applied sciences

MDPI

Article

MADDPG-Based Security Situational Awareness for Smart Grid with Intelligent Edge

Wenxin Lei [1,2], Hong Wen [1,2,*], Jinsong Wu [3,4,*] and Wenjing Hou [1,2]

1 School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu 611731, China; leiwenxin@std.uestc.edu.cn (W.L.); uestc_hwj@126.com (W.H.)
2 Aircraft Swarm Intelligent Sensing and Cooperative Control Key Laboratory of Sichuan Province, UESTC, Chengdu 611731, China
3 School of Artificial Intelligence, Guilin University of Electronic Technology, Guilin 541004, China
4 Department of Electrical Engineering, Universidad de Chile, Santiago 8370451, Chile
* Correspondence: sunlike@uestc.edu.cn (H.W.); wujs@ieee.org (J.W.)

Abstract: Advanced communication and information technologies enable smart grids to be more intelligent and automated, although many security issues are emerging. Security situational awareness (SSA) has been envisioned as a potential approach to provide safe services for power systems' operation. However, in the power cloud master station mode, massive heterogeneous power terminals make SSA complicated, and failure information cannot be promptly delivered. Moreover, the dynamic and continuous situational space also increases the challenges of SSA. By taking advantages of edge intelligence, this paper introduces edge computing between terminals and the cloud to address the drawbacks of the traditional power cloud paradigm. Moreover, a deep reinforcement learning algorithm based on the edge computing paradigm of multiagent deep deterministic policy gradient (MADDPG) is proposed. The minimum processing cost under the premise of minimum detection error rate is taken to analyze the smart grids' SSA. Performance evaluations show that the algorithm under this paradigm can achieve faster convergence and the optimal goal, namely the provision of real-time protection for smart grids.

Keywords: smart grid; situational awareness; edge computing; multi-agent DDPG; deep reinforcement learning

Citation: Lei, W.; Wen, H.; Wu, J.; Hou, W. MADDPG-Based Security Situational Awareness for Smart Grid with Intelligent Edge. *Appl. Sci.* **2021**, *11*, 3101. https://doi.org/10.3390/app11073101

Academic Editor: Andreas Sumper

Received: 30 January 2021
Accepted: 26 March 2021
Published: 31 March 2021

1. Introduction

With the rapid development of wireless communication and the internet of things (IoT) technologies, the explosive growth of intelligent terminals access to smart grids [1,2]. By taking advantages of massive data generated from a huge number of intelligent terminals, data-driven control can leverage these data to identify different situations for making decisions, which is vital for production and operation in smart grids [3,4].

However, while these new technologies facilitate smart grid services, they may also bring new security challenge issues [5–7]. Attacks in the power IoT environment have increased substantially in both number and variety, causing significant damage to the grid. For instance, a cyber threat on the Ukrainian power grid left 225,000 people powerless for several days in December 2015 [8]. A physical attack on a California substation in April 2014 severely impacted the operation of the grid [9], and the economic and environmental losses caused by such attacks are immeasurable. These examples show that the threat problem of the grid exists. If not handled properly, it will have a significant impact on people's productive life. Therefore, a smart grid with high-security requirements should be able to monitor fault information during its operation. Before an attack happens, it is necessary to automatically cut off the threat in time and repair the fault automatically to ensure the grid system's stable operation.

Today, security situational awareness (SSA) and prediction are essential functions and necessary measures to satisfy the power system's process in smart grids' information management system [10,11]. The application and development of network SSA technology in smart grids enables the acquisition, understanding, and prediction of various security factors. Moreover, it can accurately grasp the grid's security situation and achieve proactive prevention of grid security threats.

Most SSA research works for smart grids currently have relied on data acquiring devices that gather information from various power terminals or sensors [12,13]. Then this collected information would be uploaded to the power master station for further aggregation and analysis to obtain security measures applicable to multiple power systems [14–16]. However, the deployment of actual SSA technology faces many critical issues in smart grids.

(1) With massive heterogeneous power terminals and distinct communication protocols, causing interconversion and interoperability trouble in network communication. Moreover, there are considerable challenges in heterogeneous network deployment and configuration, network management, and maintenance. Therefore, the effective deployment of the massive heterogeneous terminals is necessary.
(2) The smart grid control system must always be up to running and should be addressed promptly in the face of real-time threats. SSA technology in smart grids master station based on cloud computing does not satisfy the requirement of dealing with real-time threats. Therefore, low latency SSA technology is necessary.
(3) The network of smart grids is particular and requires exceptionally high-security performance. Once attacked, it will face huge losses. Therefore, high-efficiency and high-precision SSA technology are necessary.

In such cases, the integration of edge computing and artificial intelligence (AI) into SSA technologies for smart grids can solve the above problems. Edge computing [17] enables an open platform combining connectivity, computation, storage, and application at the edge side of the system's network to provide edge intelligence services for data from power sensing nodes nearby. Deploying corresponding intelligent edge agents for different power terminal clusters solves massive heterogeneous networks and transmission and processing delays. For the security of the smart grid, the existing power system situational awareness mainly adopts the defense means of automatically detecting and then disconnecting or replacing the faulty electronic components. This is beneficial to the problem of reducing maintenance time. However, smarter autonomous threat prevention and improved self-checking and self-healing capabilities of smart grids should also be considered. Deep reinforcement learning (DRL) combines the awareness capabilities of deep learning and the decision-making capabilities of reinforcement learning, allowing control directly from the information provided by the environment. It is an AI approach closer to the human way of thinking. On top of providing detection capabilities, DRL is also able to autonomously move the system's safety action strategy toward the greatest long-term rewards by learning from the awareness of the environment. In industrial IoT, DRL is increasingly considered to handle high-complexity optimization problems [18,19] and as able to provide efficient SSA solutions in a timely manner by supporting computational resources located at the edge.

In this paper, a multiagent deep deterministic policy gradient (MADDPG) algorithm for smart grids' SSA under edge computing is proposed. The framework integrates edge computing and deep reinforcement learning (DRL) to direct efficient SSA deployment in smart grids, and enable the edge computing-based grid system to make long-term expectation-maximizing actions against security threats through the setting of reward values to achieve security situational awareness. Some of the main contributions in this paper are summarized as follows.

(1) By implementing an edge computing paradigm in smart grids' SSA architecture, issues of massive heterogeneous connections to power terminals and low latency of SSA strategies are solved.

(2) The multiagent-based deep reinforcement learning MADDPG algorithm contributes to handling the continuous and dynamic situation space in smart grids.
(3) The proposed algorithm achieves the minimum processing cost of SSA with the premise of minimum detection error rate.
(4) This paper is the first work to incorporate the edge computing paradigm and DRL in SSA for smart grids.

The remainder of this paper is organized as follows. In Section 2, we present the related work on SSA and AI combined with edge computing. Models of SSA, edge computing, and DRL are introduced in Section 3. In Section 4, we formulate the optimization problem of security situational awareness under the smart grids. The MADDPG-based SSA algorithm is proposed in Section 5. In Section 6, we perform the performance evaluation. Conclusions are given in Section 7.

2. Related Works

In this section, we first review the related work on SSA research in smart grids. Then, some relevant research works on AI-enabled edge computing are discussed.

2.1. Security Situational Awareness

Over the literature, security situational awareness has been an active research topic for recent years. There are already many promising research results in smart grids. He et al. [20] suggested that SSA is key to the safe operation of smart grids. Inadequate SSA could jeopardize the stability of the grid system and cause significant undesirable effects. The literature [21] pointed out that the scale and complexity of the future smart grids will continue to increase, and a statistical metric system based on statistical quantities is proposed to meet the challenge of high dimensional complexity of the situational elements. Wu et al. [22] indicated that cybersecurity threats extend from computer networks to smart grids and proposes an SSA mechanism based on big data analysis to improve awareness efficiency. An SSA approach that utilized consensus decision information from multiple power terminals to enhance system integrity protection has been proposed in [23]. This approach has been employed to address the impact of a single point of failure caused by a network attack on power system integrity.

Most of the works mentioned above have improved the smart grid systems' SSA from an individual perspective. However, they have ignored the awareness environment's continuity and the SSA deployment scheme with dynamically changing environmental information.

