frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 24 January 2023
DOI 10.3389/fenrg.2022.1104989

OPEN ACCESS

EDITED BY
Yongming Han,
Beijing University of Chemical Technology, China

REVIEWED BY
Yu Ma,
Chang'an University, China
Zhijiang Chen,
Frostburg State University, United States
Yalong Wu,
University of Houston–Clear Lake, United States

*CORRESPONDENCE
Dou An,
douan2017@xjtu.edu.cn

SPECIALTY SECTION
This article was submitted to Process and Energy Systems Engineering, a section of the journal Frontiers in Energy Research

RECEIVED 22 November 2022
ACCEPTED 05 December 2022
PUBLISHED 24 January 2023

CITATION
Lin X, An D, Cui F and Zhang F (2023), False data injection attack in smart grid: Attack model and reinforcement learning-based detection method.
Front. Energy Res. 10:1104989.
doi: 10.3389/fenrg.2022.1104989

False data injection attack in smart grid: Attack model and reinforcement learning-based detection method

Xixiang Lin, Dou An*, Feifei Cui and Feiye Zhang

School of Automation Science and Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi, China

The smart grid, as a cyber-physical system, is vulnerable to attacks due to the diversified and open environment. The false data injection attack (FDIA) can threaten the grid security by constructing and injecting the falsified attack vector to bypass the system detection. Due to the diversity of attacks, it is impractical to detect FDIAs by fixed methods. This paper proposed a false data injection attack model and countering detection methods based on deep reinforcement learning (DRL). First, we studied an attack model under the assumption of unlimited attack resources and information of complete topology. Different types of FDIAs are also enumerated. Then, we formulated the attack detection problem as a Markov decision process (MDP). A deep reinforcement learning-based method is proposed to detect FDIAs with a combined dynamic-static detection mechanism. To address the sparse reward problem, experiences with discrepant rewards are stored in different replay buffers to achieve efficiency. Moreover, the state space is extended by considering the most recent states to improve the perception capability. Simulations were performed on IEEE 9,14,30, and 57-bus systems, proving the validation of attack model and efficiency of detection method. Results proved efficacy of the detection method in different scenarios.

KEYWORDS

state estimation, deep reinforcement learning, attack detection, smart grid, false data injection attack

1 Introduction

Smart grid is a representative cyber-physical system (Pasqualetti et al., 2013), permitting the bidirectional communication of both information and electric power between the utility and users. In the energy management system (EMS), state estimation plays a critical part of information-physical integration. Through state estimation, EMS can recognize the actual state of electricity transmission by filtering out possible noise to improve the reliability of the real-time data (Katiraei and Iravani, 2006).

Due to the decentralized and multi-temporal coupled characteristics of measurement devices (Annaswamy and Amin, 2013), the terminal equipment often lacks effective physical protection, resulting in the susceptibility to attacks. Diversified cyber-physical

system attacks have been proposed and among the attacks, FDIA is a representative one. The FDIA attacker constructs attack vectors through specific algorithms conditioned on the power grid topology information, injects them through weak points of the grid, and avoids being detected to damage data integrity (An et al., 2019). FDIAs can directly affect the state estimation and the subsequent control elements, causing the system to lose stability and even break down. A large number of smart grid security incidents have shown that, compared with traditional attacks, it is more difficult to detect and defend cyber-physical attacks (Liang et al., 2017).

Research efforts against FDIAs can be categorized into two main types: defense and detection. First, to defend against FDIAs before being attacked, researchers have studied the deployment of grid at the cyber-physical level, such as optimizing the distribution of key nodes and devices according to their coupling characteristics (Lei et al., 2020; Wu et al., 2021). Second, to detect FDIAs after attack, substantial efforts has been made, such as dynamic state estimation method (An et al., 2022), tracking the deviation of measurement (Alnowibet et al., 2021; Mohamed et al., 2021; Sinha et al., 2022) and some stochastic game methods (Wei et al., 2018; Oozeer and Haykin, 2019). In summary, above researches had shown that both cyber and physical methods are required in FDIA studies.

However, the grid operation is full of uncertainties, in which the system states and attacks are diverse (An and Liu, 2019). Due to the diversity, enumerating all attacks is not realistic with limited resources. Moreover, empirical or off-line attack detection strategies are not optimal solutions for online network attack detection (Ashok et al., 2018; Tsobdjou et al., 2022). Therefore, reinforcement learning (RL)-based methods are introduced to avoid the complexity of empirical methods and gain the ability of detecting attacks in multiple scenarios (Wang et al., 2018; Kurt et al., 2019; Haque et al., 2021).

Deep reinforcement learning (DRL) learns the optimum strategy of sequential decision problems by exploring and interacting with the environment. The agent gets rewards for guiding the behavior, with the goal of maximizing the long-term return (Sutton and Barto, 1998). DRL combines the feature-extraction capacity of neural networks with the decision-making capability of reinforcement learning in unknown environments to achieve direct control from state to action (Arulkumaran et al., 2017). The Deep Q-Network (DQN), used in conjunction with the replay buffer and a target network, is a representative DRL algorithm that can be adapted to environment with uncertainty (Mnih et al., 2015).

As for detection process of FDIA, after the unknown start of attack, state estimation results are falsified by the attack vector with unknown attack model (Kurt et al., 2019). Moreover, detection process of FDIA has the feature of sequential decision and the transition of state can be described as model-free (An et al., 2019, 2022). Thus, FIDA detection can be described as a MDP and trained utilizing DQN algorithm to achieve detection by neural networks.

The rest part is: In **Section 2**, related studies are reviewed. In **Section 3**, the smart grid state estimation is introduced, including the static and dynamic method. The empirical bad data detection is also introduced. In **Section 4**, an FDIA model is introduced and three types of FDIAs are discussed based on the attack model. In **Section 5**, a DRL-based, combined dynamic-static FDIA detection method is proposed and optimized. In **Section 6**, simulations of attacks and detections are performed on IEEE grid systems in multiple scenarios. In **Section 7**, this paper is concluded.

2 Related work

Weighted Least Squares (WLS) is the basic and widely-used method for power grid state estimation (Schweppe and Rom, 1970; Schweppe and Wildes, 1970). (Debs and Larson, 1970) applied Kalman filter (KF) to the power grid. As the study deepened, the extended Kalman filter (EKF) applied KF to non-linear systems. Moreover, unscented Kalman filter (UKF) and particle filtering (PF) were applied on state estimation, which improved the accuracy and stability of filtering (Wan and Van Der Merwe, 2000; Julier and Uhlmann, 2004).

(Liu et al., 2009) proposed FDIA and proved that the attack vector can bypass the detection element and cause damages on the system (Pang et al., 2016). studied attack method with the minimum cost to avoids anomaly detection (He et al., 2017). constructed a parallel FDIAs detection scheme, utilizing static-dynamic state estimation to detect attacks, which is robust (Li and Wang, 2019). investigated the method to construct a less costly and undetectable attack vector by partial topology information (Li et al., 2019). studied the selection of optimal buses during the attack and proposed a data-driven optimal bus attack method (Jiang et al., 2020). studied two types of FDIAs and proposed a detection-defense method (Chen and Wang, 2020). proposed a new state estimation method that estimates the grid state by sequential Monte Carlo filtering to detect multiple attacks.

Deep reinforcement learning matured later, but is widely used in sequential decision-making problems in recent years (Mnih et al., 2013). proposed DQN algorithm in 2013, and published a paper in 2015, in which DQN reached a high level over human players (Mnih et al., 2015). In the fields of smart grid security, (Wang et al., 2018), proposed an autonomous FDIA method adopting the nearest sequence memory Q-learning (Liu et al., 2020). investigated the vulnerability of power grids with new energy based on DRL (Wang et al., 2021). studied a hybrid cyber-physical topological attack in power grids, and proposed DRL-based method for detecting attacks with minimum cost (Luo and Xiao, 2021). proposed a FDIA method based on reinforcement learning (RL), utilizing measurements,

grid states and other parameters to construct attacks, without dependence on topology information.

As for RL-based FDIA detection, (Kurt et al., 2019), formulated the detection process as a MDP, and proposed a model-free RL-based detection scheme (Zhang and Wu, 2021). proposed a RL-based detection method without the attack model, utilizing a Q-table to detect attacks by Sarsa algorithm. To address the complexity of storing Q-table, (An et al., 2019; Sinha et al., 2022), applied DQN algorithm to detect FDIAs by neural networks. Moreover, (Alnowibet et al., 2021; Mohamed et al., 2021), studied FDIA detection on energy trading and energy management systems by intelligent priority selection-based RL method.

Researches have proved the efficacy of RL-based method in FIDA detection. However, most studies focused on detection against single attack model, and studies on different types of FIDAs are not sufficient. Moreover, few studies combined multiple state estimation methods in the detection scheme, bringing out the focus of this paper.

3 Preliminaries

In this section, static and dynamic state estimation algorithms are introduced, laying the foundation of combined dynamic-static FIDA detection mechanism. Bad data detection method is shown to verify the efficacy of FDIA. Basic information about DQN algorithm is also introduced.

