrusion detection system in a multi-layer network architecture of smart grids', *IEEE Trans. Smart Grid*, 2011, **2**, (4), pp. 796–808

[29] Anderson, R.N., Boulanger, A., Powell, W.B., *et al.*: 'Adaptive stochastic control for the smart grid', *Proc. IEEE*, 2011, **99**, (6), pp. 1098–1115

[30] Rudin, C., Waltz, D., Anderson, R.N., *et al.*: 'Machine learning for the New York City power grid', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012, **34**, (2), p. 328

[31] Ozay, M., Esnaola, I., Vural, F.T.Y., *et al.*: 'Machine learning methods for attack detection in the smart grid', *IEEE Trans. Neural Netw. Learn. Syst.*, 2016, **27**, (8), pp. 1773–1786

[32] Esmalifalak, M., Nguyen, H., Zheng, R., *et al.*: 'Stealth false data injection using independent component analysis in smart grid'. 2011 IEEE Int. Conf. on Smart Grid Communications (SmartGridComm), Brussels, Belgium, 2011, pp. 244–248

[33] Yan, J., Tang, B., He, H.: 'Detection of false data attacks in smart grid with supervised learning'. 2016 Int. Joint Conf. on Neural Networks (IJCNN), Vancouver, Canada, 2016, pp. 1395–1402

[34] Wilson, D., Tang, Y., Yan, J., *et al.*: 'Deep learning-aided cyber-attack detection in power transmission systems'. 2018 IEEE Power & Energy Society General Meeting (PESGM), Portland, OR, USA, 2018, pp. 1–5

[35] Soares, T.M., Bezerra, U.H., Tostes, M.E.d.L.: 'Full-observable three-phase state estimation algorithm applied to electric distribution grids', *Energies*, 2019, **12**, (7), p. 1327

[36] Singh, A.K., Pal, B.C.: 'Decentralized dynamic state estimation in power systems using unscented transformation', *IEEE Trans. Power Syst.*, 2014, **29**, (2), pp. 794–804

[37] Ghahremani, E., Kamwa, I.: 'Dynamic state estimation in power system by applying the extended Kalman filter with unknown inputs to phasor measurements', *IEEE Trans. Power Syst.*, 2011, **26**, (4), pp. 2556–2566

[38] Baran, M.E., Kelley, A.W.: 'State estimation for real-time monitoring of distribution systems', *IEEE Trans. Power Syst.*, 1994, **9**, (3), pp. 1601–1609

[39] Wang, S., Gao, W., Meliopoulos, A.P.S.: 'An alternative method for power system dynamic state estimation based on unscented transform', *IEEE Trans. Power Syst.*, 2012, **27**, (2), pp. 942–950

[40] Valverde, G., Terzija, V.: 'Unscented Kalman filter for power system dynamic state estimation', *IET Gener. Transm. Distrib.*, 2011, **5**, (1), pp. 29–37

[41] Haughton, D.A., Heydt, G.T.: 'A linear state estimation formulation for smart distribution systems', *IEEE Trans. Power Syst.*, 2012, **28**, (2), pp. 1187–1195

[42] Teng, J.-H.: 'Using voltage measurements to improve the results of branch-current-based state estimators for distribution systems', *IEE Proc. Gener. Transm. Distrib.*, 2002, **149**, (6), pp. 667–672

[43] Majumdar, A., Pal, B.C.: 'A three-phase state estimation in unbalanced distribution networks with switch modelling'. 2016 IEEE First Int. Conf. on Control, Measurement and Instrumentation (CMI), Amitava Chatterjee, India, 2016, pp. 474–478

[44] Monticelli, A.: 'Electric power system state estimation', *Proc. IEEE*, 2000, **88**, (2), pp. 262–282

[45] Gao, Y., Yu, N.: 'State estimation for unbalanced electric power distribution systems using AMI data'. 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conf. (ISGT), Arlington, VA, USA, 2017, pp. 1–5

[46] Monticelli, A.: '*State estimation in electric power systems: a generalized approach*' (Springer Science & Business Media, USA, 2012)

[47] Jain, A., Shivakumar, N.R.: 'Power system tracking and dynamic state estimation'. 2009 IEEE/PES Power Systems Conf. and Exposition, Seattle, WA, USA, 2009, pp. 1–8