2.2. Artificial Intelligence Enabled Edge Computing

Due to the emergence of AI, recent works have been done to investigate how to design efficient and secure smart applications based on AI algorithms in edge computing. Libri et al. [24] deployed high-resolution IoT monitoring devices driven by AI to detect anomalies and perform security analysis in an emerging edge computing paradigm for IoT monitoring systems. Wang et al. [25] addressed the challenge of invalidating the acquired data when smart sensors were attacked, exploiting mobile edge computing nodes with computational resources and storage capacity. Thus, an AI-based trust assessment and management mechanism have been proposed to secure the sensors. To conquer smart grids' shortcomings under cloud computing platform, an edge computing paradigm under IoT applied in smart grids is proposed in [26]. AI algorithms facilitate the real-time analysis of smart grids' data and privacy protection.

Each of the above works has addressed security issues in related fields through an AI additive edge computing model. However, the research on SSA under AI-based edge computing has not been covered.

3. System Model

3.1. Security Situational Awareness in Smart Grids

In the network environment of smart grids systems, the acquired information includes grid system dynamic data, power terminal's security states, network topology, operation environments, temporary malfunction, and steady-state operation, etc. By leveraging network SSA technology, the current security state of the network system is accurately monitored and predicts future security development trends. Active and effective defense and countermeasures are available in advance to deal with the upcoming large-scale attacks. It will significantly transform the unfavorable situation of after-the-fact handling and passive protection of power network security management over the past.

We consider the wide-area situational awareness (WASA) [27] technology to acquire and combine information on the smart grids' situational elements. Specifically, it includes external attack information, the internal vulnerability of power terminal nodes, and threat awareness of possible attacks, as shown in Figure 1.

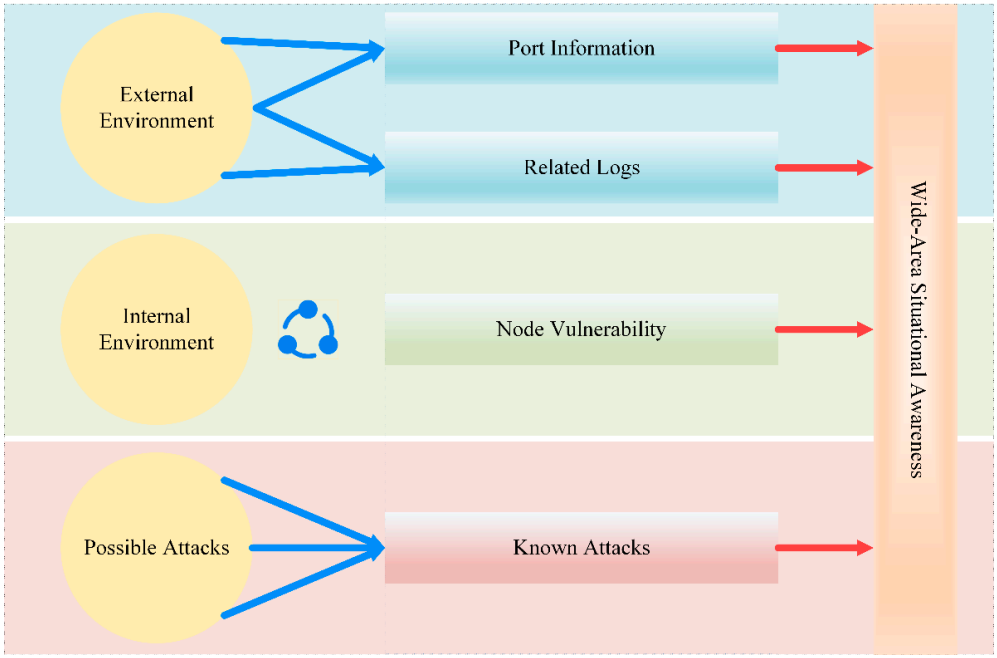

Figure 1. Wide-Area Situational Awareness (WASA).

The external attacks mainly consist of the network port traffic information and the logs of various power terminals. When the power terminal's service port encounters continuous malicious connection requests, the port will not respond to new legitimate connection requests rapidly, causing the power system to fail to provide standard network services. Port traffic information can timely reflect port threat information. Also, log information records the operation and usage of power terminals, including alarm information of abnormal events. As the topological distribution of a massive number of power terminals forms the power network, the vulnerability of the power terminal nodes in the network constitutes the internal environmental elements. The external and internal elements are considered from the perspective of the power terminal itself, and this information is accessible from the terminal directly or from the network constituted by the terminals. However, there may still be deliberate attacks by attackers in the physical grid, such as the injection of malware, viruses, and advanced persistent threats, etc., to the power information management platform. Therefore, the behavior of possible attacks is also necessary as situational elements. By accurately grasping the behavior of potential attacks, it is capable of suppressing threats. The known types of possible attacks can have a large impact on the power grid. Controlling the information of situational elements in the smart

grids is crucial to the analysis of security situational awareness. Based on those situational elements, one can obtain assessment results, to accurately grasp the smart grids' security situation, and provide a basis for an active defense against threats.

3.2. Edge Computing

The edge computing paradigm is presented in the background of the fast evolution of IoT networks due to the traditional cloud computing paradigm is unable to solve the explosive growth of information and data [28]. By placing edge computing agents between the terminals and the cloud, partial computing tasks from the original cloud center are transferred to the vicinity of the data source for execution. This new network paradigm reduces data transmission's physical path and enables timely response to terminals' task requests. In other words, the computing and storage capacity of multiple edge agents shares the pressure of traditional cloud computing [29]. As we know, the architecture of edge computing is generally composed of three layers, including a terminals layer, edge computing layer, and cloud computing layer [30].

Edge computing, as an emerging network infrastructure, empowers new capabilities for SSA in smart grids. Figure 2 shows the scenario of applying SSA under smart grids based on the edge computing paradigm. The power terminal layer includes primary power devices equipped with smart sensors, microgrid facilities, and intelligent charging piles. At edge SSA layer consists of edge agents that acquire the terminal layer's situational elements for awareness. Then, the edge SSA layer collaborates with the power cloud master station layer to accomplish SSA tasks.

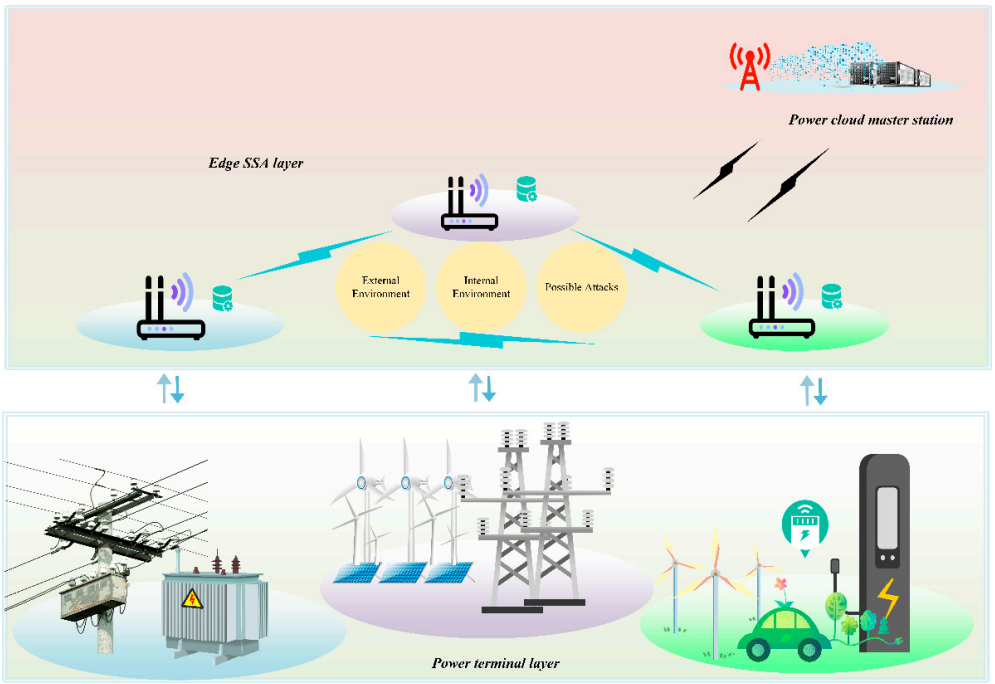

Figure 2. Edge computing paradigm for smart grids security situational awareness.

Since the edge agents are close to the power terminals, they possess the capability to interconnect a massive number of heterogeneous terminals. Task processing latency is dramatically decreased by the fact that data acquisition and analysis are carried out directly at the edge and avoid potential congestion of situational elements in other parts of the network. Practical deployment of edge computing has naturally decentralized characteristics. Such enable distributed computing and storage, dynamic scheduling, unified management of resources, thus possessing distributed security capabilities.