3.1 Measurement equations

Relation between the measured power flows and state of the grid is (Schweppe and Wildes, 1970):

$$\begin{aligned} \boldsymbol{z} &= \boldsymbol{h}(\boldsymbol{x}) + \boldsymbol{v} \\ \boldsymbol{x} &= [\varphi_i, V_i]^T \\ \boldsymbol{z} &= [V_i, P_i, Q_i, P_{ij}, Q_{ij}]^T \end{aligned} \tag{1}$$

where, $\boldsymbol{z}$ denotes the system measurement, $\boldsymbol{h}$ denotes the measurement equation, $\boldsymbol{x}$ denotes the state of the grid, v denote the measurement noise, φ_i and V_i denote the voltage phase angle and magnitude of node i, P_i and Q_i denote the power of node i, P_{ij} and Q_{ij} denote the tributary power flow, from i to j, whose detailed equations are:

$$\begin{aligned} P_i &= \sum_{j\in N_i} V_i V_j \left(G_{ij} cos\left(\varphi_i - \varphi_j\right) + B_{ij} sin\left(\varphi_i - \varphi_j\right)\right) \\ Q_i &= \sum_{j\in N_i} V_i V_j \left(G_{ij} sin\left(\varphi_i - \varphi_j\right) - B_{ij} cos\left(\varphi_i - \varphi_j\right)\right) \\ P_{ij} &= V_i^2\left(g_{si} + g_{ij}\right) - V_i V_j cos\left(\varphi_i - \varphi_j\right) - V_i V_j b_{ij} sin\left(\varphi_i - \varphi_j\right) \\ Q_{ij} &= -V_i^2\left(b_{si} + b_{ij}\right) - V_i V_j sin\left(\varphi_i - \varphi_j\right) \\ &\quad - V_i V_j b_{ij} cos\left(\varphi_i - \varphi_j\right) \end{aligned} \tag{2}$$

where, G_{ij}, B_{ij} are the real and imaginary part of the i, j term in the node conduction matrix, g_{ij}, b_{ij} denote the conductance and susceptance between i, j, $g_{si} + jb_{si}$ denotes the conductance of the parallel branch of i, $g_{sj} + jb_{sj}$ denotes the conductance of the parallel branch of j.

3.2 Static state estimation

Due to the pervasive noise, measurements can be unreliable and inconsistent with the actual state. Static state estimation filters the noise based on the current-measured data (Schweppe and Wildes, 1970). Due to this factor, when attacked by FDIAs, results of static state estimation will deviate significantly from the true state. That deviation is utilized in attack detection mechanism of this paper. WLS method is adopted in this paper, whose iterative form is:

$$\begin{cases} \Delta \boldsymbol{z}^{(k)} = \boldsymbol{z} - \boldsymbol{h}\left(\hat{\boldsymbol{x}}^{(k)}\right) \\ \Delta \hat{\boldsymbol{x}}^{(k)} = \left[\boldsymbol{H}^T \boldsymbol{R}^{-1} \boldsymbol{H}\right]^{-1} \boldsymbol{H}^T \boldsymbol{R}^{-1} \Delta \boldsymbol{z}^{(k)} \\ \hat{\boldsymbol{x}}^{(k+1)} = \hat{\boldsymbol{x}}^{(k)} + \Delta \hat{\boldsymbol{x}}^{(k)} \end{cases} \tag{3}$$

where, k denotes the step of iteration, $\hat{\boldsymbol{x}}$ denotes the estimated state, $\boldsymbol{H}(\boldsymbol{x})$ denotes the Jacobian matrix of the measurement equation, with the dimension of $m \times n$. The iteration ends when $\Delta \hat{\boldsymbol{x}}$ is sufficiently small.

3.3 Dynamic state estimation

Dynamic state estimation bases on KF to estimate the state and eliminate noise. While static method focuses primarily on real-time states, dynamic method tries to predict the state of the next step and the estimation result is closer to true state under attacks. Due to the non-linearity of power system, EKF and UKF are applied in our works.

3.3.1 Extended kalman filter

EKF is effective for non-linear models, and performs better in systems with weak non-linearities and perturbations (Li et al., 2015).

A second order expansion of $\boldsymbol{h}(\hat{\boldsymbol{x}})$ around $\tilde{\boldsymbol{x}}$ is:

$$\boldsymbol{h}(\hat{\boldsymbol{x}}) = \boldsymbol{h}(\tilde{\boldsymbol{x}}) + \boldsymbol{H}(\tilde{\boldsymbol{x}})\Delta \boldsymbol{x} + \boldsymbol{S} \tag{4}$$

where, $\Delta \boldsymbol{x} = \hat{\boldsymbol{x}} - \tilde{\boldsymbol{x}}$, $\tilde{\boldsymbol{x}}$ is the prior estimation of state, $\hat{\boldsymbol{x}}$ is the posterior estimation of state, $\boldsymbol{H}$ denotes the Jacobian matrix of the measurement function, and $\boldsymbol{S}$ is the remainder term of the second and higher order. Omitting $\boldsymbol{S}$, a linearized model of grid

TABLE 1 Equations of EKF.

Step	Formula
Prior estimation	$\tilde{\boldsymbol{x}}_{k+1} = \boldsymbol{F}_k \hat{\boldsymbol{x}}_k$
	$\boldsymbol{M}_{k+1} = \boldsymbol{F}_k \boldsymbol{\Sigma}_k \boldsymbol{F}_k^T + \boldsymbol{Q}_k$
Kalman gain	$\boldsymbol{K}_{k+1} = \boldsymbol{M}_{k+1} \boldsymbol{H}_{k+1}^T \left(\boldsymbol{H}_{k+1} \boldsymbol{M}_{k+1} \boldsymbol{H}_{k+1}^T + \boldsymbol{R}_{k+1}\right)^{-1}$
Post estimation	$\hat{\boldsymbol{x}}_{k+1} = \tilde{\boldsymbol{x}}_{k+1} + \boldsymbol{K}_{k+1}\left(\boldsymbol{z} - \boldsymbol{h}\left(\tilde{\boldsymbol{x}}_{k+1}\right)\right)$
	$\boldsymbol{\Sigma}_{k+1} = \left(\boldsymbol{I} - \boldsymbol{K}_{k+1} \boldsymbol{H}_{k+1}\right) \boldsymbol{M}_{k+1}$

state is obtained.

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{Q}_k \\ \boldsymbol{z}_{k+1} &= \boldsymbol{H} \boldsymbol{x}_{k+1} + \boldsymbol{R}_k \end{aligned} \tag{5}$$

where, $\boldsymbol{F}_k$ denotes the state-transition function, $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$ denote the system and measurement noise.

Equations of EKF are shown in **Table 1**, and the explanation is:

1. Prior estimation: Calculate $\tilde{\boldsymbol{x}}_{k+1}$ and covariance matrix of prior estimation $\boldsymbol{M}_{k+1}$ by the post estimation results of step k.
2. Kalman gain: Calculate gain $\boldsymbol{K}_{k+1}$ by $\boldsymbol{M}_{k+1}$ and $\boldsymbol{H}$.
3. Post estimation: Calculate $\hat{\boldsymbol{x}}_{k+1}$ and covariance matrix of post estimation $\boldsymbol{\Sigma}_{k+1}$ for the next step.

3.3.2 Unscented kalman filter

UKF applies KF to non-linear systems utilizing the Unscented Transformation (UT). UKF performs better under systems with strong non-linearity compared with EKF (Julier and Uhlmann, 2004). The non-linear form of grid state is:

$$\begin{cases} \boldsymbol{x}_{k+1} = \boldsymbol{f}\left(\boldsymbol{x}_k\right) + \boldsymbol{\omega}_k \\ \boldsymbol{z}_k = \boldsymbol{h}\left(\boldsymbol{x}_k\right) + \boldsymbol{v}_k \end{cases} \tag{6}$$

where, $\boldsymbol{f}\left(\boldsymbol{x}_k\right)$ is $n \times 1$ dimensional non-linear state-transition function, $\boldsymbol{\omega}_k$ and $\boldsymbol{v}_k$ are $n \times 1$ and $m \times 1$-dimensional Gaussian white noise with the zero mean.

Equations of UKF are shown in **Table 2**, and the explanation is:

1. Generate sigma-points: Generate $2n+1$ sigma-points (Julier and Uhlmann, 2004).
2. Prior estimation: Utilizing sigma points to calculate the prior estimation $\tilde{\boldsymbol{x}}_{k+1}$, and the prior estimation error covariance $\boldsymbol{M}_{k+1}$.
3. Measurement correction: Calculate the prior estimated measurement $\tilde{\boldsymbol{z}}_{k+1}$. Difference between $\tilde{\boldsymbol{z}}_{k+1}$ and $\boldsymbol{z}_{k+1}$ is used to calculate covariance matrices $\boldsymbol{\Sigma}_{k+1}^{zz}$ and $\boldsymbol{\Sigma}_{k+1}^{xz}$.
4. Kalman gain: Gain $\boldsymbol{K}_{k+1}$ and post estimated state $\hat{\boldsymbol{x}}_{k+1}$ are calculated by $\boldsymbol{\Sigma}_{k+1}^{zz}$ and $\boldsymbol{\Sigma}_{k+1}^{xz}$.

TABLE 2 Equations of UKF.

