Attack-Resilient Optimal PMU Placement via Reinforcement Learning Guided Tree Search in Smart Grids

Meng Zhang, Zhuorui Wu, Jun Yan, *Member, IEEE*, Rongxing Lu, *Fellow, IEEE*, and Xiaohong Guan, *Fellow, IEEE*

***Abstract*— The operation of smart grids heavily relies on secure and accurate meter measurements provided by phasor measurement units (PMUs). Therefore, the optimal PMU placement (OPP) aiming to achieve the complete system observability of smart grids with as few PMUs as possible has been extensively investigated. Although many existing studies have focused on the OPP, few of them are concerned with the placement order of PMUs. To protect as many buses as possible in smart grids when installing PMUs in stages owing to high cost, this paper proposes the attack-resilient OPP strategy which places PMUs in order by using reinforcement learning guided tree search, where the sequential decision making of reinforcement learning is utilized to explore placement orders. The least-effort attack model is carried out to screen vulnerable buses such that the buses adjacent to these buses can be placed PMUs in advance to reduce the state space and action space of the large-scale smart grid environment. Based on that, the reinforcement learning guided tree search approach is used to explore the key buses which need placing PMUs, where the repeated exploration of the agent is avoided by tree search. Then, a reasonable placement order of PMUs is obtained according to the action sequence the proposed method provides. Finally, the effectiveness of the proposed method is verified on various IEEE standard test systems and the comparison results with existing methods are provided.**

***Index Terms*— Reinforcement learning, tree search, phasor measurement unit, optimal PMU placement, smart grid.**

I. Introduction

IN SMART grids, system operators make control decisions based on the current system states and formulate the dispatch plan. The supervisory control and data acquisition (SCADA) system is in charge of gathering measured system data, and the state estimation is implemented by the control center to process data from the SCADA system. To guarantee the secure operation of smart grids, the bad data detector (BDD) is widely adopted in smart grids to reveal the deviation between estimated system states and the true values [1]. However, the research in [2]–[6] has shown a malicious attack named false data injection (FDI) attack can invade smart grids while keeping stealthy to the BDD mechanism. The FDI attack can modify the estimated states by compromising measurements in the SCADA system, causing severe consequences such as key lines overloading and load shedding [7].

To improve the security and accuracy of the state estimation of smart grids, many researchers have considered various applications of phasor measurement units (PMUs) recently [8]–[12]. PMUs can provide real-time synchronous phasor measurements with Global Positioning System (GPS) time stamp [13]. Despite the fact that PMUs may be vulnerable to cyber threats such as time-synchronization and GPS spoofing attacks, PMUs and their communication protocols do provide more secure supports than traditional meters in the SCADA system [14]–[16]. Therefore, PMUs can be used to verify state variables of the state estimation such as voltage phase angles independently. However, the costs of PMUs are expensive, and it is not necessary to equip PMUs for every bus since some system states can be calculated through the measurements collected by PMUs on adjacent buses [17]. Once a PMU is placed at one bus, the voltage phasor of the bus and current phasors on the branches incident to the bus can be measured, thus voltage phasors of adjacent buses can be calculated according to Kirchhoff's laws. Therefore, the objective of OPP is to deploy the fewest PMUs in smart grids while obtaining the complete system observability, i.e., using the fewest PMUs to monitor all system measurements [18].

The OPP is considered as an NP-hard combinatorial optimisation problem and many achievements have been made on related researches [19]–[21]. The methods to solve the OPP problem in existing studies can be roughly divided into two categories, i.e., heuristic techniques and mathematical programming techniques [22]. Heuristic techniques relying on search process to solve the OPP, which include greedy algorithm [23], iterated local search [24], spanning tree search

Manuscript received November 11, 2021; revised February 28, 2022 and April 15, 2022; accepted May 4, 2022. Date of publication May 9, 2022; date of current version May 26, 2022. This work was supported by the National Natural Science Foundation of China under Grant 61903292 and Grant 62033005. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. George Loukas. *(Corresponding author: Meng Zhang.)*

Meng Zhang, Zhuorui Wu, and Xiaohong Guan are with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security, School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: mengzhang2009@xjtu.edu.cn; wzr445576941@stu.xjtu.edu.cn; xhguan@xjtu.edu.cn).

Jun Yan is with the Concordia Institute for Information Systems Engineering, Concordia University, Montréal, QC H3G 1M8, Canada (e-mail: jun.yan@concordia.ca).

Rongxing Lu is with the Faculty of Computer Science, University of New Brunswick, Fredericton, NB E3B 5A3, Canada (e-mail: rlu1@unb.ca).

Digital Object Identifier 10.1109/TIFS.2022.3173728

[25], decision tree [26], immune algorithm [27], particle swarm optimization [28], non-dominated sorting genetic [29], genetic algorithm [30], tabu search [31], etc. Mathematical programming techniques relying on analytical calculations to solve the OPP include integer linear programming (ILP) [32] and mixed integer linear programming (MILP), etc. Although aforementioned techniques have significantly promoted the research of the OPP, most of them can not obtain orderly solutions directly. The number of PMUs to ensure the complete observability is generally very large in a large-scale smart grid. The high cost of the PMU installation makes system designers prefer to place PMUs in multistage [33]. Therefore, PMUs placed in each stage should protect as many buses as possible to defend potential data integrity attacks. In this respect, [34] uses ILP approach to realize the optimal multi-stage scheduling of PMU placement, but it does not consider to protect vulnerable buses preferentially during the process of placement. [35] prioritizes the protection of vulnerable buses under the false data injection attack based on the greedy algorithm, but the results may fall into suboptimal solutions.

Motivated by above discussions, this paper focuses on identifying the reasonable PMU placement order, which can observe as many buses as possible with limited PMUs while ensuring the complete observability of the smart grid with the minimum number of PMUs. To this end, the attack-resilient OPP is modeled as a sequential decision making problem which is equivalent to a Markov decision process. The reinforcement learning guided tree search is proposed to solve the sequential decision making problem, i.e., identifying key buses and placing PMUs to measure these buses until the complete observability of the smart grid is obtained. Meanwhile, a reasonable reward function is constructed to help the reinforcement learning guided tree search preferentially taking the actions having more reward and achieving optimal sequential decisions. The proposed method is tested on five IEEE standard test systems, where the corresponding simulations with comparisons are presented to demonstrate its effectiveness.

The main contributions of this paper are summarized as:

- The reinforcement learning guided tree search without modeling smart grids is proposed to solve the OPP problem for the first time. Since the placement order is commonly neglected in existing OPP research, the proposed method can obtain the minimum number of PMUs that ensures the complete observability of the smart grid as well as the placement order that helps to observe as many buses as possible with limited PMUs.
- Considering the environment of the OPP has a limited exploration space, the tree search method is used to accelerate the learning process by avoiding repeated exploration. The proposed reinforcement learning guided tree search can not only avoid the local optimum problem [36], but also solve the OPP problem efficiently.
- The least-effort attack model [37] is used to screen vulnerable buses which need PMUs to observe. Then deep Q network (DQN) is used to approximate the value function in reinforcement learning, which facilitates the proposed method to work well in the large state and action spaces especially in the large-scale grid environment.

The rest of the paper is organized as follows. Section II introduces the preliminaries that are used throughout the paper. Section III proposed the OPP strategy based on reinforcement learning guided tree search. Section IV presents test results of the proposed method on IEEE standard test systems. Finally, the paper is concluded in Section V.

II. Preliminaries

This section will briefly introduce the DC state estimation, the principle of OPP and the least-effort FDI attack model in the proposed scheme.