3.3. Deep Reinforcement Learning

Traditional machine learning mainly focuses on solving the mapping relationship between inputs and outputs, describing this mapping's error as a loss function with coefficients to be determined [31]. Then, the value of the loss function with minimal error is solved by optimization ideas. However, in SSA of smart grids under edge computing, the environment contains essential situational elements. How to acquire these situational elements from the environment and train them to obtain accurate awareness ability is our research's primary concern.

Reinforcement learning (RL) is environment-based, enabling an agent to choose actions based on the current state and thus obtain as much rewards from the environment as possible [32]. Since SSA under smart grids involves a continuous space of situational elements and perceptual interaction of multiple edge agents, MADDPG is a promising solution. As shown in Figure 3, MADDPG involves the respective actor-critic networks of K agents. Each agent has an online actor and critic network, respectively, and the same for the target network.

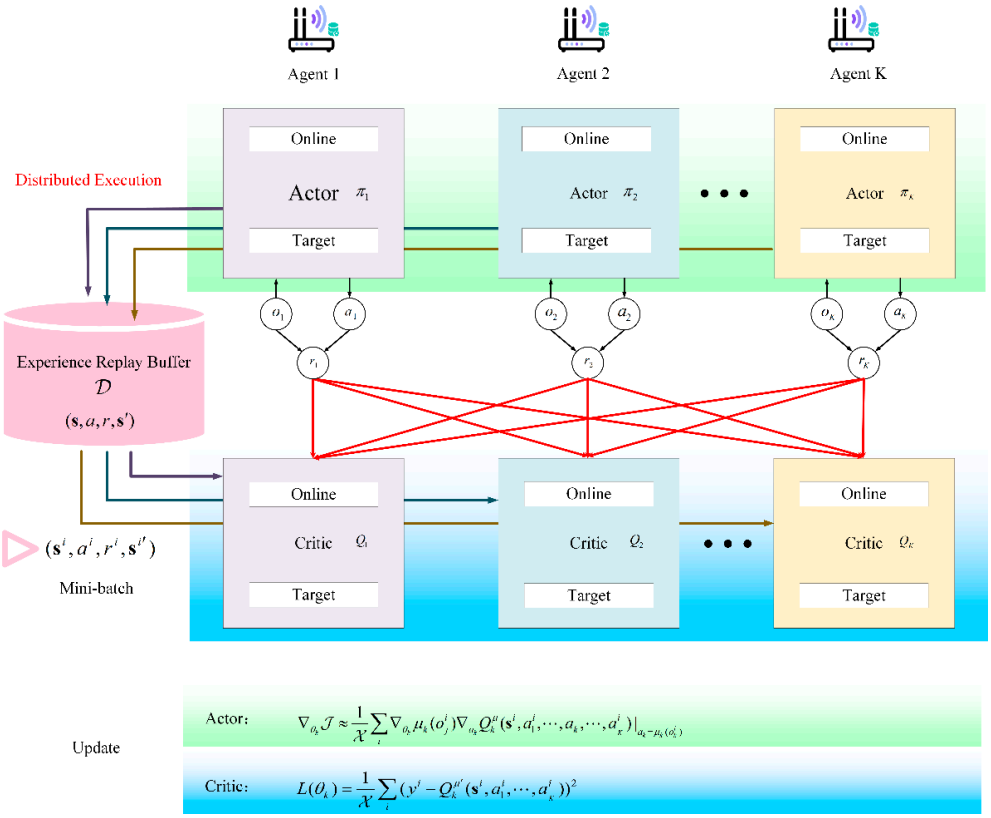

Figure 3. Multiagent deep deterministic policy gradient (MADDPG) framework.

First, each agent distributed executes the acquisition of current state $\mathbf{s}$, action value a, rewards r, and next state $\mathbf{s}^{'}$, and then deposits the sequence $(\mathbf{s}, a, r, \mathbf{s}^{'})$ in the experience replay buffer $\mathcal{D}$. When the number of caches in $\mathcal{D}$ is greater than a threshold, the network starts learning. Each actor-network updates the policy parameters individually according to maximizing the gradient ascent. Here, $\pi = \{\pi_1, \pi_2, \ldots, \pi_K\}$ is denoted as the policy for K agents. Then, each critic-network updates the parameters of action value by minimizing the distance between Q-function y^i of samples selected from $\mathcal{X}$ and the state-action function $Q_k^{\mu'}(\mathbf{s}^i, a_1^i, \cdots, a_K^i)$, respectively.

4. Problem Formulation

In the framework of SSA with edge computing, we define the situational elements obtained by an edge agent in time slot t as $O(t) = \{O_1(t), \ldots, O_k(t), \ldots, O_K(t)\}$, where $O_k(t)$ denotes the situational elements obtained by edge agent k in time slot t and K is the number of edge agents. For edge agent k, $k = 1, 2, \ldots, K$, we divide the situational elements in time slot t into three dimensions.

The first dimension is the information of the external environment elements, here denoted as $O_k^{ex}(t)$, $O_k^{ex}(t) = \{O_k^{exPot}(t), O_k^{exLog}(t)\}$. Note that $O_k^{exPot}(t)$ is the traffic information of ports. Typically, the port traffic information is continuous. Therefore, we set the number of ports here as $P(t)$, then $O_k^{exPot}(t) = \{O_{k1}^{exPot}(t), O_{k2}^{exPot}(t), \ldots, O_{kP(t)}^{exPot}(t)\}$. For each of the available ports,

$$O_{kp(t)}^{exPot}(t) \in [0,1],\ p(t) = \{1, 2, \ldots, P(t)\}. \tag{1}$$

The power terminals have the function of recording operation logs, and similarly, where $O_k^{exLog}(t)$ indicates log information such as overvoltage or overcurrent signals, alarm events, etc. In contrast to port traffic information, there are only two states of events, happening or not. Accordingly, let the total number of events extracted from the logs be $E(t)$, then $O_k^{exLog}(t) = \{O_{k1}^{exLog}(t), O_{k2}^{exLog}(t), \ldots, O_{kE(t)}^{exLog}(t)\}$. For each type of event,

$$O_{ke(t)}^{exLog}(t) \in \{0,1\},\ e(t) = \{1, 2, \ldots, E(t)\}. \tag{2}$$

The second dimension is the internal environment elements, including internal nodes vulnerability information, denoted as $O_k^{in}(t)$, and we define the number of internal power terminal nodes as $N(t)$. As vulnerability of a node characterizes a weak component in the grid, the vulnerability measurement protects against a chain of breakdowns. Consequently, we adopt continuous variables to signify the vulnerability of nodes. Thus, the internal environment elements of the edge agent k at time t is represented as $O_k^{in}(t) = \{O_{k1}^{in}(t), O_{k2}^{in}(t), \ldots, O_{kN(t)}^{in}(t)\}$. For each specific node, the vulnerability is denoted as

$$O_{kn(t)}^{in}(t) \in [0,1],\ n(t) = \{1, 2, \ldots, N(t)\}. \tag{3}$$

The third dimension is the possible attacks, denoted by $O_k^{At}(t)$. Considering the attacks that the smart grid has encountered, we regard them as known attacks and form an attack library to determine the new attacks. Meanwhile, the known attacks in this library work as possible attacks that the grid will encounter. Therefore, we define the categories of possible attacks as $A(t)$. Thus $O_k^{At}(t) = \{O_{k1}^{At}(t), O_{k2}^{At}(t), \ldots, O_{kA(t)}^{At}(t)\}$. Considering that the possible attacks can be blocked in-process, we define continuous variables to represent information about the situation's elements where a particular possible attack is encountered, denoted as

$$O_{ka(t)}^{At}(t) \in [0,1],\ a(t) = \{1, 2, \ldots, A(t)\}. \tag{4}$$

Situational elements under edge computing provide an essential basis for situational decisions. However, the defense tools are generally distinct for different threats. When a threat is detected, we should immediately activate the defense mechanism to suppress and offset the attack's impact on smart grids. However, these defense mechanisms come at a cost. $\mathcal{C}_{kp(t)}^{exPot}(t)$, $\mathcal{C}_{ke(t)}^{exLog}(t)$, $\mathcal{C}_{kn(t)}^{in}(t)$, and $\mathcal{C}_{ka(t)}^{At}(t)$ represent the cost of processing anomalies traffic on port $p(t)$, alarm event $e(t)$ in the logs, restoring internally vulnerable nodes $n(t)$, and defending against possible attack's category $a(t)$ at time slot t, respectively. Here, $\mathcal{A}_{bk}^{p(t)}$, $\mathcal{A}_{bk}^{e(t)}$, $\mathcal{A}_{bk}^{n(t)}$, and $\mathcal{A}_{bk}^{a(t)}$ denote the anomalous situational behaviors detected by the edge agent k at time slot t, respectively. In particular, at time slot t, define $\mathcal{A}_{bk}^{p(t)} = 1$ to indicate that the anomaly traffic is detected for port $p(t)$, and $\mathcal{A}_{bk}^{p(t)} = 0$ otherwise; $\mathcal{A}_{bk}^{e(t)} = 1$ to