Step	Formula
Generate sigma points	$\boldsymbol{x}_0, \boldsymbol{x}_i, \omega_i^{(m)}, \omega_i^{(c)}$
Prior estimation	$\boldsymbol{x}_{i,k+1} = \boldsymbol{f}\left(\boldsymbol{x}_{i,k}^a, k\right)$
	$\tilde{\boldsymbol{x}}_{k+1} = \sum_{i=0}^{L} \omega_i^{(m)} \boldsymbol{x}_{i,k+1}$
	$\boldsymbol{M}_{k+1} = \sum_{i=0}^{L} \omega_i^{(c)} \left(\boldsymbol{x}_{i,k+1} - \tilde{\boldsymbol{x}}_{k+1}\right)\left(\boldsymbol{x}_{i,k+1} - \tilde{\boldsymbol{x}}_{k+1}\right)^T$
Measurement correction	$\boldsymbol{z}_{i,k+1} = \boldsymbol{h}\left(\boldsymbol{x}_{i,k+1}\right), \tilde{\boldsymbol{z}}_{k+1} = \sum_{i=0}^{L} \omega_i^{(m)} \boldsymbol{z}_{i,k+1}$
	$\boldsymbol{\Sigma}_{k+1}^{zz} = \sum_{i=0}^{L} \omega_i^{(c)} \left(\boldsymbol{z}_{i,k+1} - \tilde{\boldsymbol{z}}_{k+1}\right)\left(\boldsymbol{z}_{i,k+1} - \tilde{\boldsymbol{z}}_{k+1}\right)^T + \boldsymbol{R}_{k+1}$
	$\boldsymbol{\Sigma}_{k+1}^{xz} = \sum_{i=0}^{L} \omega_i^{(c)} \left(\boldsymbol{x}_{i,k+1} - \tilde{\boldsymbol{x}}_{k+1}\right)\left(\boldsymbol{z}_{i,k+1} - \tilde{\boldsymbol{z}}_{k+1}\right)^T$
Kalman gain	$\boldsymbol{K}_{k+1} = \boldsymbol{\Sigma}_{k+1}^{xz} \left(\boldsymbol{\Sigma}_{k+1}^{zz}\right)^{-1}$
	$\hat{\boldsymbol{x}}_{k+1} = \tilde{\boldsymbol{x}}_{k+1} + \boldsymbol{K}_{k+1}\left(\boldsymbol{z}_{k+1} - \tilde{\boldsymbol{z}}_{k+1}\right)$
	$\boldsymbol{\Sigma}_{k+1} = \boldsymbol{M}_{k+1} - \boldsymbol{K}_{k+1} \boldsymbol{\Sigma}_{k+1}^{zz} \boldsymbol{K}_{k+1}^T$

3.4 Bad data detection

Errors in the initial measurement data can be the source of distortion of estimated states, leading to wrong decisions of EMS. Therefore, detection of bad data in measurements is applied to detect possible errors. The most common method is constructing an empirical threshold and detect by the residual function (Merrill and Schweppe, 1971):

$$\tau < \left\| \boldsymbol{z} - \boldsymbol{h}\left(\hat{\boldsymbol{x}}\right) \right\|_2 \tag{7}$$

where, $\hat{\boldsymbol{x}}$ denotes the state estimation results, $\left\| \boldsymbol{z} - \boldsymbol{h}\left(\hat{\boldsymbol{x}}\right) \right\|_2$ denotes the l^2-norm of residuals and τ denotes the empirical threshold generated form historical data.

Holding of Eq. 7 denotes that residuals of the estimated states exceed the threshold. Then a bad data alarm will be triggered, indicating the existence of bad data.

3.5 DQN algorithm

The DQN algorithm, used with replay buffer and target network, is a representative DRL algorithm. Applying DQN algorithm can overcome the complexity of storing Q-table in Q-learning. Other improvements of DQN over Q-learning are (Mnih et al., 2015):

1. Construct replay buffer: At each step, store the experiences in buffer $\mathbb{D}$. When updating the neural network, a mini batch is extracted to update weights θ. Format of experience $\boldsymbol{e}_t$ is:

$$\boldsymbol{e}_t = \left(s_t, a_t, r_t, s_{t+1}\right) \tag{8}$$

where, s_t, a_t, and r_t denote the state, action and reward of step t during the interacting between the agent and environment.

2. Use target Q network: DQN is a dual-network model. A target network is defined and periodically updated, generating target Q. Thus, the equation of gradient descent is:

$$\nabla_{\theta_i} L(\theta_i) = \mathbb{E}_{s,a,r,s'}\left[\left(r + \gamma max Q\left(s_{t+1}, a_{t+1}, \theta_i^-\right) - Q\left(s_t, a_t, \theta_i\right)\right)\right] \tag{9}$$

where, $Q\left(s_{t+1}, a_{t+1}, \theta_i^-\right)$, $Q\left(s_t, a_t, \theta_i\right)$ are generated by the weights of target and current Q network, respectively.

3. Normalize reward: Restrain the reward r in $(-1, 1)$, which can reduce the gradient during updating.
4. Adopt ε-greedy strategy: Adopt a random strategy at each step with a chance of $1-\varepsilon$, and ε increases with training.

4 Smart grid FDIA

4.1 FDIA model based on complete topology information

In this section, we construct FDIAs under the assumption of complete topology information and unlimited cost. Thus the attacker can extract whole measurement function $\boldsymbol{h}(\boldsymbol{x})$ and construct attack without considering the cost, resulting in the inefficacy of empirical bad data detection mechanism.

Equations for constructing attacks are (Liu et al., 2009):

$$\begin{aligned} \boldsymbol{z}_a &= \boldsymbol{z} + \boldsymbol{a} \\ \hat{\boldsymbol{x}}_a &= \boldsymbol{H}^{-1}\boldsymbol{z}_a = \hat{\boldsymbol{x}} + \boldsymbol{c} \end{aligned} \tag{10}$$

where, $\boldsymbol{z}_a$ denotes the attacked measurement values that the system obtains, $\boldsymbol{z}$ denotes the real measurement values of the grid, $\boldsymbol{a}$ denotes the attack vector, $\hat{\boldsymbol{x}}_a$ denotes the estimated states under attacks, $\hat{\boldsymbol{x}}$ denotes the estimated states without attacks, $\boldsymbol{c}$ denotes the change of state values.

As for a non-linear power system, FIDA can also satisfy the measurement equation $\boldsymbol{z}_a = \boldsymbol{h}(\boldsymbol{x}_a)$ by:

$$\boldsymbol{a} = \boldsymbol{z}_a - \boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x} + \boldsymbol{c}) - \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{x}_a) - \boldsymbol{h}(\boldsymbol{x}) \tag{11}$$

$$\begin{aligned} \left\|\boldsymbol{z}_a - \boldsymbol{h}(\hat{\boldsymbol{x}}_a)\right\|_2 &= \left\|(\boldsymbol{z} - \boldsymbol{h}(\hat{\boldsymbol{x}})) + (\boldsymbol{a} + \boldsymbol{h}(\hat{\boldsymbol{x}}) - \boldsymbol{h}(\hat{\boldsymbol{x}}_a))\right\|_2 \\ &= \left\|\boldsymbol{z} - \boldsymbol{h}(\hat{\boldsymbol{x}})\right\|_2 \end{aligned} \tag{12}$$

If the static state estimation is operated as always, there is $\hat{\boldsymbol{x}}_a \approx \boldsymbol{x}_a$ and $\boldsymbol{h}(\hat{\boldsymbol{x}}_a) \approx \boldsymbol{h}(\boldsymbol{x}_a)$. Comparing Eq. 12 with Eq. 7, ignoring the inherent Gaussian noise in Eq. 6, the residuals under valid FDIAs are the same as the residuals without an attack, namely $\left\|\boldsymbol{z}_a - \boldsymbol{h}(\hat{\boldsymbol{x}}_a)\right\|_2 = \left\|\boldsymbol{z} - \boldsymbol{h}(\hat{\boldsymbol{x}})\right\|_2$.

In other words, a FIDA constructed this way doesn't change the residuals in Eq. 7. Therefore, a valid FIDAdoesn't trigger the bad data detection alarm mentioned in Section 3.4.

TABLE 3 Classification of FDIAs.

Duration	Attack intensity	Types of FDIAs
continuous	constant	continuous-constant-intensity attack
continuous	variable	continuous-variable-intensity attack
transient	constant	transient-constant-intensity Attack
transient	variable	transient-variable-intensity attack

Algorithm 1. Strategies of three attacks.

```
Initialize x and x_a = x
for t = 1 to T do
    for t_start < t < t_end do
        if Attack1 is adopted then
            Construct z_a by Equation 13
        else if Attack2 is adopted then
            With a probability ε_attack to construct z_a by Equation 13
            Otherwise z_a = z
        else if Attack3 is adopted then
            Construct z_a by Equation 14
        else
            Break
        end
        Falsify the measurement by z ← z_a
    end
end
```

4.2 Types of attacks

Considering the diversity of attacks and attackers' intentions, types of FDIAs are also diverse. In this paper, we classified FDIAs by duration and variation of intensity.

According to the duration, attacks can be divided into transient attack and continuous attack (Jiang et al., 2020). The transient attack tends to have stronger perturbation in a short period, while the continuous attack can remain undetected for a longer period by applying weak perturbation.