[48] Shivakumar, N.R., Jain, A.: 'A review of power system dynamic state estimation techniques'. 2008 Joint Int. Conf. on Power System Technology and IEEE Power India Conf., New Delhi, India, 2008, pp. 1–6

[49] Merrill, H.M., Schweppe, F.C.: 'Bad data suppression in power system static state estimation', *IEEE Trans. Power Appar. Syst.*, 1971, **PAS-90**, (6), pp. 2718–2725

[50] Handschin, E., Schweppe, F.C., Kohlas, J., *et al.*: 'Bad data analysis for power system state estimation', *IEEE Trans. Power Appar. Syst.*, 1975, **94**, (2), pp. 329–337

[51] Zhao, J., Zhang, G., La Scala, M., *et al.*: 'Enhanced robustness of state estimator to bad data processing through multi-innovation analysis', *IEEE Trans. Ind. Inf.*, 2016, **13**, (4), pp. 1610–1619

[52] Lin, Y., Abur, A.: 'A highly efficient bad data identification approach for very large scale power systems', *IEEE Trans. Power Syst.*, 2018, **33**, (6), pp. 5979–5989

[53] Aghamolki, H.G., Miao, Z., Fan, L.: 'SOCP convex relaxation-based simultaneous state estimation and bad data identification', arXiv preprint arXiv:1804.05130, 2018

[54] Van Cutsem, T., Ribbens-Pavella, M., Mili, L.: 'Bad data identification methods in power system state estimation–a comparative study', *IEEE Trans. Power Appar. Syst.*, 1985, **PAS-104**, (11), pp. 3037–3049

[55] Dan, G., Sandberg, H.: 'Stealth attacks and protection schemes for state estimators in power systems'. 2010 First IEEE Int. Conf. on Smart Grid Communications (SmartGridComm), Gaithersburg, MD, USA, 2010, pp. 214–219

[56] Liang, G., Weller, S.R., Zhao, J., *et al.*: 'False data injection attacks targeting dc model-based state estimation'. 2017 IEEE Power & Energy Society General Meeting, Chicago, IL, USA, 2017, pp. 1–5

[57] Teixeira, A., Dán, G., Sandberg, H., *et al.*: 'A cyber security study of a SCADA energy management system: stealthy deception attacks on the state estimator', *IFAC Proc. Vol.*, 2011, **44**, (1), pp. 11271–11277

[58] Hug, G., Giampapa, J.A.: 'Vulnerability assessment of ac state estimation with respect to false data injection cyber-attacks', *IEEE Trans. Smart Grid*, 2012, **3**, (3), pp. 1362–1370

[59] Deng, R., Xiao, G., Lu, R., *et al.*: 'False data injection on state estimation in power systems – attacks, impacts, and defense: A survey', *IEEE Trans. Ind. Inf.*, 2016, **13**, (2), pp. 411–423

[60] Choeum, D., Choi, D.-H.: 'OLTC-induced false data injection attack on volt/VAR optimization in distribution systems', *IEEE Access*, 2019, **7**, pp. 34508–34520

[61] Deng, R., Zhuang, P., Liang, H.: 'False data injection attacks against state estimation in power distribution systems', *IEEE Trans. Smart Grid*, 2018, **10**, (3), pp. 2871–2881

[62] Zhang, H., Meng, W., Qi, J., *et al.*: 'False data injection attacks on inverter-based microgrid in autonomous mode', in '*Distributed control methods and cyber security issues in microgrids*' (Elsevier, Netherlands, 2020), pp. 125–146

[63] Sou, K.C., Sandberg, H., Johansson, K.H.: 'Electric power network security analysis via minimum cut relaxation'. 2011 50th IEEE Conf. on Decision and Control and European Control Conf. (CDC-ECC), Orlando, FL, USA, 2011, pp. 4054–4059

[64] Mallat, S., Zhang, Z.: 'Matching pursuit with time-frequency dictionaries'. Tech. Rep., Courant Institute of Mathematical Sciences, New York, United States, 1993