A. DC State Estimation

The DC state estimation [38] does not consider reactive power flows and injections, where the state variable of bus i is the phase angle and denoted as θ_i. The active power flow from bus i to bus $j\,(i, j = 1, 2, \ldots, n; i \neq j)$ in DC state estimation can be represented as

$$P_{ij} = \frac{\theta_i - \theta_j}{X_{ij}}, \tag{1}$$

and the corresponding active power injection of bus i can be written as

$$P_i = \sum_{j \in N_i} P_{ij}, \tag{2}$$

where X_{ij} is the reactance of branches between bus i and bus j, and N_i represents the set of all buses connected with bus i.

The system measurement for DC state estimation can be expressed as

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{e}, \tag{3}$$

where $\mathbf{z} = [z_1, z_2, \ldots, z_m]^\top$ is the system measurement vector that consists of active power flows and injections, $\mathbf{x} = [\theta_1, \theta_2, \ldots, \theta_n]^\top$ is the state vector consisting of the phase angles of buses in the system. In a PMU-based state estimator, these phase angles can be measured or computed from the synchrophasors collected by the PMUs. $\mathbf{e} = [e_1, e_2, \ldots, e_n]^\top$ is the measurement error vector, $\mathbf{H}^{m \times n}$ is a constant Jacobian matrix implying the system configuration and connection information, m is the number of measurements used for state estimation and n is the number of system state variables.

The problem can be solved by the weighted least square (WLS) method, which obtains the value of state vector $\mathbf{x}$ by minimizing the following objective function:

$$\mathbf{J}(\mathbf{x}) = (\mathbf{z} - \mathbf{H}\hat{\mathbf{x}})^\top \mathbf{R}^{-1} (\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}), \tag{4}$$

where $\mathbf{R}$ represents a diagonal weighting matrix of measurement variances. It can be checked that the optimal value of the state vector in DC state estimation is generally given by

$$\hat{\mathbf{x}} = (\mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}) \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{z}. \tag{5}$$

B. Principle of OPP

PMUs collect system measurements such as bus voltage phasors and branch current phasors and provide GPS timestamps on their measurement reports. Compared to the traditional measurements, these synchrophasors are considered more difficult to be modified by adversaries [23]. Therefore, PMUs are popular in smart girds. Consider placing a PMU on bus i, the voltage phasor and the branch current phasors of the bus can be measured, thus the phase angle θ_i and the power flows $P_{ij}(j \in N_i)$ can be obtained. Meanwhile, according to (1), we have

$$\theta_j = \theta_i - P_{ij} \times X_{ij}, \tag{6}$$

which suggests that θ_j can be obtained from θ_i and $P_{ij}(j \in N_i)$. Therefore, once a PMU is placed on bus i, the state variables of all buses in set N_i and bus i are observable.

The objective of OPP is to ensure the complete system observability by placing the minimal number of PMUs on some critical buses for cost-effective operation. For a smart grid with n buses, the OPP problem is formulated as:

$$\min_{\mathbf{P}} \ \|\mathbf{P}\|_0 \tag{7}$$

$$s.t.\ \mathbf{CP} \geq [1, 1, \ldots, 1]^{\top}_{1\times n}. \tag{8}$$

where $\mathbf{P} = [p_1, p_2, \ldots, p_n]^{\top}$ is a binary decision vector with

$$p_i = \begin{cases} 1, & \text{if a PMU is installed on bus } i \\ 0, & \text{otherwise,} \end{cases} \tag{9}$$

and $\mathbf{C}$ is a binary connectivity matrix whose elements are defined as:

$$C_{ij} = \begin{cases} 1, & \text{if } i = j \\ 1, & \text{if } i \in N_j \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

In addition, the existence of zero injections, conventional power injections and power flows can possibly help to reduce the number of PMUs needed in the OPP problem. Zero injections mean no a generator injecting power or a load consuming power from a bus, where the sum of flows on all branch currents associated with this bus is zero according to the Kirchhoff's Current Law (KCL). If the zero-injection bus and all its neighbours are observable except one, applying the KCL to the zero-injection bus will make the unobservable bus become indirectly observable. On the other hand, conventional power injections and power flows mean some bus injection measurements and branch flow measurements in smart grid are observable beforehand. The conventional power injection has similar effects with that of zero injections, because the sum of branch flows on the bus with the conventional power injection becomes known according to KCL. Moreover, if a bus at one end of the branch with conventional power flows is observable, the bus at the other end of the branch will also be observable.

The topology transformation method can be used to deal with these situations in the OPP problem. The core idea of the topology transformation method is to merge the critical buses related to zero injections or conventional power flows, where the merging process modifies the network topology and the connectivity matrix of the smart grid. For instance, a zero injection bus can be merged with one of its neighbours. For a conventional power flow, both ends of the branch with the conventional power flow can be merged together according to the method. Then, the OPP problem can be solved in the new topology and more details can be found in [39].

C. Least-Effort FDI Attack

According to the DC state estimation, if the system measurement residual satisfies

$$|\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}|^2 < \tau \tag{11}$$

with a given threshold τ determined by system operators based on sensor and system properties [38]. A small τ usually indicates a more precise measurement with limited noises from the meters, though the threshold itself does not have direct impact on the attack model nor the optimal PMU placement.

If (11) is satisfied, the BDD considers system measurements to be statistically normal; otherwise, the BDD will flag the measurement vector as abnormal and try to eliminate the specific piece of data.

The FDI attack exploits the BDD mechanism to target the data integrity [2], where the attack vector is generally constructed as:

$$\mathbf{a} = \mathbf{Hc}. \tag{12}$$

With an arbitrary nonzero vector $\mathbf{c} = [c_1, c_2, \ldots, c_n]^{\top}$, the comprised measurement vector containing attack vector $\mathbf{a} = [a_1, a_2, \ldots, a_m]^{\top}$ is represented as

$$\mathbf{z}_a = \mathbf{z} + \mathbf{a}. \tag{13}$$

As shown in the literature (e.g., [2]), the system residuals before and after an FDI attack are not distinguishable, allowing the attack to bypass BDD and inflict stealthy disruptions and damages to smart grid operations. An adversary can design $\mathbf{c} = [0, 0, \ldots, c_i, 0, \ldots, 0]^{\top}$ to compromise the state variable x_i, which implies the adversary need to manipulate system measurements according to the value of vector $\mathbf{a}$, and the number of measurements that need to be manipulated is the number of nonzero elements of the vector $\mathbf{a}$. In order to save attack cost, the adversary may design vector $\mathbf{c}$ to minimize the number of nonzero elements of the vector $\mathbf{a}$.

This paper considers the situation that the adversary only chooses a state variable to inject false data even though using possible offset in matrix $\mathbf{H}$ by injecting specific errors into several state variables may be more cost-effective [37]. That is, the adversary needs to find a column of matrix $\mathbf{H}$ that has the fewest nonzero elements corresponding to system measurements. Assuming that all power injections P_i and all power flows $P_{ij}, P_{ji}(i, j = 1, 2, \ldots, n; i > j)$ are measured by smart grids, it is observed that the bus which has the fewest adjacent buses possesses the minimal attack cost.

Take the IEEE 9-bus system in Fig. 1 as an example: the adversary can change state variable of Bus 1 as long as he/she modifies the measurements P_1, P_4, P_{14}, and P_{41}. It is shown in Fig. 1 that attack state variables of Bus 1, 2, and 3 requires

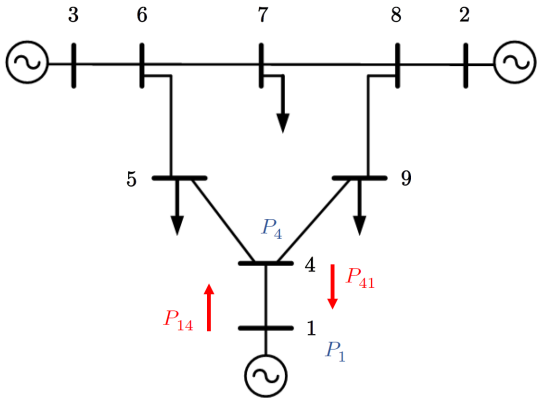


Fig. 1. Example of FDI-vulnerable buses (Buses 1, 2, and 3) in the IEEE 9-bus system.

the least cost, hence these three buses are vulnerable buses in the sense of the least-effort attack.