According to the intensity, the attack can be divided into constant-intensity and variable-intensity attack. The constant-intensity attack vectors are similar in magnitude, while the variable-intensity attack vectors can be stochastic or asymptotic.

Detailed classification of FDIA is shown in Table 3. Three types of attacks are selected and studied in this paper, the strategies are shown in Algorithm 1 and the detailed equations are:

1. Attack1. Continuous-constant-intensity attack:

Define the start and end of the attack as t_{start} and t_{end}. While $t_{start} < t < t_{end}$, construct and inject the attack by Eq. 13.

$$\begin{cases} \boldsymbol{x}_a = \boldsymbol{x} + \boldsymbol{c} \cdot \boldsymbol{\omega} \\ \boldsymbol{z}_a = \boldsymbol{h}(\boldsymbol{x}_a) \end{cases} \tag{13}$$

where, $\boldsymbol{c} = [c_{\varphi 1}, c_{\varphi 2}, \cdots c_{\varphi n}, c_{V1}, c_{V2}, \cdots c_{Vn}]$ denotes the intended deviation of phase angle $c_{\varphi i}$ and magnitude c_{Vi} on the node voltage, $\boldsymbol{\omega}$ is a standardized normal variable.

Attack-1 is a typical form of FDIA. When $c_{\varphi i} = c_{Vi} = 0$, no attack will be injected on bus i, and $\boldsymbol{c}$ depends on the intention of attackers. Since we focus on detection, the programming problem of determining $\boldsymbol{c}$ is replaced by a Gaussian variable $\boldsymbol{\omega}$. Multiplying by $\boldsymbol{\omega}$, the attack vector varies in a reasonable range $(\mathbf{0}, \boldsymbol{c})$. Thus, diversified attack intention can be included, and value of $\boldsymbol{c}$ can be fixed. For example, if we define $c_{V1} = 0.1$p.u., all attacks with the intensity between 0 and 0.1p.u. on bus one are considered as long as there are enough episodes. Moreover, $\boldsymbol{\omega}$ can also make FDIAs hard to be detected by empirical method.

2. Attack2. Transient-constant-intensity attack:

While $t_{start} < t < t_{end}$, at each step, with a probability of ε_{attack} to construct and inject the attack by Eq. 13. In other cases, no attack is conducted. Attack-2 aims to test the response speed of the detection method.

3. Attack3. Continuous-variable-intensity (incremental) attack:

While $t_{start} < t < t_{end}$, construct the attack by:

$$\begin{cases} \boldsymbol{x}_a = \boldsymbol{x}_a + \dfrac{\boldsymbol{c}}{t_{end} - t_{start}} \\ \boldsymbol{z}_a = \boldsymbol{h}(\boldsymbol{x}_a) \end{cases} \tag{14}$$

Attack3 is valid during steady-state grid operation when state $\boldsymbol{x}$ undergoes little change. One obvious feature of **Attack3** is that $\boldsymbol{x}_a$ is cumulative. Since the cumulation of deviation is slow, Attack3 is hard to be detected at an early stage.

5 DQN-based FDIA detection

In this section, we first introduce a combined dynamic-static detection mechanism. Then, to avoid the complexity and achieve more effectiveness, we proposed a DQN-based FDIA detection method.

5.1 Combined dynamic-static empirical FDIA detection

According to **Section 4.1**, it is hard to detect well-constructed FDIAs by bad data detection. However, when attacked by FDIAs, results of different state estimation methods produce a significant difference: Result of static state estimation will deviate from the true state, since it only depends on the real-time measurements. Result of dynamic method is closer to the true state due to the prediction steps.

So in our works, we combine results of static method (WLS) with dynamic method (EKF and UKF) and detect attack based on their inconsistency:

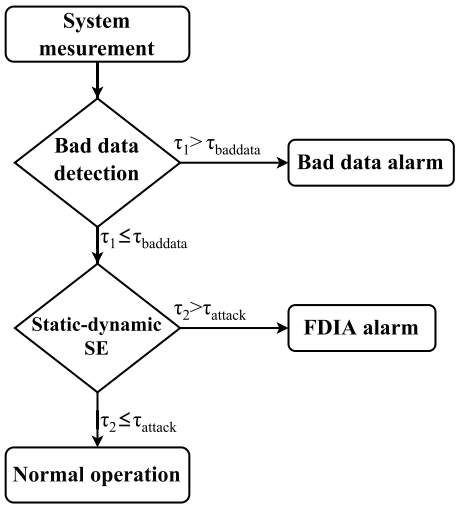

FIGURE 1
FDIA detection mechanism.

$$\tau_1 = \left| \boldsymbol{x}_{KF} - \boldsymbol{x}_{WLS} \right\|_2 > \tau_{attack} \tag{15}$$

where, x_{KF} and x_{WLS} denote the result of KF and WLS, τ_{attack} is the threshold for determining an attack. When Eq. 15 holds, the system is determined to be attacked.

However, grid states change abruptly sometimes due to other factors, which can also leads to the deviation of state estimation. Thus Eq. 15 can be false-positive, so we combine it with bad data detection:

$$\tau_2 = \left\| \boldsymbol{z} - \boldsymbol{h}(\boldsymbol{x}_{WLS}) \right\|_2 > \tau_{baddata} \tag{16}$$

The mechanism is summarized as **Figure 1**, when Eq. 16 holds, the system is determined to have bad data, When Eq. 16 does't hold but Eq. 15 holds, the system is determined to be attacked by FDIAs.

However, since the gird is vulnerable to disturbances, evaluating the performance of the method only by accuracy is incomplete. Considering the time sensitivity, an effective detection of FDIA in this paper is defined in Eq. 17. Utilizing Eq. 17, performance of detection is evaluated by detection rate.

$$t_{start} \leqslant t_{alarm} \leqslant t_{start} + 2 \tag{17}$$

where, t_{start} denotes the start of attack, and t_{alarm} denotes the time that the attack is detected.

If Eq. 17 holds, it shows that the attack is detected within a short period of time. Thus the detection is effective and the safety of grid can be protected.

5.2 DQN-based FDIA detection scheme

Due to the changing load of grid and randomness of attacks, the threshold of Eq. 15 varies greatly in different scenarios. Therefore, it is impractical and costly to apply a certain empirical threshold τ_{attack} in a wide range of grids.

To address the shortcomings of empirical detection, FIDA detection is formulated as a MDP and trained utilizing DQN algorithm. Detection is achieved through neural network, equivalent to a dynamic threshold instead of the empirical threshold τ_{attack} in Eq. 15. The neural network is trained by interacting with the environment during the MDP of FDIA detection (An et al., 2019; Kurt et al., 2019). After an action of detection, agent receives a feedback (reward) from the environment for guiding the actions by updating the neural network (Sutton and Barto, 1998).

5.2.1 MDP-based attack detection model

MDP is the model for sequential decision making (Baxter, 1995). When the state of environment is Markovian, MDP can simulate the strategies and rewards that an agent can achieve. We formulate FIDA detection process as a MDP due to the feature of sequential decision and uncertainty of attack model.

Main components of MDP are state space $\boldsymbol{S}$, action space $\boldsymbol{A}$, state transition $\boldsymbol{P}$ and reward $\boldsymbol{R}$, denoted by $\{\boldsymbol{S},\boldsymbol{A},\boldsymbol{P},\boldsymbol{R}\}$ (Luong et al., 2019). For the FDIA detection, we defined $\boldsymbol{S}$ and $\boldsymbol{A}$ as:

$$\begin{aligned} \boldsymbol{S} &= [s_n, s_a] \\ \boldsymbol{A} &= [a_c, a_s] \end{aligned} \tag{18}$$

where, s_n represents that no attack exists in the grid, s_a represents that the gird is under an attack, a_c denotes that no attack is detected and the system continues to operate, a_s denotes that the attack is detected, and the MDP ends when an attack is detected or time ends.

$\boldsymbol{P}$ represents the state transition function. To address the random and unpredictable characteristics of cyber attacks, a model-free approach is taken to define state transition, i.e., the state transition probability $p(s'|s,a)$ is unknown (An et al., 2019). When the system chooses to continue the operation, the state of the next step is calculated by state estimation and perceived by the agent.