indicate that anomalous event $e(t)$ is detected, and $\mathcal{A}_{bk}^{e(t)} = 0$ otherwise; $\mathcal{A}_{bk}^{n(t)} = 1$ to indicate that vulnerable node $n(t)$ is detected, and $\mathcal{A}_{bk}^{n(t)} = 0$ otherwise; and $\mathcal{A}_{bk}^{a(t)} = 1$ to indicate that the possible attack's category $a(t)$ is detected, and $\mathcal{A}_{bk}^{a(t)} = 0$ otherwise. Then, the cost of the SSA's processing under the edge agent k at time slot t is

$$\mathcal{C}(t) = \sum_{p(t)=1}^{P(t)} \mathcal{A}_{bk}^{p(t)} \mathcal{C}_{kp(t)}^{exPot}(t) + \sum_{e(t)=1}^{E(t)} \mathcal{A}_{bk}^{e(t)} \mathcal{C}_{ke(t)}^{exLog}(t) + \sum_{n(t)=1}^{N(t)} \mathcal{A}_{bk}^{n(t)} \mathcal{C}_{kn(t)}^{in}(t) + \sum_{a(t)=1}^{A(t)} \mathcal{A}_{bk}^{a(t)} \mathcal{C}_{ka(t)}^{At}(t). \tag{5}$$

Assume that in the power terminals under the jurisdiction of time slot t, edge agent k, the actual case of port $p'(t)$, if the traffic information is abnormal, $\mathcal{N}_k^{p'(t)}(t) = 1$, and otherwise 0; similarly, $\mathcal{N}_k^{e'(t)}(t) = 1$, $\mathcal{N}_k^{n'(t)}(t) = 1$, and $\mathcal{N}_k^{a'(t)}(t) = 1$ denote the alarm event $e'(t)$, vulnerable node $n'(t)$, and the possible attack $a'(t)$ is existing in the actual case, respectively. Then, the error rate of anomaly traffic port detection for external environment information is expressed as

$$\mathcal{E}_k^{exPot} = 1 - \frac{\sum_{p(t)=1}^{P(t)} \Delta_{p(t)}}{P(t)}. \tag{6}$$

If $\mathcal{A}_{bk}^{p(t)} = \mathcal{N}_k^{p'(t)}(t)$, then $\Delta_{p(t)} = 1$, and inversely, $\Delta_{p(t)} = 0$. The detection error rate of alarm events for external environment information is expressed as

$$\mathcal{E}_k^{exLog} = 1 - \frac{\sum_{e(t)=1}^{E(t)} \Delta_{e(t)}}{E(t)}. \tag{7}$$

If $\mathcal{A}_{bk}^{e(t)} = \mathcal{N}_k^{e'(t)}(t)$, then $\Delta_{e(t)} = 1$, and inversely, $\Delta_{e(t)} = 0$. The detection error rate of vulnerable nodes in the internal environment is expressed as

$$\mathcal{E}_k^{in} = 1 - \frac{\sum_{n(t)=1}^{N(t)} \Delta_{n(t)}}{N(t)}. \tag{8}$$

If $\mathcal{A}_{bk}^{n(t)} = \mathcal{N}_k^{n'(t)}(t)$, then $\Delta_{n(t)} = 1$, and inversely, $\Delta_{n(t)} = 0$. The detection error rate of the possible attack is expressed as

$$\mathcal{E}_k^{At} = 1 - \frac{\sum_{a(t)=1}^{A(t)} \Delta_{a(t)}}{A(t)}. \tag{9}$$

Similarly, if $\mathcal{A}_{bk}^{a(t)} = \mathcal{N}_k^{a'(t)}(t)$, then $\Delta_{a(t)} = 1$, and inversely, $\Delta_{a(t)} = 0$.

The security situational awareness for smart grids under edge computing (SSASGEC) problem to minimize the processing cost under low-error rate by combining situational elements in different dimensions is as follows:

$$\min_{t \in [0,T]} \mathcal{C}(t) \tag{10}$$

$$s.t.\ (1) - (4) \tag{10a}$$

$$\begin{cases} \mathcal{A}_{bk}^{p(t)}, \mathcal{A}_{bk}^{e(t)}, \mathcal{A}_{bk}^{n(t)}, \mathcal{A}_{bk}^{a(t)} \in \{0,1\}, \forall\ k \in \{1,2,\ldots,K\} & (10b) \\ \mathcal{E}_k^{exPot} < \varepsilon_{k,\exp}^{exPot}, \forall\ k \in \{1,2,\ldots,K\} & (10c) \\ \mathcal{E}_k^{exLog} < \varepsilon_{k,\exp}^{exLog}, \forall\ k \in \{1,2,\ldots,K\} & (10d) \\ \mathcal{E}_k^{in} < \varepsilon_{k,\exp}^{in}, \forall\ k \in \{1,2,\ldots,K\} & (10e) \\ \mathcal{E}_k^{At} < \varepsilon_{k,\exp}^{At}, \forall\ k \in \{1,2,\ldots,K\}\ , & (10f) \end{cases}$$

where $\varepsilon_{k,\exp}^{exPot}$, $\varepsilon_{k,\exp}^{exLog}$, $\varepsilon_{k,\exp}^{in}$, and $\varepsilon_{k,\exp}^{At}$ denote the expectation detection error rate, respectively. (10a) represents the information of the current moment of the situational elements. (10b) indicates the situation awareness variables. (10c), (10d), (10e), and (10f) denote the four threat detections' error rates under the three situational dimensions that should not surpass our expectation.

5. MADDPG-Based Security Situational Awareness

Traditional methods are not applicable to rapidly solve the above formulated optimization problem for the following reasons,

(1) This problem is a mixed-integer type of programming problem.
(2) The anomalies detected by situational awareness and the optimization goals are associated with all current situational elements.
(3) The detection error rate and the processing cost are coupled, and the complexity of the computation increases with the number of power terminals and situational elements.
(4) Considering the dynamic changes of the situational elements and the smart grids' high security requirements, the formulated problem needs to be solved rapidly.

Therefore, we adopt a DRL approach. Considering the advantages of utilizing distributed intelligent edge in smart grids scenario, we design a multiagent-based DRL approach. Each edge server acts as an agent to acquire information on the power terminals' situational elements under its coverage area. In particular, we initially reconstruct the smart grids' SSA model under edge computing as an extension of multi-agent Markov decision making (MDP). In the following, an approach to the problem based on the MADDPG algorithm is described.

5.1. Transformation of the Problem

By interacting with the environment many times, each edge agent accumulates a certain amount of experience in awareness of the situation elements. As a Markov process, we denote it as $(s_k(t), a_k(t), r_k(t), s_{k+1}(t))$. Here, $s_k(t) \in \mathcal{S}$, denotes the state in which the edge agent k at time slot t is in, i.e., the observed environmental situation elements. $a_k(t) \in \mathcal{A}$ indicates the action taken by the edge agent k in the current state $s_k(t)$ at time slot t. $r_k(t) \in \mathcal{R}$ denotes the rewards received by the edge agent k at time slot t after taking action $a_k(t) \in \mathcal{A}$ in the current state $s_k(t)$. For each given state $s \in \mathcal{S}$, the edge agent takes policy $\pi : \mathcal{S} \mapsto \mathcal{A}$ to select an action from its action space to act on the current state s. In the following, we provide the state space, action space, and rewards that are required by the algorithm.