[65] Rahman, M.A., Mohsenian-Rad, H.: 'False data injection attacks with incomplete information against smart power grids'. 2012 IEEE Global Communications Conf. (GLOBECOM), Anaheim, CA, USA, 2012, pp. 3153–3158

[66] Kekatos, V., Giannakis, G.B., Baldick, R.: 'Grid topology identification using electricity prices'. 2014 IEEE PES General Meeting| Conf. & Exposition, Washington, DC, USA, 2014, pp. 1–5

[67] Kim, J., Tong, L., Thomas, R.J.: 'Subspace methods for data attack on state estimation: a data driven approach', *IEEE Trans. Signal Process.*, 2015, **63**, (5), pp. 1102–1114

[68] Yu, Z.-H., Chin, W.-L.: 'Blind false data injection attack using PCA approximation method in smart grid', *IEEE Trans. Smart Grid*, 2015, **6**, (3), pp. 1219–1226

[69] Liu, X., Bao, Z., Lu, D., *et al.*: 'Modeling of local false data injection attacks with reduced network information', *IEEE Trans. Smart Grid*, 2015, **6**, (4), pp. 1686–1696

[70] Yuan, Y., Li, Z., Ren, K.: 'Modeling load redistribution attacks in power systems', *IEEE Trans. Smart Grid*, 2011, **2**, (2), pp. 382–390

[71] Liang, G., Zhao, J., Luo, F., *et al.*: 'A review of false data injection attacks against modern power systems', *IEEE Trans. Smart Grid*, 2017, **8**, (4), pp. 1630–1638

[72] Rahman, M.A., Al-Shaer, E., Kavasseri, R.: 'Impact analysis of topology poisoning attacks on economic operation of the smart power grid'. 2014 IEEE 34th Int. Conf. on Distributed Computing Systems (ICDCS), Madrid, Spain, 2014, pp. 649–659

[73] Lin, J., Yu, W., Yang, X., *et al.*: 'On false data injection attacks against distributed energy routing in smart grid'. Proc. 2012 IEEE/ACM Third Int. Conf. on Cyber-Physical Systems, Beijing, People's Republic of China, 2012, pp. 183–192

[74] Xie, L., Mo, Y., Sinopoli, B.: 'False data injection attacks in electricity markets'. 2010 First IEEE Int. Conf. on Smart Grid Communications (SmartGridComm), Gaithersburg, MD, USA, 2010, pp. 226–231

[75] Xie, L., Mo, Y., Sinopoli, B.: 'Integrity data attacks in power market operations', *IEEE Trans. Smart Grid*, 2011, **2**, (4), pp. 659–666

[76] Anubi, O.M., Konstantinou, C.: 'Enhanced resilient state estimation using data-driven auxiliary models', *IEEE Trans. Ind. Inf.*, 2020, **16**, (1), pp. 639–647

[77] Anubi, O.M., Konstantinou, C., Roberts, R.: 'Resilient optimal estimation using measurement prior', arXiv preprint arXiv:1907.13102, 2019

[78] Kosut, O., Jia, L., Thomas, R.J., *et al.*: 'Malicious data attacks on the smart grid', *IEEE Trans. Smart Grid*, 2011, **2**, (4), pp. 645–658

[79] Bi, S., Zhang, Y.J.: 'Graphical methods for defense against false-data injection attacks on power system state estimation', *IEEE Trans. Smart Grid*, 2014, **5**, (3), pp. 1216–1227

[80] Mishra, S., Li, X., Pan, T., *et al.*: 'Price modification attack and protection scheme in smart grid', *IEEE Trans. Smart Grid*, 2017, **8**, (4), pp. 1864–1875