$\boldsymbol{R}$ represents the reward function. For the sparse characteristics of the power grid under attack, we define $\boldsymbol{R}$ by efficacy of detection. When agent detects attacks during normal

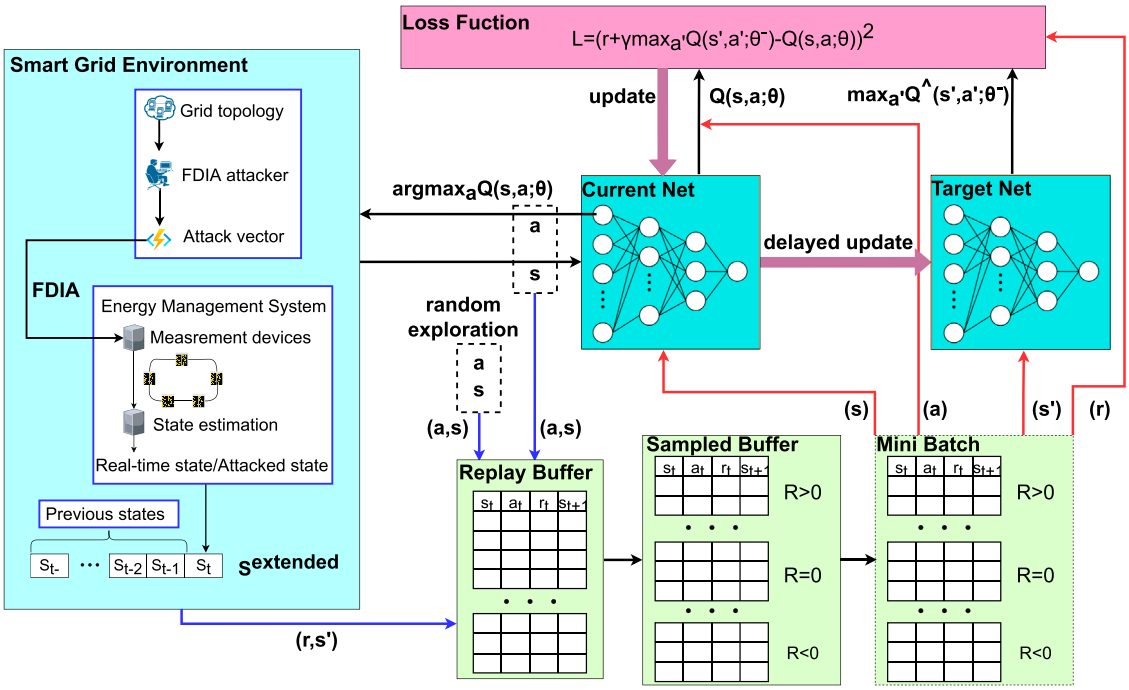

FIGURE 2
Framework of optimized DQN-based detection schem.

Initialize power grid topology
for *episode= 1 to* N_0 **do**
  Initialize system state
  for *time= 1 to* T **do**
    if *in the attack time* **then**
      Construct and inject the attack vector to the measurement
    end
    Get state s_t by state estimation
    Extend state space to get o_t
    Randomly selected actions a_t
    Get reward r_t, preprocess to get s_{t+1} and extend to o_{t+1}
    Store experience (o_t, a_t, r_t, o_{t+1}) in the sampled replay buffer $\mathbb{D}^-$, $\mathbb{D}^+$, and $\mathbb{D}^0$
    if $a_t = a_s$ **then**
      break
    end
  end
end
for *epoch= 1 to* N_1 **do**
  for *episode= 1 to* N_2 **do**
    Initialize system state
    for *time= 1 to* T **do**
      if *in the attack time* **then**
        Construct and add the attack vector to the measurement value
      end
      Get state s_t by state estimation
      Extend state space to get o_t
      Get action a_t by ε-greedy strategy
      Get reward r_t, preprocess to get s_{t+1} and extend to o_{t+1}
      Store experience (o_t, a_t, r_t, o_{t+1}) in the sampled replay buffer $\mathbb{D}^-$, $\mathbb{D}^+$, and $\mathbb{D}^0$
      Updated θ by gradient descent
      if $a_t = a_s$ **then**
        break
      end
    end
    Every certain period, let $\theta^- = \theta$
  end
end

Algorithm 2. Training of the a optimized DQN-based FDIA detection algorithm.

Initialize power grid topology
for *episode= 1 to* N_3 **do**
  Initialize system state
  for *time= 1 to* T **do**
    if *in the attack time* **then**
      Construct and inject the attack vector to the measurement
    end
    Get state s_t by state estimation and extended to get o_t
    Get action a_t by the trained network
    if $a_t = a_s$ **then**
      break
    end
  end
end
Calculate the detection rate

Algorithm 3. Testing of the FIDA detection algorithm.

operation, the reward is negative. When the agent detects attacks under attacks, the reward is positive, and the more timely the detection, the more the rewards. Rewards of the other cases are 0. The detailed function is:

$$r_t(s_t, a_t) = \begin{cases} 0 & s_t = s_n \quad a_t = a_c \\ 0 & s_t = s_a \quad a_t = a_c \\ -\beta^- & s_t = s_n \quad a_t = a_s \\ \beta^+ \dfrac{t - t_{start}}{t_{end} - t_{start}} & s_t = s_a \quad a_t = a_s \end{cases} \tag{19}$$

where, t_{start}, t_{end} denote the start and end of the attack, respectively, β^-, β^+ denote the reward coefficient.

5.2.2 Optimized DQN-based detection scheme

Framework of the detection scheme is shown in **Figure 2**. The detailed training and testing algorithms are given in **Algorithm 2** and **Algorithm 3**.

TABLE 4 Simulation settings.

Environment	Settings
Number of buses on IEEE systems	9, 14, 30, 57
Time steps of each episode T	100
Duration of attack T_a	50
Probability of ε_{attack} in Attack2	0.2
Intended deviation of voltage angle $c_{\varphi i}$	0°
Intended deviation of voltage magnitude c_{Vi}	0.1p.u
Parameters of UKF	$\alpha = 10^{-3}, \beta = 2, \kappa = 0$
Learning rate α_l	0.001
Discount factor γ	0.9
Dimension of extended State space N	4
Reward coefficient β^-, β^+	1:1
ϵ-greddy strategy initial ϵ	0.9
ϵ-greddy strategy increment of ϵ	0.0005
Replay buffer size	1,000
Mini-batch size	32
Sampled mini-batch size $\mathbb{D}^-$, $\mathbb{D}^+$, and $\mathbb{D}^0$	4, 4, 24
Time interval of updating target network	10
Episodes of random exploring $N0$	100
Epoches of training $N1$	10, 20
Episodes of training $N2$	500
Episodes of testing $N3$	100
Random seeds	101,102,103,104,105

6 Simulations

6.1 Simulations setup

Extensive simulations performed to simulate the actual scenarios of FIDA attack-detection process. First, to ensure the practicability, simulations are based on IEEE 9, 14, 30 and 57-bus networks by MATPOWER (Zimmerman et al., 2011). Second, due to the diversity of attacks and attack intentions (An and Liu, 2019), three types of FDIAs are adopted, namely Attack-1, 2, and 3. Cases with single attack and multiple attacks are both considered during simulations. Then, the attacks aim at the magnitude of node voltages, with the intensity to cause voltage violation (Zhu and Liu, 2016; Zheng et al., 2020). Third, WLS is adopted in static estimation while EKF and UKF are adopted for dynamic state estimation in different cases. In addition, random seeds were used to reduce the random error. Details of settings are shown in **Table 4**.

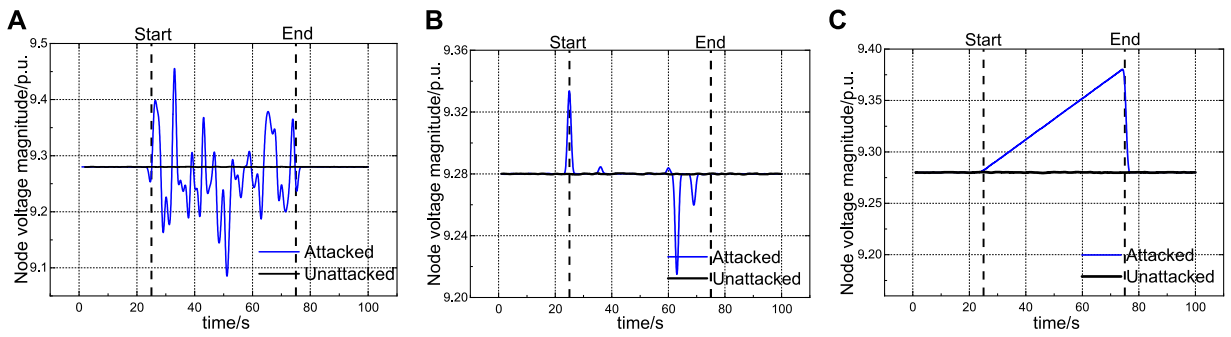

FIGURE 3
Effects of three typical attacks. **(A)** Effect of Attack-1, **(B)** Effect of Attack-1, and **(C)** Effect of Attack-1.

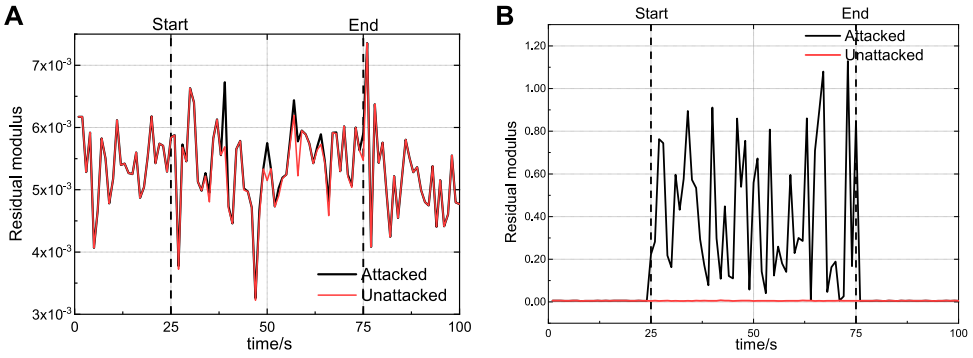

FIGURE 4
Bad data detection results **(A)** Bad data detection results under valid attack and **(B)** Bad data detection results under invalid attack.