1. State space: The state space of each edge agent composes of three dimensions of situational elements. Specifically, it includes information about the external environment, the internal node vulnerability, and the possible attacks. These situational elements constitute the smart grids' environment information at each given time slot t, which is taken as a state here. For a cluster of edge agents at time slot t, the state space is represented as

$$s(t) = (s_1(t), s_2(t), \ldots, s_K(t)). \tag{11}$$

For a given agent k, $k \in \{1, 2, \ldots, K\}$, the state at time slot t is represented as

$$s_k(t) = \left\{ O_{kp(t)}^{exPot}(t), O_{ke(t)}^{exLog}(t), O_{kn(t)}^{in}(t), O_{ka(t)}^{At}(t), \mathcal{E}_k^{exPot}, \mathcal{E}_k^{exLog}, \mathcal{E}_k^{in}, \mathcal{E}_k^{At} \right\}. \tag{12}$$

2. Action space: Each edge agent chooses the action to execute from its state space in the current state by using policy $\pi : \mathcal{S} \mapsto \mathcal{A}$. The action here is the edge agent's security-aware behavior based on the situational elements in the current environment. We define it as the edge agent's awareness of four situational elements in three dimensions. For a cluster of edge agents, the action space in time slot t is represented as

$$a(t) = (a_1(t), a_2(t), \ldots, a_K(t)). \tag{13}$$

For each agent k, $k \in \{1, 2, \ldots, K\}$, its act of detecting anomalies at time slot t is taken as an action, denoted as

$$a_k(t) = \left\{ \mathcal{A}_{bk}^{p(t)}, \mathcal{A}_{bk}^{e(t)}, \mathcal{A}_{bk}^{n(t)}, \mathcal{A}_{bk}^{a(t)} \right\}. \tag{14}$$

3. Rewards: After each edge agent performs an action in the current state, we define $r_k(t)$ as the immediate rewards given to edge agent k in the present time slot t. Hence, $r_k(t)$ is a state and action function that will guide edge agent k to the optimal state-action policy. Considering that SSA is to obtain the minimum processing cost with the minimum detection error rate, we design the immediate rewards according to the initially formulated problem goal. Thus, we set the immediate rewards of edge agent k, $k \in \{1, 2, \ldots, K\}$ at time slot t as

$$r_k(t) = -\mathcal{C}(t) - \Gamma(t). \tag{15}$$

Note that $\Gamma(t)$ is the perceived failure punishment for each edge agent at time slot t, considering the loss to the smart grids from actions taken that fail to detect threats. The value of $\Gamma(t)$ will increase if a normal situational element is detected as a threat or if an attack occurs but not be detected.

5.2. Algorithm Design

Algorithm 1 is the pseudo-code of our given MADDPG algorithm for the SSASGEC problem. The details of the proposed MADDPG algorithm are described as follows.

Algorithm 1. Multi-agent deep deterministic policy gradient algorithm for the SSASGEC problem.

Initialize: the actor's evaluation and critic's target networks for each edge agent
1: **for** episode = 1 to M **do**
2: Initialize a random process $\mathcal{G}$ for exploration of action
3: Receive initial state $\mathbf{s}$
4: **for** t = 1 to $\mathbb{N}$ **do**
5: for each edge agent k, select action $a_k(t) = \mu_{\theta_k}(o_k) + \mathcal{G}_t$ from the action space, w.r.t. the current policy and exploration
6: Execute actions $a(t) = (a_1(t), a_2(t), \ldots, a_K(t))$, then observe rewards r and the new state $\mathbf{s}^{'}$
7: Store $(\mathbf{s}, a, r, \mathbf{s}^{'})$ in replay buffer $\mathcal{D}$
8: $\mathbf{s} \leftarrow \mathbf{s}^{'}$
9: **for** agent k = 1 to K **do**
10: Sample a random minibatch of $\mathcal{X}$ samples $(\mathbf{s}^i, a^i, r^i, \mathbf{s}^{'i})$ from $\mathcal{D}$
11: Set $y^i = r_k^i + \gamma Q_k^{\mu'}(\mathbf{s}^{'i}, a_1', \cdots, a_K')\Big|_{a_j' = \mu_j'(o_j^i)}$
12: Update the parameter of critic's evaluation network by minimizing the loss
$L(\theta_k) = \frac{1}{\mathcal{X}} \sum_i \left(y^i - Q_k^{\mu'}(\mathbf{s}^i, a_1^i, \cdots, a_K^i)\right)^2$
13: Update the parameter of actor's evaluation network by maximizing the policy gradient
$\nabla_{\theta_k} \mathcal{J} \approx \frac{1}{\mathcal{X}} \sum_i \nabla_{\theta_k} \mu_k(o_j^i) \nabla_{a_k} Q_k^{\mu}(\mathbf{s}^i, a_1^i, \cdots, a_k, \cdots, a_K^i)\Big|_{a_k = \mu_k(o_k^i)}$
14: **end for**
15: Update target network parameters for each agent k:
$\theta_k' \leftarrow \tau\theta_k + (1 - \tau\theta_k')$
16: **end for**
17: **end for**

First, the actor and critic networks are initialized for each edge agent. At the beginning of each episode, the detection noise $\mathcal{G}$ is initialized, and the initial state $\mathbf{s}$ is acquired (lines 1–3). Then the iterations are carried out, and each edge agent chooses the action to perform based on its policy and the noise (lines 4–5). After each edge agent performs an action, the total rewards r is observed and a new state $\mathbf{s}^{'}$ is generated (line 6). The current state $\mathbf{s}$, the performed action a, the rewards r, and the new state $\mathbf{s}^{'}$ are stored in

the experience replay buffer $\mathcal{D}$. In addition, the new state $\mathbf{s}^{'}$ is used as the state at the beginning of the next iteration (lines 7–8).

After executing a full episode, each edge agent k randomly selects a small sample from the experience replay buffer $\mathcal{D}$ among $\mathcal{X}$. Here, the samples are approximations of the agents other than agent k (line 10). To set the target value of Q-function into y^i (line 11). Then, the parameters θ of the critic-network are updated by minimizing the distance between y^i and $Q_k^{\mu'}(\mathbf{s}^i, a_1^i, \cdots, a_K^i)$ for the selected samples among $\mathcal{X}$. In the same way, the update of the policy parameter θ of the actor-network is derived by maximizing the gradient ascent (lines 12–13). After each edge server agent updates the actor and critic networks, the target-network parameter θ_k' is updated (line 15). Such an update method is designed to achieve learning stability by limiting the target values' update rate.

6. Performance Evaluation

In this section, we present simulation experiments to evaluate the proposed MADDPG-based algorithm's performance, which is employed to solve the SSASGEC problem. Considering the Wide-Area Situational Awareness technique, the edge agent gathers information on external environments, internal environments, and possible attacks of power terminals as situational elements. Specifically, this includes traffic on ports, alarm events in logs, vulnerable nodes, and known attacks. We have constructed an edge computing-based grid testbed that connects different categories of power terminals to each edge agent, typically smart meters, microgrid central controllers, relay protection devices, charging piles, etc. The port traffic information and log information of the power terminals are actively uploaded to the edge agent. The edge agent monitors the vulnerability information and the attack information of the power terminal nodes in real time. Then, the situational information is aggregated to the edge agent and quantified as the original training dataset of MADDPG. During the simulation, we assume a certain quantity of ports and alarm events, a topological network composed of power terminal nodes under intelligent edge agent, and a certain number of various types of known attacks. By changing the quantified values of the situational elements that the edge agent gathered to simulate various threats and attack behaviors. What's more, in an environmental state where a massive number of heterogeneous power terminals provide diverse situational elements, we trained the MADDPG based on the proposed algorithm during the training phase. The trained model is then used for performing tests in a new situational awareness environment to verify its performance.

In the following, we first demonstrate the proposed algorithm's convergence performance and compare it with the single-agent deep deterministic policy gradient (DDPG) algorithm. Then, we verify the effect of detection error rate on the awareness results.

6.1. Parameter Setting

In the assumed simulation, we set up three edge agents with 100–200 power terminals randomly distributed under each edge agent. Each power terminal randomly generates different dimensions of situational element information. Table 1 lists the detailed parameters of the edge agent. Further, the specific parameter settings of the neural network and the training parameters are shown in Table 2.

Table 1. Specification of edge agent in detail.

Specifications	Edge Agent
Processor	Atom E3950, Max. Frequency 2.0 GHz, Base Frequency 1.60 GHz
Memory	LPDDR4-2400MT/s, 4 GB
Graphics	Atom SoC integrated, Burst Frequency 650 MHz, 2 GB
Storage	eMMC, 32 GB

Table 2. The neural network and training parameters settings.