It is notable that in a non-least-effort attack, the increased effort will still need to follow the same stealthiness requirement set forth in the FDI attack. With the increased effort, the attacker will target more buses as the vulnerable victim to manipulate their measurements, instead of different buses with more criticality.

III. Attack-Resilient OPP

This section introduces the problem of attack-resilient OPP and presents the proposed OPP strategy based on reinforcement learning guided tree search, where the placement process is divided into two stages: the priority placement stage and the main placement stage. The details are as follows.

A. Problem Statement

In the traditional OPP problem (7)-(8), the attention is generally paid on the optimal number of PMUs while the placement order is neglected. Due to the high cost of PMUs, a reasonable PMU placement order can protect more buses to defend potential attacks when only a limited number of PMUs can be installed.

In the attack-resilient OPP, a PMU will be placed at a step. Therefore, the binary decision vector $\mathbf{P}_t$ at step t ($t = 1, 2, \ldots, T$) satisfies

$$\|\mathbf{P}_t\|_0 = t, \tag{14}$$

and

$$(\mathbf{P}_t - \mathbf{P}_{t-1}) \geqslant [0, 0, \ldots, 0]^\top. \tag{15}$$

Therefore, at step t, the number of PMUs is t. And the complete observability is achieved at the final step T.

The objective of the attack-resilient OPP is to identify the reasonable PMU placement order, which can protect as many buses as possible with limited PMUs while ensuring the complete observability of the grid with the minimum total number of PMUs. Therefore, the objective function can be designed as:

$$\sum_{t=0}^{T-1} \gamma^t (\|\mathbf{s}_{t+1} - \mathbf{s}_t\|_0 - 1), \tag{16}$$

where γ is the discount factor. The ith element of the binary observability vector $\mathbf{s}_t$ at step t is defined as

$$s_i = \begin{cases} 1, & \text{if } B_i \neq 0 \\ 0, & \text{if } B_i = 0, \end{cases} \tag{17}$$

where $s_i = 1$ means the bus i is observable while $s_i = 0$ means the bus i is not observable. And B_i is the ith element of vector $\mathbf{B}$ defined as

$$\mathbf{B} = \mathbf{CP}. \tag{18}$$

Therefore, $\|\mathbf{s}_{t+1} - \mathbf{s}_t\|_0$ represents increased number of observable buses after placing a PMU at step $t+1$. Since γ is usually slightly less than 1, the objective function approximates to

$$\sum_{t=0}^{T-1} (\|\mathbf{s}_{t+1} - \mathbf{s}_t\|_0 - 1) = \|\mathbf{s}_T - \mathbf{s}_0\|_0 - T = n - T, \tag{19}$$

which needs minimizing the total number of PMUs. Meanwhile, due to the existing of γ, the objective function will be larger if protecting more buses at earlier step.

Accordingly, the attack-resilient OPP can be formulated as

$$\max_{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_T, T} \sum_{t=0}^{T-1} \gamma^t (\|\mathbf{s}_{t+1} - \mathbf{s}_t\|_0 - 1) \tag{20}$$

$$s.t.\ \mathbf{CP}_T \geq [1, 1, \ldots, 1]_{1\times n}^\top \tag{21}$$

$$(14) - (15). \tag{22}$$

In the problem (20)-(22), the PMU placement order $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_T$ and the total number of PMUs T need to be decided to maximize the objective function.

B. Priority Placement Stage

According to the least-effort FDI attack model, these buses which have only one adjacent bus may be preferred target of the adversaries and are considered to be vulnerable buses in the sense of the least effort attack cost. In priority placement stage, the vulnerable buses are identified and their adjacent buses are selected to place the PMUs first.

In Fig. 2, Bus 1 is considered vulnerable since manipulating measurements of Bus 1 needs the least-effort attack cost. A PMU can be placed on Bus 1 or Bus 2 to protect the vulnerable Bus 1, and it is observed that placing a PMU on Bus 2 may protect more buses. It can be proved that the OPP problem always exists an optimal solution which contains the bus adjacent to the vulnerable buses and the following proof by contradiction is introduced briefly. Assume there is no one optimal solution containing the bus adjacent to the vulnerable buses in smart grids. Without loss of generality, consider Fig. 2 as a part of the power system topology. According to the assumption, no PMU is placed on Bus 2. Therefore, a PMU must be placed on Bus 1 to protect itself, and the corresponding optimal solution contains bus 1 while Bus 2 is not contained. However, moving the PMU from Bus 1 to Bus 2 does not change the complete observability of the system and the number of the PMUs will also not change; there also exists an optimal solution contains Bus 2 while Bus 1 is not contained, which contradicts the assumption. It shall be noted

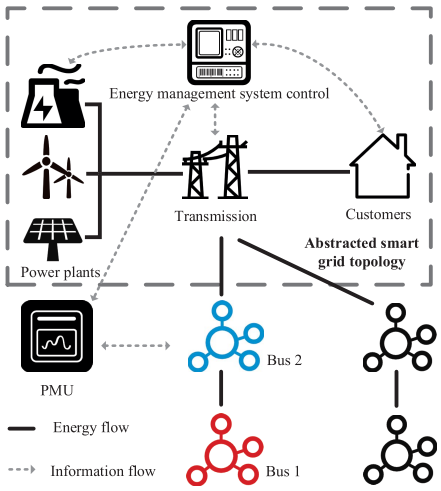


Fig. 2. FDI-vulnerable bus in the smart grids.

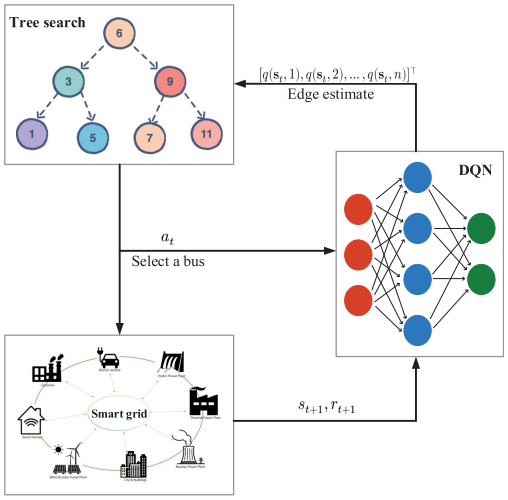


Fig. 3. The flowchart of the reinforcement learning guided tree search.

that several vulnerable buses share the same adjacent bus in the interconnected radial system, placing a PMU on the bus adjacent to these vulnerable buses is optimal.

The discussion above shows that there always exists an optimal solution that contains the bus adjacent to the vulnerable buses, which implies PMUs can be placed on buses adjacent to the vulnerable buses in the priority placement stage. This operation not only shows how to place part of PMUs in advance, but also helps to reduce the state and action spaces of a large-scale grid. If the power grid does not have the bus that possessing only one adjacent bus, the priority placement stage will not be required.