6.2 Simulations and effects of attacks

Considering the change in voltage magnitude of static state estimation result, effects of three typical attacks is in Figure 3. The dashed lines indicate the start and end of attacks.

Figure 3A shows effect of Attack-1, namely continuous-constant-intensity attack. Attack-1 injects attack continuously and magnitude of the attack vector follows the same Gaussian distribution. Thus, deviation in voltage magnitude is obvious under Attack-1.

Figure 3B shows effect of Attack-2, namely transient-constant-intensity attack. Attack-2 injects the attack vector intermittently and magnitude of the vector follows the same Gaussian distribution. Deviation generated by Attack-2 is also considerable, but the attack duration is compressed. Thus, for the detector, higher response speed is required.

Figure 3C shows effect of Attack-3, namely continuous-variable-intensity (incremental) attack. Attack-3 injects the attack vector continuously and magnitude of the vector is cumulative. The deviation between two steps generated by Attack-3 is smaller than other attacks, which can avoid being detected. However, the deviation accumulates over a period of time and the amplitude at the end of attack is also considerable, so the consequence of Attack-3 can be severe.

Taking Attack-1 as an example, Figure 4 shows the differences between valid and invalid attacks. In Figure 4A, the residual modulus of the no-attack case is about 0, and almost overlaps with the valid-attack case. But in Figure 4B, the residual modulus fluctuates greatly at a high level under an invalid attack, exposing the attack to detector. In summary, the difference of effectiveness between valid and invalid attacks to bypass the bad data detection is obvious.

Changes in node voltage of IEEE 9-bus system when attacked by the FDIA are shown in Figure 5. Since the intended deviation of angle $c_{\varphi i} = 0°$, the phase angle deviates slightly during the attack in Figure 5A. Moreover, the intended deviation of magnitude $c_{Vi} = 0.1$p.u., so the intensity of attack is within

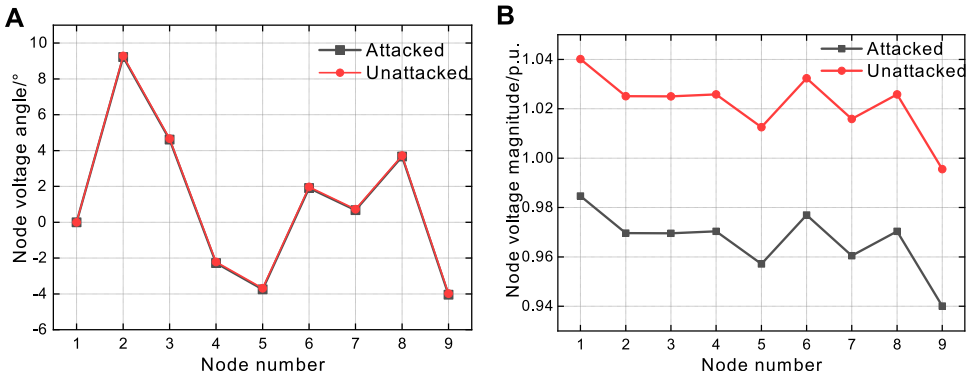

FIGURE 5
Static state estimation results under valid attack. **(A)** Change of node voltage angle after the valid attack and **(B)** Change of node voltage magnitude after the valid attack.

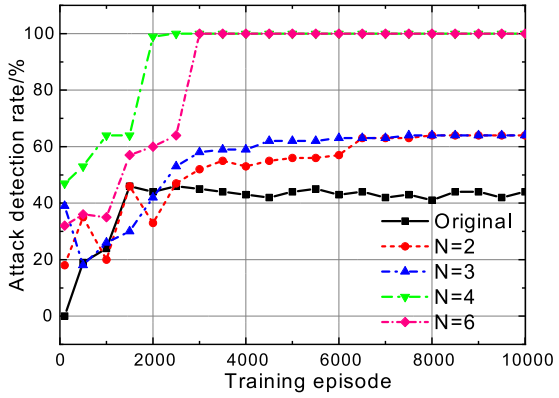

FIGURE 6
Attack detection rate under different state spaces.

$(0, c_j)$. Correspondingly, the bus voltage magnitude undergoes a large deviation in Figure 5B, misleading the following grid operation.

6.3 Effectiveness of optimized DQN-based method

Simulations in this section compare the optimized DQN-based method with the original DQN-based method and empirical threshold method. Moreover, cases with different dimension of state space are also compared Effectiveness of adopting sampled replay buffers and extending state space is proved.

First, to verify that effectiveness of extending state space, we changed the dimension of state space in different cases. Results are shown in Figure 6. The detection of original method is unstable during the training and reaches convergence after 3,000 episodes, the attack detection rate fluctuates at a low level (43%), and the detection rate is unstable with fluctuation.

With the extended state space, the detection rate increases by at least 21%–64%, and the fluctuation decreases significantly. Comparing the above cases, the detection rate reaches near 100% after convergence in cases that $N \geq 4$, so in the rest of this paper $N = 4$.

Second, to prove efficacy of the optimized method, we simulated the detections with optimized DQN-based method, original DQN-based method and empirical method.

The empirical threshold method uses a fixed threshold consturcted from experiences. In this paper the algorithm is: By $\tau = \|x_{KF} - x_{WLS}\|_2$, calculate τ_1 in no-attack cases and τ_2 in attacked cases, $\tau_{attack} = \tau_2 - \tau_1$. So the detection rate is a fixed number and behaves as a horizontal line since the empirical threshold is hardly updated online in practice.

Results are shown in Figure 7. Each case is simulated in five parallel groups utilizing random seeds in Table 4. Detection rates are averaged and the shadows denote the standard deviations between different groups.

Comparing the performances in Figure 7, empirical method doesn't perform well in detection. Detection rate of empirical method is 61% and 80% against Attack-1 and Attack-2, and is only 30% against Attack-3. What's more, the method with original DQN performs well at certain episodes in Figures 7C, F, but the detection rate fluctuates substantially throughout the training in Figures 7A, D, E. The convergence of training with original DQN is difficult, too.

As for the optimized DQN-based method, cases with EKF converges around 8,000 episodes, and the converged detection rate is 98.42% against Attack-1, 99.70% against Attack-2, and 100% against Attack-3, with some fluctuation. Cases with UKF converges around 5,000 episodes, and the converged detection

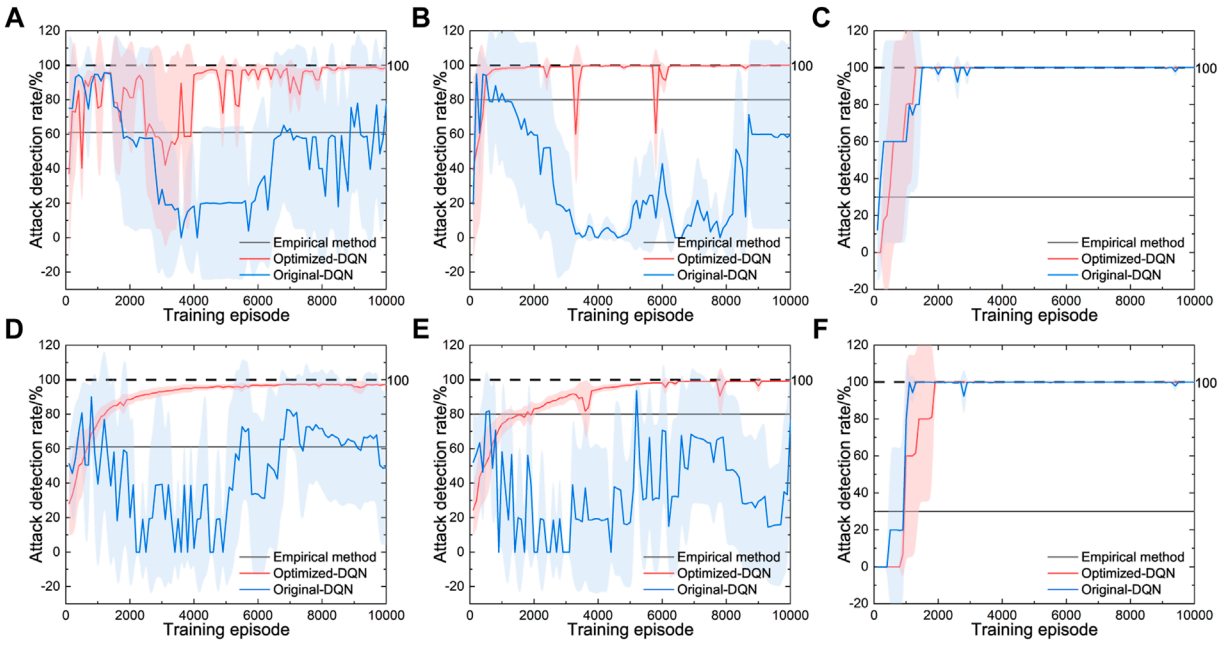

FIGURE 7
Detection with optimized-DQN, original-DQN and empirical method against attacks. **(A)** Detection with EKF against Attack-1, **(B)** Detection with EKF against Attack-2, **(C)** Detection with EKF against Attack-3, **(D)** Detection with UKF against Attack-1, **(E)** Detection with UKF against Attack-2 and **(F)** Detection with UKF against Attack-3.