Parameter	Value
Layers of Critic network	5
Layer type of Critic	Fully connected
Neurons of hidden layers for critic networks	[1024,512,256]
Learning rate of Critic	0.0001
Layers of Actor network	4
Layer Type of Actor	Fully connected
Neurons of hidden layers for Actor networks	[512,128]
Learning rate of Actor	0.0001
Activation Function	Relu
Mini-batch	256
Buffer Size	20,000
Discount factor	0.95

6.2. Performance Evaluation

According to the parameter settings given in Section 6.1, we have compared the total rewards values obtained by the MADDPG-based and the single-agent DDPG-based algorithms as follows. The single-agent DDPG-based here means that only one edge agent gathers all the power terminals' situational elements. Then, DDPG is performed for awareness training. Similar to MADDPG, the rewards of DDPG are also set to the penalty for awareness failure coupled with the minimum processing cost. Further, we set the situational elements' detection error rate all to 0.01 and then iterate the proposed MADDPG-based and DDPG-based algorithm for 50,000 episodes, respectively. Figure 4 shows the total episode rewards obtained by the 3-agent MADDPG algorithm and the single-agent DDPG algorithm over 50,000 episodes of the same initial situational awareness environment, respectively. In each episode, the total episode rewards obtained by both algorithms are different. In the early training phase, the total episode rewards of both algorithms fluctuate drastically due to the agent's exploration of action strategies. As the training episode grows, the reward values gradually stabilize. After reaching the 5000 episode, both achieve convergence. While the total episode rewards obtained by the MADDPG-based algorithm are clustered around -100 to 0, the single-agent-based one is around -200 to 0. It illustrates that the total episode rewards of the MADDPG-based algorithm proposed in this paper are preferred over the single-agent DDPG algorithm after training.

On the other hand, for the training process, we take the mean rewards for every 1000 episodes of the training. Figure 5 shows the mean rewards received by a single edge agent with MADDPG-based and DDPG-based algorithm, respectively. Note that the MADDPG-based algorithm shows the mean rewards for one edge agent, and the DDPG-based algorithm shows the mean rewards for the edge agent executing the training mission divided by the value of the number of edge agents 3. It is observed that the rewards under the proposed MADDPG-based algorithm for a single edge agent are always greater than those of the DDPG-based algorithm. This indicates that the mean processing cost of the proposed multi-agent SSA model is lower than that of the single-agent DDPG-based. Further, the comparison of the mean rewards after convergence also verified this view.

Moreover, we verify the effect of the detection error rate of the situational elements on the algorithm's performance. The proposed MADDPG-based algorithm's rewards are the minimum processing cost, which is obtained under the condition that the detection error rate is less than expected. Under the WASA model, we evaluate the algorithm's performance by controlling the detection error rate of external environment, vulnerable internal nodes, and the possible attacks. Figure 6 shows the impact of four threat detection error rates on the proposed MADDPG-based algorithm's rewards. Note that, the rewards value here is the system's total rewards after convergence of the training model under different error rate requirements, including the sum of the rewards of all edge agents. It can be seen that with a comparatively high error rate, the edge agents frequently make

wrong action strategies to be aware of the situational elements, and the total rewards of the proposed algorithm are relatively low. However, as the error rate decreases, the rewards gradually grow. The lower the error rate is, the more rapidly the reward value increases. By the time the error rate approaches 0, the optimal rewards of the model are achieved. This is because the error rate reduction enables the edge agent to make correct actions based on the situational elements, resulting in higher total long-term rewards for situational awareness. It is also shown that awareness for port information has less impact on the rewards than related logs, vulnerable nodes, and known attacks. The detection error rate of known attacks has the most significant impact on the rewards.

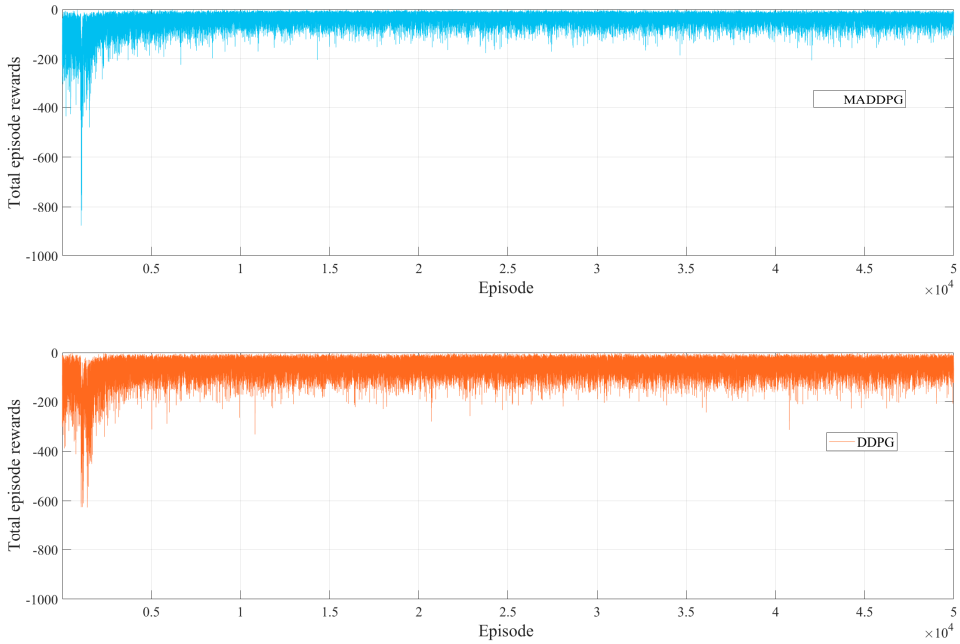

Figure 4. The total episode rewards obtained by the 3-agent MADDPG algorithm and the single-agent DDPG algorithm over 50,000 episodes.

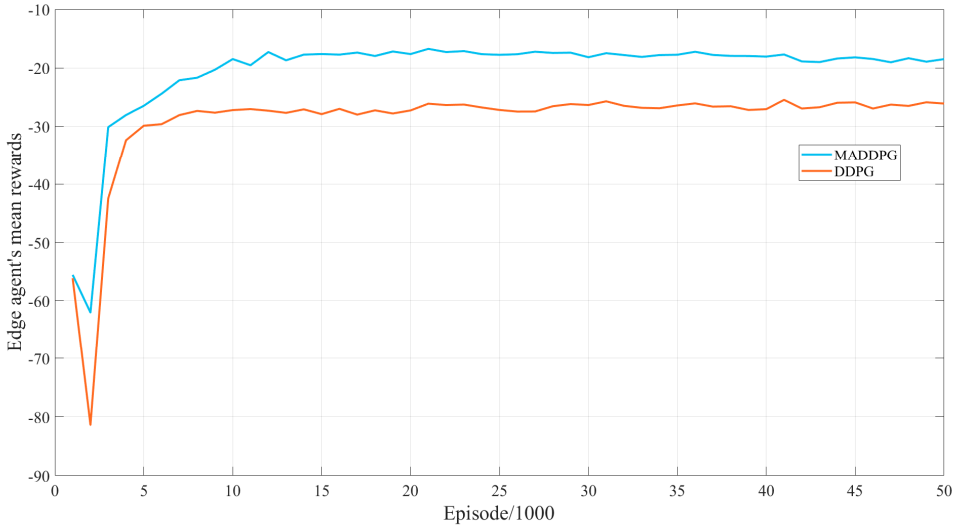

Figure 5. Comparison of MADDPG-based and DDPG-based algorithm's mean rewards for an edge agent.

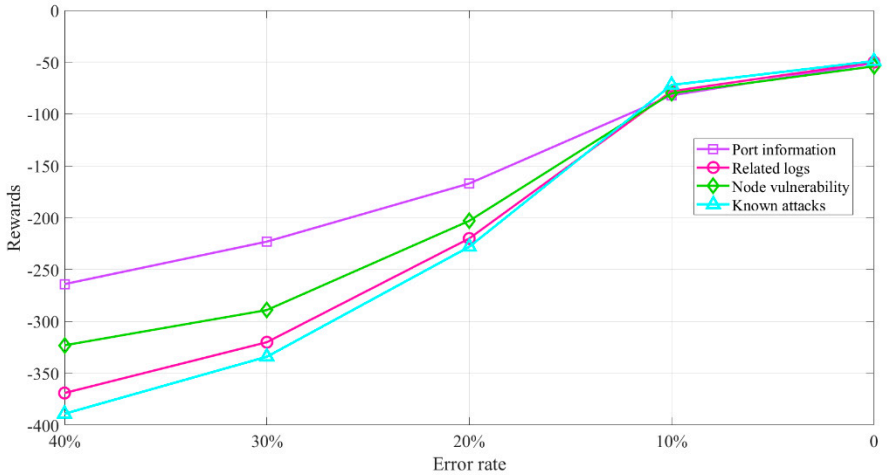

Figure 6. Impact of detection error rate on the rewards of the proposed MADDPG-based algorithm.