C. Main Placement Stage

After the priority placement stage, the main placement stage aims to find optimal locations from remaining buses toward the objective of attack-resilient OPP using the reinforcement learning approach. The reinforcement learning algorithm aims to find an action sequence from the repeated trial-and-error process in order to maximize total rewards [40]. An agent takes an action at a state and receives a reward related to the goal from the environment. The agent can learn the optimal policy from the cumulative rewards by adjust its actions. In Q-learning, the q value is the cumulative reward function of states and actions. For each triplet of q, state and action create an entry in a Q-Table. However, it is impossible to build and update an oversize Q-Table facing to the large state and action space of the large-scale power system. Meanwhile, the deep neural network that is called DQN can be used to approximate q value and overcome the drawback of Q-Table.

In this paper, we proposed the diagram of the reinforcement learning guided tree search algorithm for PMU placement, which is show in Fig. 3. The agent intends to find the locations of PMU placement by utilizing the tree search algorithm to avoid repeated action sequences. DQN is used to approximate the q value and offers the selection criteria to the search tree. The smart grid can be viewed as the environment that interacts with the agent. The action a represents the PMU location, while the state $\mathbf{s}$ represents the system observability.

The details about the state, action and reward in the algorithm are shown as follows. The state variable is defined as the system observability, which is the same as the equation (17). The desired final state is

$$\mathbf{s}_T = [1, 1, \ldots, 1]^\top. \tag{23}$$

The action $a \in \{1, 2, \ldots, n\}$ in each step is to place a PMU on a bus, i.e., set the ith element of $\mathbf{P}$ as 1 when $a = i$. And the next state s_{t+1} can be obtained according to (17)-(18). Corresponding to the objective function (20), the reward is defined as

$$r_{t+1} = c(\|\mathbf{s}_{t+1} - \mathbf{s}_t\|_0 - 1), \tag{24}$$

where c is a gain coefficient to enhance the difference between rewards of different actions. To reduce the action space, action set A needs to be filtered out actions with smaller reward and the action $a \in A$ is required to satisfy

$$r^a_{t+1} \geq \max_a r^a_{t+1} - c \times b, \tag{25}$$

where $r^a_{t+1}(r^a_{t+1} > 0)$ represents the instant reward of the agent after taking an action a at step $t+1$, and b is the limited coefficient which controls the lower limit of the reward r^a_{t+1}. If b is oversized, the filter function will lose the filtering effect; otherwise, an overly small b may filter out the optimal solution. When the agent is exploring, the reward of all actions will be calculated according to (24) and the action which satisfies (25) has chance to be selected by the agent though there are n kinds of actions.

According to the Bellman equation [40], the value function q is updated by

$$q_{i+1}(\mathbf{s}, a) = \mathbb{E}\left[r_{t+1} + \gamma \max_{a'} q_i(\mathbf{s}_{t+1}, a')|\mathbf{s}, a\right], \tag{26}$$

where i is the iterative index and a' is the action in the $(t+1)$-th step, and discount factor γ is set slightly smaller than 1 to ensure the convergence of the q during the learning process [40]. When $i \to \infty$, the optimal value function from

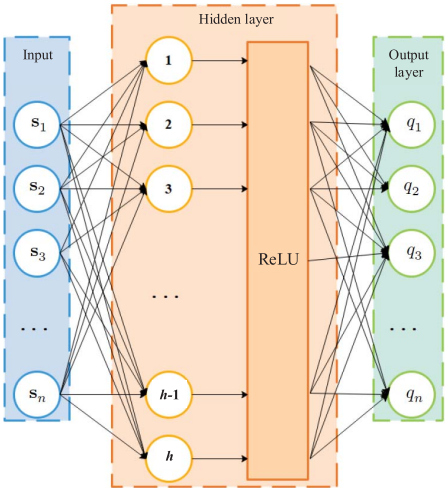


Fig. 4. The structure of DQN for the attack-resilient OPP.

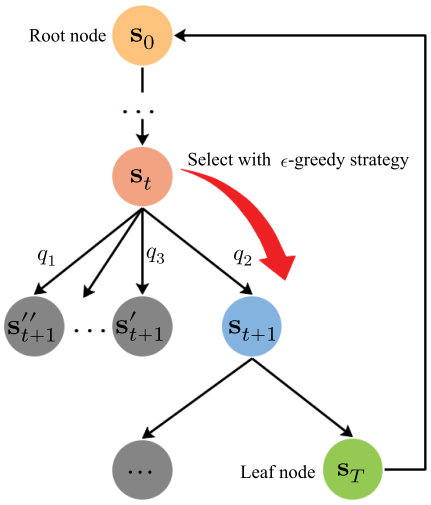


Fig. 5. The work flow of a tree search.

the current step t to the final step T can be obtained by

$$q^*(\mathbf{s}, a) = \max_{\pi} \mathbb{E}\left[\sum_{k=0}^{T-t} \gamma^k r_{t+k} | \mathbf{s}_t = \mathbf{s}, a_t = a\right], \quad (27)$$

where π is the action strategy. Therefore, the corresponding optimal action is obtained as

$$a^* = \arg\max_{a \in A} q^*(\mathbf{s}, a). \quad (28)$$

The DQN is a two-layer fully connected neural network established as Fig. 4. The state is used as input, which first passes through the fully connected hidden layer that has l neurons with the output

$$\mathbf{w} = f(\mathbf{M}_1 \mathbf{s} + \mathbf{d}_1), \quad (29)$$

where $\mathbf{M}_1$ is the weight matrix, $\mathbf{d}_1$ is the vector of biases, and f is activation function. The output layer is fully connected and provides the q value of each action by

$$[q(\mathbf{s}, 1), q(\mathbf{s}, 2), \ldots, q(\mathbf{s}, n)]^\top = \mathbf{M}_2 \mathbf{w} + \mathbf{d}_2, \quad (30)$$

where $\mathbf{M}_2$ is the weight matrix and $\mathbf{d}_2$ is the vector of biases.

Unlike common reinforcement learning applications which require repeated exploration to update the q value and obtain an optimal policy, the attack-resilient OPP only needs to explore and identify an action sequence that obtains a high reward. In other words, the action sequences that the agent explored all can be used as solutions, and the scheme with a higher reward will be selected as the final solution. Therefore, the repeated action sequence is not expected in the training process. In this respect, reinforcement learning guided tree search [36] is used to explore environment more efficiently. Fig. 5 shows the detailed structure of the tree search, which begins on the root node denoting the initial state in an episode of the training process. Then, the agent uses ϵ-greedy strategy to select a child node which represents the action and next state obtained from this action according to the q value of each child node, until the leaf node representing the final state is selected. After a trajectory is selected, the leaf node and the nodes having no child nodes of the trajectory will be deleted from bottom to top, which means these nodes are fully explored. Therefore, the tree search ensures that the agent does not explore repeated solutions in the limited solution space.