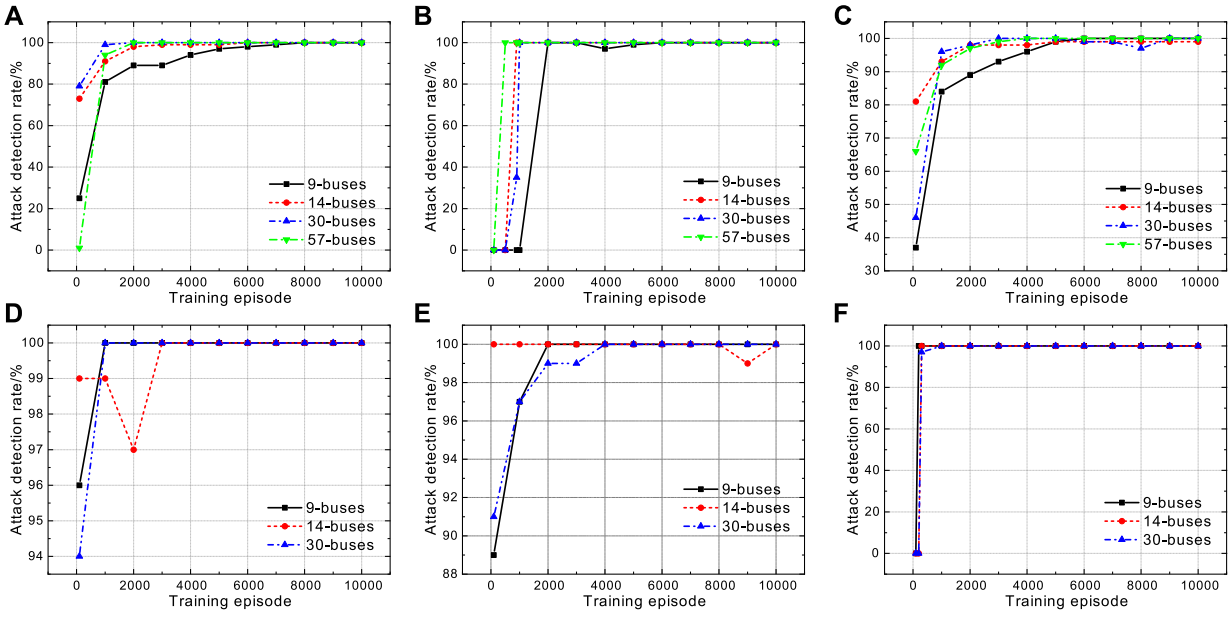

FIGURE 8
Attack detection rate while training against multiple attacks under multiple systems based on EKF or UKF. **(A)** Training against Attack-1 based on EKF, **(B)** Training against Attack-2 based on EKF, **(C)** Training against Attack-3 based on EKF, **(D)** Training against Attack-1 based on UKF, **(E)** Training against Attack-2 based on UKF, and **(F)** Training against Attack-3 based on UKF.

TABLE 5 Performance of detection in different systems against multiple attacks.

Type of attack	Number of buses	Detection rate (EKF)/%	Detection rate (UKF)/%
Attack-1	9	99.46	99.88
	14	99.58	99.72
	30	100.00	100.00
	57	100.00	\
Attack-2	9	99.87	100.00
	14	99.00	99.95
	30	99.32	100.00
	57	100.00	\
Attack-3	9	99.46	99.88
	14	100.00	100.00
	30	100.00	100.00
	57	100.00	\

rate is 96.95% against Attack-1, 98.99% against Attack-2, and 100% against Attack-3 with little fluctuation. After convergence, fluctuation of detection rate has been restricted within 4%. In addition, the training process of EKF-based method is less stable than the UKF-based method.

In conclusion, detection rate is improved by at least 15.95% utilizing the optimized DQN-based method. Stability of the training is also improved fundamentally over the original DQN-based method, especially the UKF-based method.

6.4 Simulation in multiple cases

In this section, we compared three types of FDIAs in power systems in different networks to prove that the proposed method is effective for multiple scenarios. In addition, each case was repeated at least three times and results are averaged. Results are shown in Figure 8 and Table 5.

In Figure 8, the training process converges after about 8,000 episodes against Attack-1 and 6,000 episodes against Attack-2. Meanwhile, the speed of convergence is faster in cases based on UKF, especially in the cases against Attack-3. At the final stages of training in above cases, the detection rates fluctuate by 2%. In addition, the convergence speed is slightly faster of a more complex network, since the accumulation of state deviation is faster.

First, in Table 5, the detection performances are similar in different networks, since the detection mechanism only depends on the state estimation performance and is not affected by the network complexity. Second, detection rates in different cases are consistently close to 100% after convergence. Third, UKF-based method performs better in detection than EKF-based method.

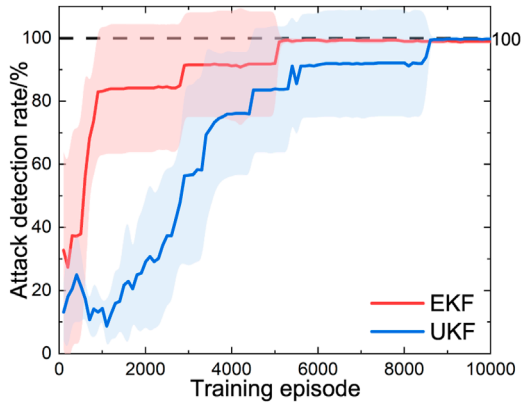

FIGURE 9
Attack detection rate against the hybrid attack.

In summary, the method performs well under different attacks in multiple scenarios.

6.5 Simulation against hybrid attacks

To prove utility of the detection method, a hybrid attack model is constructed. In each episode, the type and start of attack is random and unknown. One of Attack-1, 2, 3 is randomly selected and conducted during the attack based on IEEE 14-bus network.

Rusult of training is shown in Figure 9. Detection rates are also averaged by results of five groups, and standard deviations are given by the shadows.

After training, the detection rate of EKF-based method reaches 99.01%, and the UKF-based method reaches 99.71%. In **Figure 9**, since the attack is hybrid, the detection rate fluctuates at the early stage of training. Trainings converge more slowly compared to the cases against single attack. EKF-based method converges at about 5,500 episodes and UKF-based method converges at about 8,500 episodes.

7 Conclusion

In this paper, a FDIA model with complete topology information and unlimited cost is introduced first. Attacks constructed under this model is verified to have the ability of bypassing the empirical bad data detection. FDIAs are classified by duration and intensity. Three types of attacks and their effects are performed. Then, a detection mechanism is proposed by combining static and dynamic state estimation. Second, the FIDA detection process was formulated as a MDP, and a DQN-based detection method is constructed. To address the problems while training and detection, optimizations were made to improve the efficacy. The DQN-based method is adaptive and has a non-deterministic threshold. Third, sufficient simulations were conducted, including a variety of cases, laying the foundation for studying multiple types of FDIAs. Simulation results prove that the detection rate against FDIA is improved by at least 15.95% over the empirical threshold method. The fluctuation of detection rate has been restricted within 4% during the final stage of training. Moreover, the highest detection rate reached 99.71% against the proposed hybrid attack.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

XL developed the methodology, performed the experiment, analyzed the data, and wrote the manuscript; DA contributed to the conception of the study and manuscript preparation; FC helped perform the analysis with constructive discussions. FZ contributed significantly to analysis and manuscript preparation.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 62173268, Grant 61803295, Grant 61973247, and Grant 61673315; in part by the Major Research Plan of the National Natural Science Foundation of China under Grant 61833015; in part by the National Postdoctoral Innovative Talents Support Program of China under Grant BX20200272; in part the National Key Research and Development Program of China under Grant 2019YFB1704103; and in part by the China Postdoctoral Science Foundation under Grant 2018M643659.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Alnowibet, K., Annuk, A., Dampage, U., and Mohamed, M. A. (2021). Effective energy management via false data detection scheme for the interconnected smart energy hub–microgrid system under stochastic framework. *Sustainability* 13, 11836. doi:10.3390/su132111836

An, D., Yang, Q., Liu, W., and Zhang, Y. (2019). Defending against data integrity attacks in smart grid: A deep reinforcement learning-based approach. *IEEE Access* 7, 110835–110845. doi:10.1109/ACCESS.2019.2933020

An, D., Zhang, F., Yang, Q., and Zhang, C. (2022). Data integrity attack in dynamic state estimation of smart grid: Attack model and countermeasures. *IEEE Trans. Autom. Sci. Eng.* 19, 1631–1644. doi:10.1109/TASE.2022.3149764

An, Y., and Liu, D. (2019). Multivariate Gaussian-based false data detection against cyber-attacks. *IEEE Access* 7, 119804–119812. doi:10.1109/ACCESS.2019.2936816

Annaswamy, A. M., and Amin, M. (2013). Ieee vision for smart grid controls: 2030 and beyond. *IEEE Vis. Smart Grid Controls 2030 Beyond.* doi:10.1109/IEEESTD.2013.6577608

Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Process. Mag.* 34, 26–38. doi:10.1109/MSP.2017.2743240