[81] Chen, J., Abur, A.: 'Placement of PMUs to enable bad data detection in state estimation', *IEEE Trans. Power Syst.*, 2006, **21**, (4), pp. 1608–1615
[82] Kim, T.T., Poor, H.V.: 'Strategic protection against data injection attacks on power grids', *IEEE Trans. Smart Grid*, 2011, **2**, (2), pp. 326–333
[83] Wei, L., Sarwat, A.I., Saad, W., *et al.*: 'Stochastic games for power grid protection against coordinated cyber-physical attacks', *IEEE Trans. Smart Grid*, 2018, **9**, (2), pp. 684–694
[84] Ma, C.Y., Yau, D.K., Lou, X., *et al.*: 'Markov game analysis for attack-defense of power networks under possible misinformation', *IEEE Trans. Power Syst.*, 2013, **28**, (2), pp. 1676–1686
[85] Wang, C., Hou, Y., Ten, C.-W.: 'Determination of Nash equilibrium based on plausible attack-defense dynamics', *IEEE Trans. Power Syst.*, 2017, **32**, (5), pp. 3670–3680
[86] Sanjab, A., Saad, W.: 'Data injection attacks on smart grids with multiple adversaries: a game-theoretic perspective', *IEEE Trans. Smart Grid*, 2016, **7**, (4), pp. 2038–2049
[87] Sun, Y., Mao, Y., Liu, T., *et al.*: 'A dynamic secret-based encryption method in smart grids wireless communication'. IEEE PES Innovative Smart Grid Technologies, Washington, DC, USA, 2012, pp. 1–5
[88] Manandhar, K., Cao, X., Hu, F., *et al.*: 'Combating false data injection attacks in smart grid using Kalman filter'. 2014 Int. Conf. on Computing, Networking and Communications (ICNC), Honolulu, HI, USA, 2014, pp. 16–20
[89] Abdallah, A., Shen, X.S.: 'Efficient prevention technique for false data injection attack in smart grid'. 2016 IEEE Int. Conf. on Communications (ICC), Kuala Lumpur, Malaysia, 2016, pp. 1–6
[90] Shahid, M.A., Nawaz, R., Qureshi, I.M., *et al.*: 'Proposed defense topology against cyber attacks in smart grid'. 2018 Int. Conf. on Power Generation Systems and Renewable Energy Technologies (PGSRET), Islamabad, Pakistan, 2018, pp. 1–5
[91] Li, B., Lu, R., Xiao, G., *et al.*: 'PAMA: a proactive approach to mitigate false data injection attacks in smart grids'. 2018 IEEE Global Communications Conf. (GLOBECOM), Abu Dhabi, UAE, 2018, pp. 1–6
[92] Ansari, M.H., Vakili, V.T., Bahrak, B., *et al.*: 'Graph theoretical defense mechanisms against false data injection attacks in smart grids', *J. Mod. Power Syst. Clean Energy*, 2018, **6**, (5), pp. 860–871
[93] Konstantinou, C., Sazos, M., Musleh, A.S., *et al.*: 'GPS spoofing effect on phase angle monitoring and control in a real-time digital simulator-based hardware-in-the-loop environment', *IET Cyber-Phys. Syst., Theory Appl.*, 2017, **2**, (4), pp. 180–187
[94] Liang, X., Xiao, Y.: 'Game theory for network security', *IEEE Commun. Surv. Tutorials*, 2012, **15**, (1), pp. 472–486
[95] Zografopoulos, I., Konstantinou, C.: 'DERauth: a battery-based authentication scheme for distributed energy resources'. 2020 IEEE Computer Society Annual Symp. on VLSI (ISVLSI), Limassol, Cyprus, 2020
[96] Schuld, M., Sinayskiy, I., Petruccione, F.: 'An introduction to quantum machine learning', *Contemp. Phys.*, 2015, **56**, (2), pp. 172–185
[97] Esmalifalak, M., Liu, L., Nguyen, N., *et al.*: 'Detecting stealthy false data injection using machine learning in smart grid', *IEEE Syst. J.*, 2017, **11**, (3), pp. 1644–1652
[98] Ayad, A., Farag, H.E., Youssef, A., *et al.*: 'Detection of false data injection attacks in smart grids using recurrent neural networks', IEEE, 2018
[99] Foroutan, S.A., Salmasi, F.R.: 'Detection of false data injection attacks against state estimation in smart grids based on a mixture Gaussian distribution learning method', *IET Cyber-Phys. Syst., Theory Appl.*, 2017, **2**, (4), pp. 161–171