Algorithm 1 shows the details of the reinforcement learning guided tree search method. The deep reinforcement learning uses two networks to eliminate the data correlation: one is the main network that is updated every step during training, and the other is the target network that provides the objective to be approximated by the main network. In Algorithm 1, Line 1 initializes the experience buffer, the parameter of DQN and the search tree. Line 2 means the training begins in a loop for M episodes. Line 3 means the initial state of each episode starts after the priority placement stage. Line 4 begins exploration of the episode. Line 5 filters the action set according to (25). In Line 6, the agent selects action a based on the tree search algorithm and interacts with environment

Algorithm 1 Reinforcement Learning Guided Tree Search Algorithm

Input: The benchmark system; Discount factor γ;
Output: Action sequence;
1: Initialize the experience buffer D, the DQN parameter ω and ω^- and the search tree;
2: **for** episode $g = 1 : M$ **do**
3: Obtain the state after the priority placement stage;
4: **while** $\mathbf{s} \neq [1, 1, \ldots, 1]^\top$ **do**
5: Filter action set A according to (25);
6: Collect the trajectories' information $[\mathbf{s}, a, r, \mathbf{s}']$; Running the tree search algorithm;
7: Store the experience $[\mathbf{s}, a, r, \mathbf{s}']$ in buffer D;
8: Sample a mini-batch of N experience $[\mathbf{s}_i, a_i, r_i, \mathbf{s}_i']$ from buffer D;
9: $y_i = \begin{cases} r_i, & \text{if } \mathbf{s}_i' = [1, 1, \ldots, 1]^\top \\ r_i + \gamma \max_{a'} q(\mathbf{s}_i', a'; \omega^-), & \text{otherwise.} \end{cases}$
10: Update parameter ω by using gradient rule (31);
11: Reset $\omega^- = \omega$ after every S steps;
12: $\mathbf{s} \leftarrow \mathbf{s}'$.
13: **end while**
14: **end for**

collecting the experience state $\mathbf{s}$, action a, next state $\mathbf{s}_i'$ and reward r. The detailed tree search algorithm is shown in Algorithm 2, which is used to record explored edges and avoid completely expanded nodes. Line 7 stores the experiences in

Algorithm 2 Tree Search Algorithm

Input: The benchmark system; Greedy rate ϵ; The value function q;
Output: Action sequence;
1: Initialize root node n_r;
2: **for** episode $g = 1 : M$ **do**
3: node $n \leftarrow n_r$;
4: **while** node n is not leaf node **do**
5: Use ϵ-greedy strategy to select an action a from the action set of the node n;
6: Take the action a and obtain the child node n_c of n;
7: $n \leftarrow n_c$;
8: **end while**
9: **while** The action set of the node n is empty **do**
10: Delete the action from the action set of the parent node n_p;
11: $n \leftarrow n_p$.
12: **end while**
13: **end for**

buffer D. From Line 8 to Line 9, the algorithm samples N experiences $[\mathbf{s}_i, a_i, r_i, \mathbf{s}_i']$ and calculates y_i according to the equation in Line 9. Line 10 updates the parameters of the main network based on the following gradient rule

$$\omega_{t+1} = \omega_t - \alpha \nabla_{\omega_t} L(\omega_t), \tag{31}$$

where α is the length of the updated step and L is a loss function:

$$L = \frac{1}{N} \sum_{i=1}^{N} (y_i - q(\mathbf{s}_i, a_i; \omega)). \tag{32}$$

The parameters of the target network ω^- is replaced by the parameters of the main network ω after every S steps in Line 11. Line 12 updates the state and starts a new episode.

According to (27), the value function represents the mathematical expectation of subsequent cumulative rewards because taking an action from a state $\mathbf{s}_t$ may obtain different reward r_t and several different states $\mathbf{s}_{t+1}$. Since the probability of each different scenario is usually unknown, the reinforcement learning algorithm sets a step length α and the value function is updated with a little step length α towards the value of the explored experiences as (31). On the other hand, the more experiences the algorithm accumulates, the q value is closer to the mathematical expectation, but certain state $\mathbf{s}_t$ and action a_t yield certain state $\mathbf{s}_{t+1}$ and reward in the environment. Therefore, the step length α can be set close to 1 to accelerate the convergence.

IV. Case Studies

The proposed method is validated on various IEEE standard test systems to verify its effectiveness. All experiments have been done on the Lenovo laptop with 2.7 GHz Intel Core i7-7500U processor and 8 GB RAM on Windows 10 system. The topology information of the benchmark systems are obtained from [41] and the algorithms are implemented on Python 3.7. The discount factor γ is set as 0.96. The greedy rate ϵ is set as 0.6 initially and linearly decreases with the training process. Moreover, the coefficient c is set as 10 and b is set as 2, respectively. The time interval of updating the target network S is set as 200.

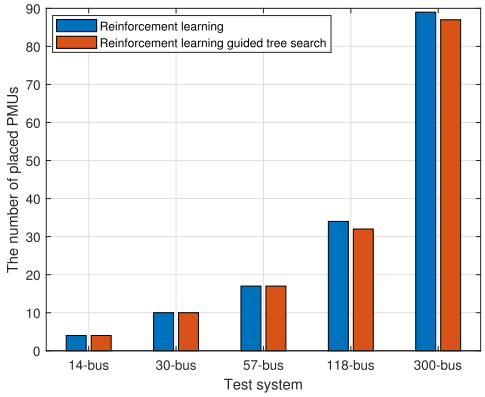


Fig. 6. Comparison between reinforcement learning and the proposed reinforcement learning guided tree search approach.

Table I shows the results of the proposed reinforcement learning guided tree search approach in various standard test systems, including the IEEE 14-bus, 30-bus, 57-bus, 118-bus and 300-bus systems. It can be found that compared with the other works, the proposed method achieves satisfying solutions. Meanwhile, it is observed the number of PMUs achieving the system complete observability is less than one third of the number of smart grid buses. On the other hand, we compare the reinforcement learning and the proposed reinforcement learning guided tree search, and the comparison results are presented in Fig. 6. As the figure shows, the optimal solution can also be obtained only using reinforcement learning approach in IEEE 14-bus, 30-bus and 57-bus system. However, reinforcement learning often falls into local optima for the large-scale power systems, thus the proposed reinforcement learning guided tree search is more effective.

Table II shows the detailed location of the optimal solution in each test system and provides an orderly placement program. Fig. 7 shows the placement process in IEEE 14-bus and 30-bus system respectively according to the Table II, where green indicates PMU placements in the priority placement stage and blue indicates the main placement stage. As shown in Fig. 7, the PMUs placed in priority placement stage to protect vulnerable buses reduces the state space and action space of the reinforcement learning for the main placement stage.

To further verify the effectiveness of the proposed method, the effect of the placement order obtained by the proposed method is compared with random placement and greedy algorithm (GA) in IEEE 30-bus, 57-bus, 118-bus, and 300-bus system, as shown in Fig. 8. We simulate 100 random placement orders in the main stage placement of each standard test system and draw the max number of the observable buses at each step. In addition, we use the GA to select

TABLE I
OPP RESULTS USING THE PROPOSED METHOD AND OTHER METHODS IN IEEE TEST SYSTEMS

Test system	Number of PMUs in priority placement stage	Number of PMUs in main placement stage	Total number	Ref. [42]	Ref. [43]	Ref. [21]
14-bus	1	3	4	4	4	4
30-bus	3	7	10	-	10	10
57-bus	1	16	17	17	17	17
118-bus	6	26	32	32	32	32
300-bus	52	35	87	87	-	87

TABLE II
LOCATIONS OF PMUS IN IEEE TEST SYSTEMS

Test system	Location of vulnerable buses	Location of PMUs in priority stage	Location of PMUs in main stage
14-bus	8	7	6, 2, 9
30-bus	11, 13, 26	12, 9, 25	6, 10, 3, 18, 29, 24, 7
57-bus	33	32	9, 38, 4, 29, 1, 24, 41, 20, 36, 50, 46, 57, 54, 45, 26, 30
118-bus	10, 73, 87, 111, 112, 116, 117	12, 110, 68, 71, 9, 86	80, 17, 49, 23, 56, 37, 105, 94, 101, 53, 77, 90, 62, 85, 5, 34, 64, 115, 21, 25, 40, 75, 29, 45, 2, 11
300-bus	69, 150, 164, 192, 201, 206, 209, 212, 215, 218, 220, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 275, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 292, 293, 295, 296, 297, 298, 299, 300	268, 109, 3, 64, 112, 210, 270, 11, 38, 54, 216, 2, 12, 22, 23, 43, 48, 49, 59, 60, 62, 71, 98, 99, 213, 269, 272, 1, 25, 53, 55, 65, 85, 88, 118, 173, 193, 208, 217, 267, 273, 274, 276, 17, 33, 58, 145, 163, 211, 219, 228, 294	101, 190, 116, 189, 15, 119, 86, 79, 183, 27, 152, 224, 204, 139, 187, 196, 157, 96, 111, 160, 37, 133, 143, 132, 200, 113, 93, 128, 68, 225, 29, 13, 80, 41, 177