Ashok, A., Govindarasu, M., and Ajjarapu, V. (2018). Online detection of stealthy false data injection attacks in power system state estimation. *IEEE Trans. Smart Grid* 9, 1–1646. doi:10.1109/TSG.2016.2596298

Baxter, L. A. (1995). Markov decision processes: Discrete stochastic dynamic programming. *Technometrics* 37, 353. doi:10.1080/00401706.1995.10484354

Chen, L., and Wang, X. (2020). Quickest attack detection in smart grid based on sequential Monte Carlo filtering. *IET Smart Grid* 3, 686–696. doi:10.1049/iet-stg.2019.0320

Debs, A. S., and Larson, R. E. (1970). A dynamic estimator for tracking the state of a power system. *IEEE Trans. Power Apparatus Syst.* 89, 1670–1678. doi:10.1109/TPAS.1970.292822

Haque, N. I., Shahriar, M. H., Dastgir, M. G., Debnath, A., Parvez, I., Sarwat, A., et al. (2021). "A survey of machine learning-based cyber-physical attack generation, detection, and mitigation in smart-grid," in 2020 52nd North American Power Symposium (NAPS). doi:10.1109/NAPS50074.2021.9449635

He, Y., Mendis, G. J., and Wei, J. (2017). Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism. *IEEE Trans. Smart Grid* 8, 2505–2516. doi:10.1109/TSG.2017.2703842

Jiang, Q., Chen, H., Xie, L., and Wang, K. (2020). Learning-based cooperative false data injection attack and its mitigation techniques in consensus-based distributed estimation. *IEEE Access* 8, 166852–166869. doi:10.1109/ACCESS.2020.3023117

Julier, S., and Uhlmann, J. (2004). Unscented filtering and nonlinear estimation. *Proc. IEEE* 92, 401–422. doi:10.1109/JPROC.2003.823141

Katiraei, F., and Iravani, M. (2006). Power management strategies for a microgrid with multiple distributed generation units. *IEEE Trans. Power Syst.* 21, 1821–1831. doi:10.1109/TPWRS.2006.879260

Kurt, M. N., Ogundijo, O., Li, C., and Wang, X. (2019). Online cyber-attack detection in smart grid: A reinforcement learning approach. *IEEE Trans. Smart Grid* 10, 5174–5185. doi:10.1109/TSG.2018.2878570

Lei, D., Zhao, J., Hu, M., Chang, X., Zhang, X., and Song, X. (2020). "Optimized configuration scheme of harmonic measuring device considering practical situations of grid nodes and monitoring device," in 2020 IEEE 4th Conference on Energy Internet and Energy System Integration (EI2). doi:10.1109/EI250167.2020.9346993

Li, Q., Li, R., Ji, K., and Dai, W. (2015). "Kalman filter and its application," in 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), 74–77. doi:10.1109/ICINIS.2015.35

Li, Q., Li, S., Xu, B., and Liu, Y. (2019). Optimal node attack on causality analysis in cyber-physical systems: A data-driven approach. *IEEE Access* 7, 16066–16077. doi:10.1109/ACCESS.2019.2891772

Li, Y., and Wang, Y. (2019). False data injection attacks with incomplete network topology information in smart grid. *IEEE Access* 7, 3656–3664. doi:10.1109/ACCESS.2018.2888582

Liang, G., Zhao, J., Luo, F., Weller, S. R., and Dong, Z. Y. (2017). A review of false data injection attacks against modern power systems. *IEEE Trans. Smart Grid* 8, 1630–1638. doi:10.1109/TSG.2015.2495133

Liu, X., Ospina, J., and Konstantinou, C. (2020). Deep reinforcement learning for cybersecurity assessment of wind integrated power systems. *IEEE Access* 8, 208378–208394. doi:10.1109/ACCESS.2020.3038769

Liu, Y., Reiter, M., and Ning, P. (2009). False data injection attacks against state estimation in electric power grids. *ACM Trans. Inf. Syst. Secur.* 14, 21–33. doi:10.1145/1952982.1952995

Luo, W., and Xiao, L. (2021). "Reinforcement learning based vulnerability analysis of data injection attack for smart grids," in 2021 40th Chinese Control Conference (CCC), 6788–6792. doi:10.23919/CCC52363.2021.9550523

Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., Liang, Y. C., et al. (2019). Applications of deep reinforcement learning in communications and networking a survey. *IEEE Commun. Surv. Tutorials* 21, 3133–3174. doi:10.1109/COMST.2019.2916583

Merrill, H. M., and Schweppe, F. C. (1971). Bad data suppression in power system static state estimation. *IEEE Trans. Power Apparatus Syst.* 90, 2718–2725. doi:10.1109/TPAS.1971.292925

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., et al. (2013). Playing atari with deep reinforcement learning. *CoRR* 1312, 5602.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A., Veness, J., Bellemare, M., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518, 529–533. doi:10.1038/nature14236

Mohamed, M. A., Hajjiah, A., Alnowibet, K. A., Alrasheedi, A. F., Awwad, E. M., and Muyeen, S. M. (2021). A secured advanced management architecture in peer-to-peer energy trading for multi-microgrid in the stochastic environment. *IEEE Access* 9, 92083–92100. doi:10.1109/ACCESS.2021.3092834

Oozeer, M. I., and Haykin, S. (2019). Cognitive dynamic system for control and cyber-attack detection in smart grid. *IEEE Access* 7, 78320–78335. doi:10.1109/ACCESS.2019.2922410

Pang, Z.-H., Liu, G.-P., Zhou, D., Hou, F., and Sun, D. (2016). Two-channel false data injection attacks against output tracking control of networked systems. *IEEE Trans. Ind. Electron.* 63, 3242–3251. doi:10.1109/TIE.2016.2535119

Pasqualetti, F., Dörfler, F., and Bullo, F. (2013). Attack detection and identification in cyber-physical systems. *IEEE Trans. Autom. Contr.* 58, 2715–2729. doi:10.1109/TAC.2013.2266831

Schweppe, F. C., and Rom, D. B. (1970). Power system static-state estimation, part ii: Approximate model. *IEEE Trans. Power Apparatus Syst.* 89, 125–130. doi:10.1109/TPAS.1970.292679

Schweppe, F. C., and Wildes, J. (1970). Power system static-state estimation, part i: Exact model. *IEEE Trans. Power Apparatus Syst.* 89, 120–125. doi:10.1109/TPAS.1970.292678

Sinha, A., Thukkaraju, A. R., and Vyas, O. P. (2022). "A multi agent framework to detect in progress false data injection attacks for smart grid," in *Advanced network technologies and intelligent computing*. Editors I. Woungang, S. K. Dhurandher, K. K. Pattanaik, A. Verma, and P. Verma (Cham: Springer International Publishing), 123–141.

Sutton, R., and Barto, A. (1998). Reinforcement learning: An introduction. *IEEE Trans. Neural Netw.* 9, 1054. doi:10.1109/TNN.1998.712192

Tsobdjou, L. D., Pierre, S., and Quintero, A. (2022). An online entropy-based ddos flooding attack detection system with dynamic threshold. *IEEE Trans. Netw. Serv. Manage.* 19, 1679–1689. doi:10.1109/TNSM.2022.3142254

Wan, E., and Van Der Merwe, R. (2000). "The unscented kalman filter for nonlinear estimation," in Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium, 153–158. doi:10.1109/ASSPCC.2000.882463

Wang, Z., Chen, Y., Liu, F., Xia, Y., and Zhang, X. (2018). Power system security under false data injection attacks with exploitation and exploration based on reinforcement learning. *IEEE Access* 6, 48785–48796. doi:10.1109/ACCESS.2018.2856520

Wang, Z., He, H., Wan, Z., and Sun, Y. (2021). Coordinated topology attacks in smart grid using deep reinforcement learning. *IEEE Trans. Ind. Inf.* 17, 1407–1415. doi:10.1109/TII.2020.2994977

Wei, L., Sarwat, A. I., Saad, W., and Biswas, S. (2018). Stochastic games for power grid protection against coordinated cyber-physical attacks. *IEEE Trans. Smart Grid* 9, 684–694. doi:10.1109/TSG.2016.2561266

Wu, Z., He, L., Li, S., Zhang, H., Hu, S., Zhang, M., et al. (2021). "Reinforcement learning based multistage optimal pmu placement against data integrity attacks in smart grid," in 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS). doi:10.1109/ICPS49255.2021.9468170

Zhang, K., and Wu, Z.-G. (2021). "A reinforcement learning-based detection method for false data injection attack in distributed smart grid," in 2021 8th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS), 38–43. doi:10.1109/ICCSS53909.2021.9722027

Zheng, Y., Hill, D. J., Song, Y., Zhao, J., and Hui, S. Y. R. (2020). Optimal electric spring allocation for risk-limiting voltage regulation in distribution systems. *IEEE Trans. Power Syst.* 35, 273–283. doi:10.1109/TPWRS.2019.2933240

Zhu, H., and Liu, H. J. (2016). Fast local voltage control under limited reactive power: Optimality and stability analysis. *IEEE Trans. Power Syst.* 31, 3794–3803. doi:10.1109/TPWRS.2015.2504419

Zimmerman, R. D., Murillo-Sánchez, C. E., and Thomas, R. J. (2011). Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Trans. Power Syst.* 26, 12–19. doi:10.1109/TPWRS.2010.2051168