[100] Joachims, T.: 'Making large-scale SVM learning practical'. Technical Report, SFB 475: Komplexitätsreduktion in Multivariaten, 1998
[101] Mohammadpourfard, M., Sami, A., Weng, Y.: 'Identification of false data injection attacks with considering the impact of wind generation and topology reconfigurations', *IEEE Trans. Sustain. Energy*, 2017, **9**, (3), pp. 1349–1364
[102] Yang, C., Wang, Y., Zhou, Y., *et al.*: 'False data injection attacks detection in power system using machine learning method', *J. Comput. Commun.*, 2018, **6**, (11), pp. 276–286. Available at http://www.scirp.org/journal/doi.aspx?DOI=10.4236/jcc.2018.611025
[103] Mohammadpourfard, M., Sami, A., Seifi, A.R.: 'A statistical unsupervised method against false data injection attacks: a visualization-based approach', *Expert Syst. Appl.*, 2017, **84**, pp. 242–261
[104] Ganjkhani, M., Fallah, S.N., Badakhshan, S., *et al.*: 'A novel detection algorithm to identify false data injection attacks on power system state estimation', *Energies*, 2019, **12**, (11), p. 2209
[105] Tabakhpour, A., Abdelaziz, M.M.: 'Neural network model for false data detection in power system state estimation'. 2019 IEEE Canadian Conf. of Electrical and Computer Engineering (CCECE), Edmonton, AB, Canada, 2019, pp. 1–5
[106] Basumallik, S., Ma, R., Eftekharnejad, S.: 'Packet-data anomaly detection in PMU-based state estimator using convolutional neural network', *Int. J. Electr. Power Energy Syst.*, 2019, **107**, pp. 690–702
[107] James, J., Hou, Y., Li, V.O.: 'Online false data injection attack detection with wavelet transform and deep neural networks', *IEEE Trans. Ind. Inf.*, 2018, **14**, (7), pp. 3271–3280
[108] He, Y., Mendis, G.J., Wei, J.: 'Real-time detection of false data injection attacks in smart grid: a deep learning-based intelligent mechanism', *IEEE Trans. Smart Grid*, 2017, **8**, (5), pp. 2505–2516
[109] Wei, L., Gao, D., Luo, C.: 'False data injection attacks detection with deep belief networks in smart grid'. 2018 Chinese Automation Congress (CAC), Hangzhou, People's Republic of China 2019, pp. 2621–2625
[110] Abdallah, A., Shen, X.: '*Security and privacy in smart grid*' (Springer, Germany, 2018)
[111] Koehrsen, W.: 'Beyond accuracy: precision and recall', March 2018. Available at https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c
[112] Stephen, I.: 'Perceptron-based learning algorithms', *IEEE Trans. Neural Netw.*, 1990, **50**, (2), p. 179
[113] Cortes, C., Vapnik, V.: 'Support-vector networks', *Mach. Learn.*, 1995, **20**, (3), pp. 273–297
[114] Altman, N.S.: 'An introduction to kernel and nearest-neighbor nonparametric regression', *Am. Stat.*, 1992, **46**, (3), pp. 175–185
[115] Hosmer Jr, D.W., Lemeshow, S., *et al.*: '*Applied logistic regression*', vol. **398** (John Wiley & Sons, USA, 2013)
[116] Wang, Q., Kulkarni, S.R., Verdú, S.: 'Divergence estimation for multidimensional densities via k-nearest-neighbor distances', *IEEE Trans. Inf. Theory*, 2009, **55**, (5), pp. 2392–2405
[117] Abe, S.: 'Feature selection and extraction', in '*Support vector machines for pattern classification*' (Springer, Germany, 2010), pp. 331–341
[118] Cramer, J.S.: 'The origins of logistic regression', *IEEE Trans. Autom. Control*, 2002, **4**
[119] Chapelle, O., Sindhwani, V., Keerthi, S.S.: 'Optimization techniques for semi-supervised support vector machines', *J. Mach. Learn. Res.*, 2008, **9**, pp. 203–233
[120] Filho, G.C.: 'Mixture models for the analysis of gene expression', PhD thesis, Freie Universität Berlin, 2008