TABLE III
LOCATION OF ZERO INJECTIONS AND CONVENTIONAL POWER FLOWS

Test system	Location of zero injections	Location of conventional power flows
14-bus	7	1-2, 9-10, 9-14
30-bus	6, 9, 22, 25, 27, 28	1-3, 6-7, 6-8, 12-13, 16-17, 18-19
57-bus	4, 7, 11, 21, 22, 24, 26, 34, 36, 37, 39, 40, 45, 46, 48	18-19, 27-28, 30-31, 52-53, 54-55
118-bus	5, 9, 30, 37, 38, 63, 64, 68, 71, 81	1-3, 11-13, 16-17, 20-21, 22-23, 27-28, 29-31, 34-43, 35-36, 41-42, 44-45, 46-48, 50-57, 51-52, 53-54, 56-58, 75-118, 77-82, 78-79, 86-87, 90-91, 95-96, 100-101, 110-111, 110-112, 114-115
300-bus	17, 58 ,233, 256, 294	1-3, 3-4, 6-7, 8-11, 11-13, 15-16, 21-22, 24-25, 25-26, 32-35, 37-38, 40-68, 68-174, 46-47, 50-51, 55-56, 70-71, 77-84, 84-86, 95-103, 108-112, 120-125, 136-138, 145-265, 156-157, 160-166, 166-167, 173-198, 198-216, 216-220, 182-190, 184-185, 200-202, 208-209, 88-235, 64-239, 2-248, 17-252, 109-263, 270-292, 270-296, 269-288, 294-300

bus locations at each step, which would increase the number of observable buses the most. In the IEEE 30-bus system, all curves are completely overlapped. Compared with random placement, the placement order of the proposed method can protect more buses in other cases until the complete observability is obtained. Although GA protects more buses at some steps, it will produce more total number of the PMUs to get the complete system observability in the IEEE 57-bus, 118-bus, and 300-bus system. Therefore, our method can get the optimal total number of PMUs and protect as many buses as possible at each step. Moreover, considering the q value in the reinforcement learning may have minor errors caused by insufficient training, we also use GA to decide placement order from PMU locations obtained by the proposed method and the corresponding results are shown in Fig. 8. It can be seen that the proposed method with GA increases the number of observable buses slightly at some steps in the IEEE 57-bus, 118-bus and 300-bus system, which makes the placement order more reasonable.

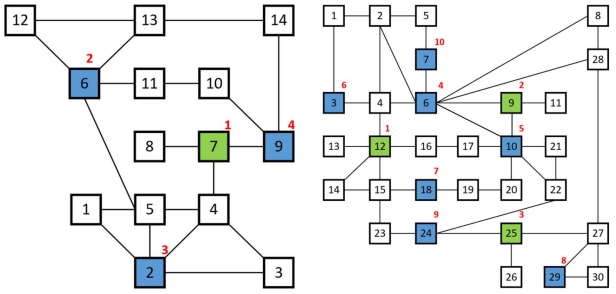
To illustrate the training process, we will use the IEEE 30-bus system as an example. We train DQN for 1,500 episodes. Fig. 9 shows the number of PMUs changes with the training process in IEEE 30-bus system. In the first 800 episodes, the agent selects actions based on the ϵ-greedy algorithm and the search space is explored randomly. After 800 episodes, the number of the PMUs becomes steady and the optimal solution of the OPP problem in the 30-bus system is identified. It can be noted that due to the discount factor, the agent prefers to place PMUs on buses that have higher instant rewards on the front of the sequence.

TABLE IV
OPP RESULTS FOR CASE OF CONSIDERING ZERO INJECTIONS

Test system	Location of PMUs in priority stage	Location of PMUs in main stage	Total number	Ref. [39]	Ref. [42]
14-bus	-	6, 2, 9	3	3	3
30-bus	12	10, 5, 24, 3, 18, 30	7	7	-
57-bus	32	13, 6, 41, 1, 25, 51, 38, 29, 18, 54	11	11	11
118-bus	86, 110, 12	49, 17, 92, 80, 85, 40, 75, 56, 72, 21, 32, 105, 77, 62, 11, 34, 102, 52, 45, 27, 94, 8, 29, 91, 3	28	28	28
300-bus	268, 109, 3, 64, 112, 210, 270, 11, 54, 216, 2, 12, 22, 23, 43, 48, 49, 59, 60, 62, 71, 98, 99, 213, 269, 272, 1, 25, 53, 55, 65, 88, 118, 173, 193, 208, 217, 267, 273, 274, 276, 33, 145, 163, 211, 219, 228	101, 190, 116, 189, 15, 119, 86, 79, 183, 27, 152, 224, 204, 139, 187, 196, 157, 96, 111, 160, 37, 133, 143, 132, 200, 113, 93, 128, 68, 225, 29, 13, 80, 41, 177	82	-	82

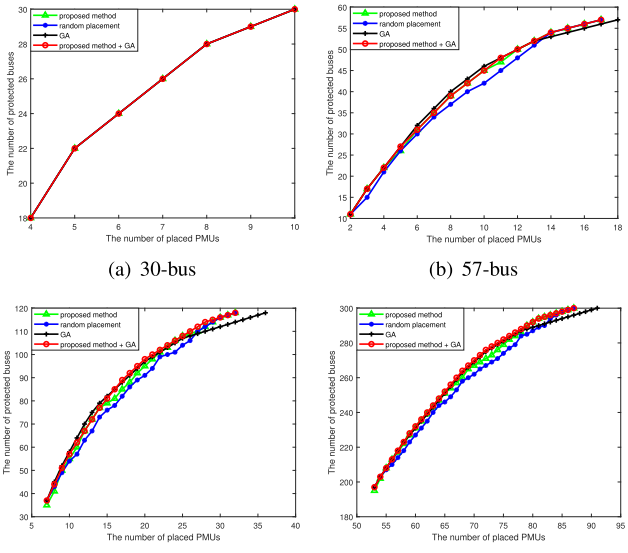
TABLE V
OPP RESULTS FOR CASE OF CONSIDERING ZERO INJECTIONS AND CONVENTIONAL POWER FLOWS

Test system	Location of PMUs in priority stage	Location of PMUs in main stage	Total number	Ref. [44]
14-bus	-	4, 6	2	2
30-bus	-	10, 29, 15, 2	4	4
57-bus	32	13, 6, 41, 1, 25, 51, 38, 29, 54	10	10
118-bus	12	80, 17, 49, 92, 59, 27, 85, 105, 24, 66, 19, 40, 110, 70, 11, 118, 8	18	18
300-bus	268, 3, 64, 112, 210, 270, 11, 54, 216, 2, 12, 22, 23, 43, 48, 49, 59, 60, 62, 71, 98, 99, 213, 269, 272, 1, 25, 53, 55, 65, 118, 173, 193, 217, 267, 273, 274, 276, 33, 163, 211, 219, 228	101, 190, 116, 189, 15, 119, 86, 79, 183, 27, 152, 224, 204, 139, 187, 196, 157, 96, 111, 160, 37, 133, 143, 132, 200, 113, 93, 128, 68, 225, 29, 13, 80, 41, 177	78	-



(a) 14-bus (b) 30-bus

Fig. 7. The process of PMU placement.