Next, we trained three different models independently. The models consist of the MADDPG-based model proposed in this paper and the DDPG-based model trained at the edge agent side, and the DDPG-based model trained at the cloud side, respectively. Then the trained three models have experimented simultaneously in the same situational environment. Setting the interval of each time slot to 1 s, we recorded the time for the models to make an awareness action in each time slot as the processing time. Figure 7 shows the processing time for each trained model working on the given environment to take awareness actions within 20-time slots. Among the algorithms performed at the edge, MADDPG-based has a shorter and more stable processing time than the DDPG-based algorithm. Yet, the DDPG-based algorithm performed at the cloud has the shortest processing time. Considering that situation awareness architecture based on the cloud paradigm needs to consider the network latency (around 200 ms) from edge to cloud, which is approaching 500 ms. Therefore, the processing time for threats is decreased with the effect of edge intelligence. Further, comparing the three models, the MADDPG-based situational awareness model proposed in this paper has the lowest latency for threat processing.

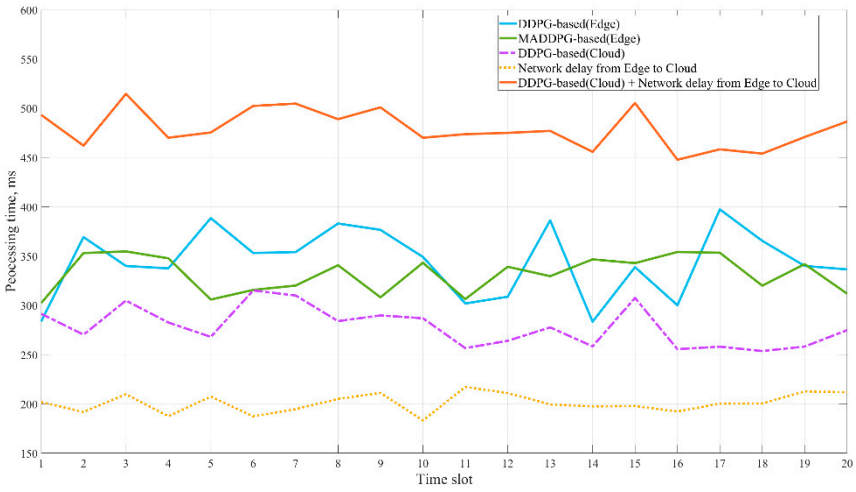

Figure 7. The processing time of different models working on the situational environment within 20-time slots.

Simulations were performed to consider the impact of a large number of heterogeneous terminals in the smart grids on the process of situational awareness. The total number of power terminals distributed under the edge agent was set to 1000 and 2000. These total terminals contain different categories. Next, we positioned the trained proposed MADDPG-

based situational awareness module to the edge and cloud, respectively. The average processing time for making each situational awareness action was calculated separately for the awareness environments of different terminals. Figure 8 shows the average processing time of the proposed model under various categories of heterogeneous terminals, including edge processing and cloud processing. To be seen, the average processing time increases with the number of terminal categories as well as the total number of terminals. The average processing time performed in the cloud is much greater than that in the edge under the same conditions. When the number of heterogeneous terminal categories approaches 100, the average processing time in the cloud is about twice that of the edge. The results show that the proposed MADDPG-based model under edge computing in this paper is effective in the problem of situational awareness with a large number of heterogeneous terminals.

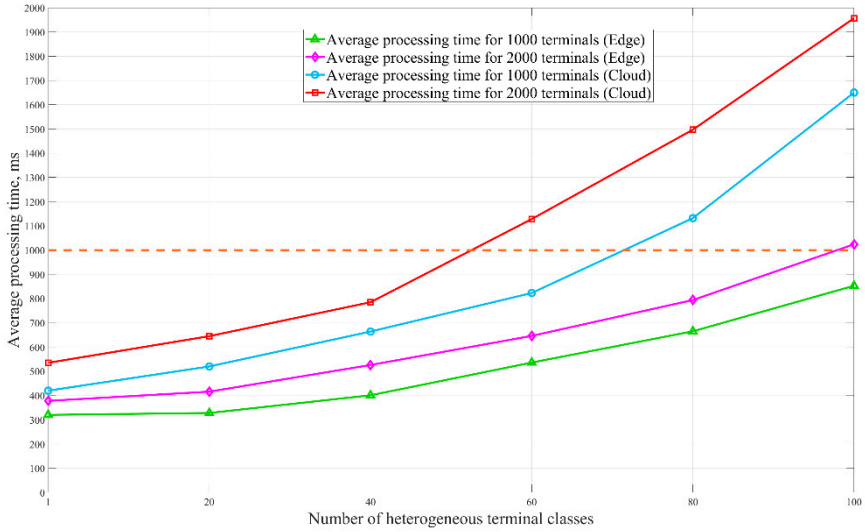

Figure 8. The processing time of situational awareness under heterogeneous terminals.

To verify the detection performance of specific categories of attacks, we implement the proposed model to detect a new type of cyber-attack on power data integrity, the false data injection (FDI) attacks [33]. FDI attacks may tamper with the measurement information acquired by power data collection terminals. If the attacker knows the topology of the smart grid, the FDI attack vector can be constructed without changing the measurement residuals. The FDI attack characteristics closely resemble the original signal, making it invisible and difficult to detect using common defense mechanisms. In this case, the traditional bad data detection (BDD) method cannot detect the FDI attack, and the same blacklist/whitelist detection mechanism for detecting known attacks also faces a failure. Thus, the power system may get wrong state estimation results, which further affects various decisions of the power system and jeopardizes the safe operation.

However, the MADDPG algorithm proposed in this paper applies to the situational space of continuously changing elements and is capable of aware small changes in situational elements. In the following, we injected false data attack vectors with different deviation amplitudes for the IEEE-14 bus system and the IEEE 118-bus system with the active power of a single node as the attack target, respectively. Tables 3 and 4 show the proposed model's detection results against the FDI attack at the edge agent side with Gaussian white noise $N(0, 0.1)$ added to the testing data. In the IEEE 14-bus system, the detection accuracy decreased as the deviation amplitude of the FDI attack vector decreased. However, when the deviation amplitude to 0.05, the detection accuracy was still more than 85%, the false acceptance rate (FAR) was less than 8%, and the false rejection rate (FRR) was below 6.8%. With the IEEE 118-bus system, the detection accuracy reached more than 92.3% when the deviation amplitude was above 0.1, the FAR was maintained

at about 7.7%, and there were no false rejection cases. However, influenced by increased data dimensionality, the detection accuracy dropped to 72% for deviation amplitude less than 0.1, and the FAR and FRR increased correspondingly. According to the above analysis, the detection accuracy of the MADDPG-based model proposed in this paper can achieve up to 92% over the small-amplitude FDI attack for small power systems. Still, up to 92% over the FDI attack for large power systems when the deviation amplitude of the false data injection attack is more than 0.1. This responds that the multiagent deep deterministic policy gradient algorithm under the edge computing paradigm works well in solving the security situational awareness problem of false data injection attacks in the smart grid.

Table 3. Results of the proposed model for FDI attack detection in the IEEE bus-14 system.

Deviation Amplitude (%)	Label	Number of Testing Data	Identified to Be Compromised	Identified to Be Normal	False Rejection Rate (%)	False Acceptance Rate (%)	Detection Accuracy (%)
0.05	Normal	500	436	64	6.4	7.8	85.7
	Compromised	500	78	421			
0.1	Normal	500	432	68	6.8	1.1	92.1
	Compromised	500	11	489			
0.3	Normal	500	498	2	0.2	0	99.8
	Compromised	500	0	500			

Table 4. Results of the proposed model for FDI attack detection in the IEEE bus-118 system.