[121] Hendrickx, J.M., Johansson, K.H., Jungers, R.M., *et al.*: 'Efficient computations of a security index for false data attacks in power networks', *IEEE Trans. Autom. Control*, 2014, **59**, (12), pp. 3194–3208
[122] Konstantinou, C., Maniatakos, M.: 'A data-based detection method against false data injection attacks', *IEEE Design Test*, 2019, pp. 1–1, Early Access
[123] Hassabis, D., Kumaran, D., Summerfield, C., *et al.*: 'Neuroscience-inspired artificial intelligence', *Neuron*, 2017, **95**, (2), pp. 245–258
[124] Karn, U.: 'A quick Introduction to neural networks', August 2016. Available at https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/
[125] Ashrafuzzaman, M., Chakhchoukh, Y., Jillepalli, A.A., *et al.*: 'Detecting stealthy false data injection attacks in power grids using deep learning'. 2018 14th Int. Wireless Communications & Mobile Computing Conf. (IWCMC), Limassol, Cyprus, 2018, pp. 219–225
[126] Hochreiter, S., Schmidhuber, J.: 'Long short-term memory', *Neural Comput.*, 1997, **9**, (8), pp. 1735–1780
[127] Hinton, G.E.: 'Deep belief networks', *Scholarpedia*, 2009, **4**, (5), p. 5947, revision #91189
[128] Aboelwafa, M.M.N., Seddik, K.G., Eldefrawy, M.H., *et al.*: 'A machine learning-based technique for false data injection attacks detection in industrial IoT', *IEEE Internet Things J.*, 2020, **7**, (9), pp. 1–1
[129] Sarikaya, R., Hinton, G.E., Deoras, A.: 'Application of deep belief networks for natural language understanding', *IEEE/ACM Trans. Audio Speech, Lang. Process.*, 2014, **22**, (4), pp. 778–784
[130] Beg, O.A., Johnson, T.T., Davoudi, A.: 'Detection of false-data injection attacks in cyber-physical dc microgrids', *IEEE Trans. Ind. Inf.*, 2017, **13**, (5), pp. 2693–2703
[131] Tan, R., Nguyen, H.H., Foo, E.Y., *et al.*: 'Modeling and mitigating impact of false data injection attacks on automatic generation control', *IEEE Trans. Inf. Forensics Secur.*, 2017, **12**, (7), pp. 1609–1624
[132] Zhou, Z.-H.: 'A brief introduction to weakly supervised learning', *Nat. Sci. Rev.*, 2018, **5**, (1), pp. 44–53
[133] Nayak, J., Naik, B., Behera, H.: 'A comprehensive survey on support vector machine in data mining tasks: applications & challenges', *Int. J. Database Theory Appl.*, 2015, **8**, (1), pp. 169–186
[134] Chen, W., Ma, C., Ma, L.: 'Mining the customer credit using hybrid support vector machine technique', *Expert Syst. Appl.*, 2009, **36**, (4), pp. 7611–7616
[135] Sayghe, A., Anubi, O.M., Konstantinou, C.: 'Adversarial examples on power systems state estimation'. 2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conf. (ISGT), Washington, DC, USA, 2020, pp. 1–5
[136] Goodfellow, I.J., Shlens, J., Szegedy, C.: 'Explaining and harnessing adversarial examples', arXiv preprint arXiv:1412.6572, 2014
[137] Sayghe, A., Zhao, J., Konstantinou, C.: 'Evasion attacks with adversarial deep learning against power system state Virtual Conference'. IEEE Power & Energy Society General Meeting (PESGM), 2020, pp. 1–5
[138] Al-Jarrah, O.Y., Yoo, P.D., Muhaidat, S., *et al.*: 'Efficient machine learning for big data: a review', *Big Data Res.*, 2015, **2**, (3), pp. 87–93
[139] Dietterich, T.: 'Overfitting and undercomputing in machine learning', *ACM Comput. Surv.*, 1995, **27**, (3), pp. 326–327
[140] Tian, J., Li, T., Shang, F., *et al.*: 'Adaptive normalized attacks for learning adversarial attacks and defenses in power systems'. 2019 IEEE Int. Conf. on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Beijing, People's Republic of China, 2019, pp. 1–6
[141] Chen, Y., Tan, Y., Deka, D.: 'Is machine learning in power systems vulnerable?'. 2018 IEEE Int. Conf. on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Aalborg, Denmark, 2018, pp. 1–6