(a) 30-bus (b) 57-bus

(c) 118-bus (d) 300-bus

Fig. 8. The effect of placement orders.

Finally, we also test the proposed method for two special cases, i.e., 1) the case of considering zero injections, and 2) the case of considering zero injections and conventional power flows. The locations of zero injections and conventional power flows are shown in Table III. We use the same locations of conventional power flows with [44] (except for IEEE 300-bus system since it is not considered in [44]) and the solutions of the two cases are shown in Table IV and Table V. It can be found that the proposed method works well compared with the results in existing literature.

Also, it is worth mentioning the possibility of a non-least-effort attack where the attacker can increase their efforts and target more vulnerable buses. As the defender is provided the full system observability through the proposed OPP strategy in the network, although the priority stage only finds the optimal locations to observe vulnerable buses through the least-effort FDI attack, the main stage can nonetheless provide the complete system observability and thus also defend against non-least-effort FDI attacks. That is, once the OPP problem is solved and the complete system observability is achieved by the proposed method, the resulting OPP strategy would be effective to defend against the least-effort attack and any non-least-effort attack.

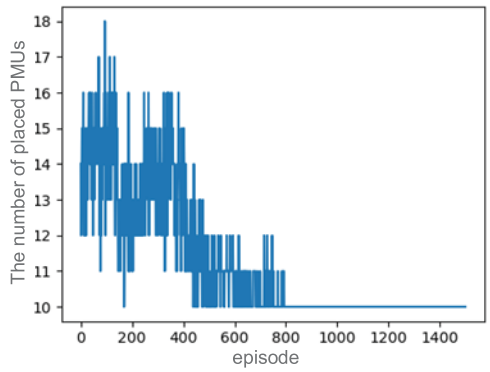


Fig. 9. The number of PMUs placed during the training process in the IEEE 30-bus system.

V. Conclusion

This paper has proposed an attack-resilient OPP strategy which considers to defend potential attacks with limited PMUs. The problem is formulated under a reinforcement learning framework and a dedicated reward function is designed to identify critical buses where PMUs can be installed in multiple stages to ensure attack resilience and system observability in smart grids. The placement is divided into two stages, i.e., the priority placement stage and the main placement stage. The priority placement stage identifies vulnerable buses whose adjacent buses are selected to place PMUs preferentially, while the main placement stage aims to find the optimal locations from remaining buses such that the complete system observability can be guaranteed. Considering the large state space and action space of the environment in large-scale smart grids, a Markov process is designed to describe the OPP problem and the DQN is used to approximate the value function instead of oversized Q-Table. Moreover, the tree search method is utilized to avoid repeated exploration, which helps the reinforcement learning approach to search solutions efficiently. Finally, an orderly OPP sequence is achieved by the proposed method and the test results on various benchmark systems prove the effectiveness of the proposed method. Following this work, we will focus on an important future direction to investigate the case of PMU channel limit. While deciding branches to assign PMU channels will increase the complexity of the learning problem, it may be an important addition given the need to further save the cost of PMU channels in this critical network.

References

[1] A. Monticelli, *State Estimation in Electric Power Systems: A Generalized Approach*. Boston, MA, USA: Springer, 1999.

[2] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Trans. Inf. Syst. Secur.*, vol. 14, no. 13, pp. 1–33, May 2011.

[3] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Sparse attack construction and state estimation in the smart grid: Centralized and distributed models," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1306–1318, Jul. 2013.

[4] J. Hao, R. J. Piechocki, D. Kaleshi, W. H. Chin, and Z. Fan, "Sparse malicious false data injection attacks and defense mechanisms in smart grids," *IEEE Trans. Ind. Informat.*, vol. 11, no. 5, pp. 1–12, Oct. 2015.

[5] M. Jin, J. Lavaei, and K. H. Johansson, "Power grid AC-based state estimation: Vulnerability analysis against cyber attacks," *IEEE Trans. Autom. Control*, vol. 64, no. 5, pp. 1784–1799, May 2018.

[6] M. Zhang *et al.*, "False data injection attacks against smart gird state estimation: Construction, detection and defense," *Sci. China Technol. Sci.*, vol. 62, no. 12, pp. 2077–2087, Dec. 2019.

[7] Y. Yuan, Z. Li, and K. Ren, "Modeling load redistribution attacks in power systems," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 382–390, Jun. 2011.

[8] M. H. R. Koochi, P. Dehghanian, and S. Esmaeili, "PMU placement with channel limitation for faulty line detection in transmission systems," *IEEE Trans. Power Del.*, vol. 35, no. 2, pp. 819–827, Apr. 2019.

[9] S. Kumar, B. Tyagi, V. Kumar, and S. Chohan, "Optimization of phasor measurement units placement under contingency using reliability of network components," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, pp. 9893–9906, Dec. 2020.

[10] N. M. Manousakis and G. N. Korres, "Optimal allocation of phasor measurement units considering various contingencies and measurement redundancy," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 6, pp. 3403–3411, Jun. 2019.

[11] T. A. Alexopoulos, G. N. Korres, and N. M. Manousakis, "Complementarity reformulations for false data injection attacks on PMU-only state estimation," *Electr. Power Syst. Res.*, vol. 189, Dec. 2020, Art. no. 106796.

[12] M. K. Arpanahi, H. H. Alhelou, and P. Siano, "A novel multiobjective OPP for power system small signal stability assessment considering WAMS uncertainties," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3039–3050, May 2019.

[13] A. G. Phadke *et al.*, "Synchronized sampling and phasor measurements for relaying and control," *IEEE Trans. Power Del.*, vol. 9, no. 1, pp. 442–452, Jan. 1994.

[14] C. Pei, Y. Xiao, W. Liang, and X. Han, "PMU placement protection against coordinated false data injection attacks in smart grid," *IEEE Trans. Ind. Appl.*, vol. 56, no. 4, pp. 4381–4393, Aug. 2020.

[15] S. Cui, Z. Han, S. Kar, T. T. Kim, H. Poor, and A. Tajer, "Coordinated data-injection attack and detection in the smart grid: A detailed look at enriching detection solutions," *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 106–115, Sep. 2012.

[16] Q. Yang, L. Jiang, W. Hao, B. Zhou, P. Yang, and Z. Lv, "PMU placement in electric transmission networks for reliable state estimation against false data injection attacks," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1978–1986, Dec. 2017.

[17] F. Aminifar, A. Khodaei, M. Fotuhi-Firuzabad, and M. Shahidehpour, "Contingency-constrained PMU placement in power networks," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 516–523, Feb. 2010.

[18] A. Mahari and H. Seyedi, "Optimal PMU placement for power system observability using BICA, considering measurement redundancy," *Electr. Power Syst. Res.*, vol. 103, pp. 78–85, Oct. 2013.

[19] J. Aghaei, A. Baharvandi, A. Rabiee, and M. A. Akbari, "Probabilistic PMU placement in electric power networks: An MILP-based multiobjective model," *IEEE Trans. Ind. Informat.*, vol. 11, no. 2, pp. 332–341, Apr. 2015.

[20] B. Appasani and D. K. Mohanta, "Co-optimal placement of PMUs and their communication infrastructure for minimization of propagation delay in the WAMS," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2120–2132, May 2018.

[21] N. H. A. Rahman and A. F. Zobaa, "Integrated mutation strategy with modified binary PSO algorithm for optimal PMUs placement," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3124–3133, Dec. 2017.

[22] N. M. Manousakis, G. N. Korres, and P. S. Georgilakis, "Taxonomy of PMU placement methodologies," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 1070–1077, May 2012.

[23] T. T. Kim and H. V. Poor, "Strategic protection against data injection attacks on power grids," *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 326–333, Jun. 2011.

[24] M. Hurtgen and J.-C. Maun, "Optimal PMU placement using iterated local search," *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 8, pp. 857–860, Oct. 2010.

[25] R. F. Nuqui and A. G. Phadke, "Phasor measurement unit placement techniques for complete and incomplete observability," *IEEE Trans. Power Del.*, vol. 20, no. 4, pp. 2381–2388, Oct. 2005.

[26] R. F. Nuqui, A. G. Phadke, R. P. Schulz, and N. Bhatt, "Fast on-line voltage security monitoring using synchronized phasor measurements and decision trees," in *Proc. IEEE Power Eng. Soc. Winter Meeting Conf.*, vol. 3, Feb. 2001, pp. 1347–1352.

[27] F. Aminifar, C. Lucas, A. Khodaei, and M. Fotuhi-Firuzabad, "Optimal placement of phasor measurement units using immunity genetic algorithm," *IEEE Trans. Power Del.*, vol. 24, no. 3, pp. 1014–1020, Jul. 2009.

[28] M. Hajian, A. M. Ranjbar, T. Amraee, and A. R. Shirani, "Optimal placement of phasor measurement units: Particle swarm optimization approach," in *Proc. Int. Conf. Intell. Syst. Appl. Power Syst.*, Nov. 2007, pp. 1–6.
[29] B. Milosevic and M. Begovic, "Nondominated sorting genetic algorithm for optimal phasor measurement placement," *IEEE Trans. Power Syst.*, vol. 18, no. 1, pp. 69–75, Feb. 2003.
[30] F. J. Marin, F. Garcia-Lagos, G. Joya, and F. Sandoval, "Genetic algorithms for optimal placement of phasor measurement units in electric networks," *Electron. Lett.*, vol. 39, no. 19, pp. 1403–1405, 2003.
[31] J. Peng, Y. Sun, and H. F. Wang, "Optimal PMU placement for full network observability using tabu search algorithm," *Int. J. Electr. Power Energy Syst.*, vol. 28, no. 4, pp. 223–231, May 2006.
[32] B. Xu and A. Abur, "Observability analysis and measurement placement for systems with PMUs," in *Proc. IEEE PES Power Syst. Conf. Expo.*, Oct. 2004, pp. 943–946.
[33] R. Sodhi, S. Srivastava, and S. Singh, "Multi-criteria decision-making approach for multistage optimal placement of phasor measurement units," *IET Gener. Transmiss. Distrib.*, vol. 5, no. 2, pp. 181–190, 2011.
[34] D. Dua, S. Damphare, R. K. Gajbhiye, and S. A. Soman, "Optimal multistage scheduling of PMU placement: An ILP approach," *IEEE Trans. Power Del.*, vol. 23, no. 4, pp. 1812–1820, Oct. 2008.
[35] Q. Yang, D. An, R. Min, W. Yu, X. Yang, and W. Zhao, "On optimal PMU placement-based defense against data integrity attacks in smart grid," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 7, pp. 1735–1750, Jul. 2017.
[36] R. Simmons-Edler, A. Miltner, and S. Seung, "Program synthesis through reinforcement learning guided tree search," 2018, *arXiv:1806.02932*.
[37] R. Deng, G. Xiao, and R. Lu, "Defending against false data injection attacks on power system state estimation," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 198–207, Feb. 2015.
[38] A. Abur and A. G. Exposito, *Power System State Estimation: Theory and Implementation*. Boca Raton, FL, USA: CRC Press, 2004.
[39] N. H. A. Rahman and A. F. Zobaa, "Optimal PMU placement using topology transformation method in power systems," *J. Adv. Res.*, vol. 7, no. 5, pp. 625–634, Sep. 2016.
[40] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.
[41] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2010.
[42] A. Almunif and L. Fan, "Optimal PMU placement for modeling power grid observability with mathematical programming methods," *Int. Trans. Electr. Energy Syst.*, vol. 30, no. 2, p. e12182, Feb. 2020.
[43] B. K. Saha Roy, A. K. Sinha, and A. K. Pradhan, "An optimal PMU placement technique for power system observability," *Int. J. Electr. Power Energy Syst.*, vol. 42, no. 1, pp. 71–77, Nov. 2012.
[44] V.-K. Tran and H.-S. Zhang, "Optimal PMU placement using modified greedy algorithm," *J. Control, Autom. Electr. Syst.*, vol. 29, no. 1, pp. 99–109, Feb. 2018.

Meng Zhang received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, in 2013, and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2018. He is currently an Associate Professor with the School of Electronic and Information Engineering, Xi'an Jiaotong University. His research interests include nonlinear control, filtering, and cyber-physical systems.

Zhuorui Wu received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, in 2021, where he is currently pursuing the M.S. degree with the School of Electronic and Information Engineering. His research interests include smart grid security, system optimization, and cyber-physical systems. He was a recipient of the IEEE International Conference on Industrial Cyber-Physical Systems (ICPS) Best Student Paper Award in 2021.

Jun Yan (Member, IEEE) received the B.Eng. degree in information and communication engineering from Zhejiang University, China, in 2011, and the M.S. and Ph.D. degrees (Hons.) in electrical engineering from The University of Rhode Island, Kingston, RI, USA, in 2013 and 2017, respectively.

He is currently an Assistant Professor with the Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC, Canada. His research focuses on computational intelligence and cyber-physical security, with applications in smart grids, smart cities, and other smart critical infrastructures. He was a recipient of the IEEE International Conference on Communications (ICC) Best Paper Award in 2014 and the IEEE International Joint Conference on Neural Networks (IJCNN) Best Student Paper Award in 2016.

Rongxing Lu (Fellow, IEEE) received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, Canada, in 2012. He is currently a Mastercard IoT Research Chair, a University Research Scholar, and an Associate Professor with the Faculty of Computer Science (FCS), University of New Brunswick (UNB), Canada. Before that, he worked as an Assistant Professor at the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore, from April 2013 to August 2016. He worked as a Post-Doctoral Fellow at the University of Waterloo from May 2012 to April 2013. His research interests include applied cryptography, privacy enhancing technologies, and the IoT-big data security and privacy. He has published extensively in his areas of expertise. He was awarded the most prestigious "Governor General's Gold Medal" for his Ph.D. degree. He won the 8th IEEE Communications Society (ComSoc) Asia–Pacific (AP) Outstanding Young Researcher Award in 2013. He was a recipient of nine best (student) paper awards from some reputable journals and conferences. He is the Winner of the 2016–2017 Excellence in Teaching Award, FCS, UNB. Currently, he serves as the Chair for IEEE Communications and Information Security Technical Committee (IEEE ComSoc CIS-TC) and the Founding Co-Chair for IEEE TEMS Blockchain and Distributed Ledgers Technologies Technical Committee (BDLT-TC).

Xiaohong Guan (Fellow, IEEE) received the B.S. and M.S. degrees in control engineering from Tsinghua University, Beijing, China, in 1982 and 1985, and the Ph.D. degree in electrical and systems engineering from the University of Connecticut in 1993.

He is currently a Professor with the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, China. He was appointed as a Cheung Kong Professor of systems engineering in 1999 and the Dean of the Faculty of Electronic and Information Engineering in 2008. He has been the Director of the Center for Intelligent and Networked Systems, Tsinghua University, since 2001. He was the Head of the Department of Automation from 2003 to 2008. His research interests include economics and security of networked systems, optimization-based planning and scheduling of power and energy systems, manufacturing systems, and cyber-physical systems, including smart grid and sensor networks.

He is a member of Chinese Academy of Science. He is serving as an Editor for IEEE TRANSACTIONS ON SMART GRID.