Deviation Amplitude (%)	Label	Number of Testing Data	Identified to Be Compromised	Identified to Be Normal	False Rejection Rate (%)	False Acceptance Rate (%)	Detection Accuracy (%)
0.05	Normal	500	437	63	6.3	21.7	72.0
	Compromised	500	217	283			
0.1	Normal	500	500	0	0	7.7	92.3
	Compromised	500	77	423			
0.3	Normal	500	500	0	0	4.3	95.7
	Compromised	500	43	457			

7. Conclusions

In view of increasing security issues, it is necessary to apply situational awareness methods to provide comprehensive security protection for smart grids. However, there are still many inadequacies in the current situational awareness applications. The heterogeneous connections of massive power terminals with multiple communication protocols and multiple terminal interactions are added to the burden of the power network. In addition, the smart grids based on the power cloud master station pattern may not be timely for threat response. Therefore, the effective management of massive heterogeneous terminals, high bandwidth, and low latency situational awareness methods need to be updated urgently. Based on this, this paper has proposed an edge computing paradigm applied in smart grids' security situational awareness. By deploying intelligent edge agents with specific computing resources close to the power terminals, the problem of multiple power terminals and immediate threat response has been effectively solved.

On the other hand, in the pursuit of efficient situational awareness methods, traditional security situational awareness methods cannot effectively solve the problems such as dynamic changes in environmental and continuous situational elements' state space. Together with the high-security requirements of smart grids, we have proposed a multi-agent deep deterministic policy gradient algorithm based on deep reinforcement learning to solve the above problems. The situational elements of wide-area situational awareness are considered the state space of the algorithm, the awareness behavior as the action space, the

minimized cost of the situational awareness processing, and the awareness failure penalty as the rewards. The performance evaluation shows that the proposed MADDPG-based algorithm can effectively solve the formulated problem.

In the future, we would like to investigate how to apply the proposed algorithms to edge agents with different performance configurations to achieve a higher efficiency security situational awareness approach applicable to various grid environments. A wider range of situational elements for smart grids security will also be considered in the future.

Author Contributions: Conceptualization, W.L. and W.H.; methodology, W.L.; software, W.L.; validation, W.L., W.H., and H.W.; formal analysis, W.L. and W.H.; investigation, W.L.; resources, W.L., W.H., and H.W.; data curation, W.L.; writing—original draft preparation, W.L.; writing—review and editing, H.W. and J.W.; visualization, W.L., W.H., and H.W.; supervision, H.W. and J.W.; project administration, H.W.; funding acquisition, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National major R & D program under Grant 2018YFB0904900 and 2018YFB0904905, and Chile CONICYT FONDECYT Regular under Grant 1181809, and Chile CONICYT FONDEF under Grant ID16I10466.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Fang, X.; Misra, S.; Xue, G.; Yang, D. Smart Grid—The New and Improved Power Grid: A Survey. *IEEE Commun. Surv. Tutor.* **2012**, *14*, 944–980. [CrossRef]
2. Dileep, G. A survey on smart grid technologies and applications. *Renew. Energy* **2020**, *146*, 2589–2625. [CrossRef]
3. Atat, R.; Liu, L.; Wu, J.; Li, G.; Ye, C.; Yang, Y. Big Data Meet Cyber-Physical Systems: A Panoramic Survey. *IEEE Access* **2018**, *6*, 73603–73636. [CrossRef]
4. Karagiannopoulos, S.; Aristidou, P.; Hug, G. Data-Driven Local Control Design for Active Distribution Grids Using Off-Line Optimal Power Flow and Machine Learning Techniques. *IEEE Trans. Smart Grid* **2019**, *10*, 6461–6471. [CrossRef]
5. Wang, W.; Lu, Z. Cyber security in the Smart Grid: Survey and challenges. *Comput. Netw.* **2013**, *57*, 1344–1371. [CrossRef]
6. Komninos, N.; Philippou, E.; Pitsillides, A. Survey in Smart Grid and Smart Home Security: Issues, Challenges and Countermeasures. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1933–1954. [CrossRef]
7. Kumar, P.; Lin, Y.; Bai, G.; Paverd, A.; Dong, J.S.; Martin, A. Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2886–2927. [CrossRef]
8. Fairley, P. Cybersecurity at U.S. utilities due for an upgrade: Tech to detect intrusions into industrial control systems will be mandatory [News]. *IEEE Spectr.* **2016**, *53*, 11–13. [CrossRef]
9. Assault on California Power Station Raises Alarm on Potential for Terrorism. 2014. Available online: http://goo.gl/RiuhI1 (accessed on 21 November 2020).
10. Song, I.; Yun, S.; Kwon, S.; Kwak, N. Design of Smart Distribution Management System for Obtaining Real-Time Security Analysis and Predictive Operation in Korea. *IEEE Trans. Smart Grid* **2013**, *4*, 375–382. [CrossRef]
11. Liu, Y.; Yao, W.; Zhou, D.; Wu, L.; You, S.; Liu, H.; Zhan, L.; Zhao, J.; Lu, H.; Gao, W.; et al. Recent developments of FNET/GridEye—A situational awareness tool for smart grid. *CSEE J. Power Energy Syst.* **2016**, *2*, 19–27. [CrossRef]
12. Ebrahimi, M.S.; Daraei, M.H.; Behzadan, V.; Khajooeizadeh, A.; Behrostaghi, S.A.; Tajvidi, M. A novel utilization of cluster-tree wireless sensor networks for situation awareness in Smart Grids. In Proceedings of the IEEE PES Innovative Smart Grid Technologies, Perth, WA, USA, 13–16 November 2011; pp. 1–5.
13. Crotti, G.; Gallo, D.; Giordano, D.; Landi, C.; Luiso, M. Industrial Comparator for Smart Grid Sensor Calibration. *IEEE Sens. J.* **2017**, *17*, 7784–7793. [CrossRef]
14. Stevens-Adams, S.; Cole, K.; Haass, M.; Warrender, C.; Jeffers, R.; Burnham, L.; Forsythe, C. Situation Awareness and Automation in the Electric Grid Control Room. *Procedia Manuf.* **2015**, *3*, 5277–5284. [CrossRef]
15. Rusitschka, S.; Eger, K.; Gerdes, C. Smart Grid Data Cloud: A Model for Utilizing Cloud Computing in the Smart Grid Domain. In Proceedings of the First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD, USA, 4–6 October 2010; pp. 483–488.
16. Forcan, M.; Maksimović, M. Cloud-Fog-based approach for Smart Grid monitoring. *Simul. Model. Pract. Theory* **2020**, *101*, 101988. [CrossRef]

17. Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge Computing: Vision and Challenges. *IEEE Internet Things J.* **2016**, *3*, 637–646. [CrossRef]
18. Liao, R.-F.; Wen, H.; Wu, J.; Pan, F.; Xu, A.; Jiang, Y.; Xie, F.; Cao, M. Deep-Learning-Based Physical Layer Authentication for Industrial Wireless Sensor Networks. *Sensors* **2019**, *19*, 2440. [CrossRef]
19. Liao, R.; Wen, H.; Chen, S.; Xie, F.; Pan, F.; Tang, J.; Song, H. Multiuser Physical Layer Authentication in Internet of Things with Data Augmentation. *IEEE Internet Things J.* **2020**, *7*, 2077–2088. [CrossRef]
20. He, X.; Qiu, R.C.; Ai, Q.; Chu, L.; Xu, X.; Ling, Z. Designing for Situation Awareness of Future Power Grids: An Indicator System Based on Linear Eigenvalue Statistics of Large Random Matrices. *IEEE Access* **2016**, *4*, 3557–3568. [CrossRef]
21. Panteli, M.; Crossley, P.A.; Kirschen, D.S.; Sobajic, D.J. Assessing the Impact of Insufficient Situation Awareness on Power System Operation. *IEEE Trans. Power Syst.* **2013**, *28*, 2967–2977. [CrossRef]
22. Wu, J.; Ota, K.; Dong, M.; Li, J.; Wang, H. Big Data Analysis-Based Security Situational Awareness for Smart Grid. *IEEE Trans. Big Data* **2018**, *4*, 408–417. [CrossRef]
23. Wang, P.; Govindarasu, M. Multi-Agent Based Attack-Resilient System Integrity Protection for Smart Grid. *IEEE Trans. Smart Grid* **2020**, *11*, 3447–3456. [CrossRef]
24. Libri, A.; Bartolini, A.; Benini, L. pAElla: Edge AI-Based Real-Time Malware Detection in Data Centers. *IEEE Internet Things J.* **2020**, *7*, 9589–9599. [CrossRef]
25. Wang, T.; Luo, H.; Jia, W.; Liu, A.; Xie, M. MTES: An Intelligent Trust Evaluation Scheme in Sensor-Cloud-Enabled Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2020**, *16*, 2054–2062. [CrossRef]
26. Chen, S.; Wen, H.; Wu, J.; Lei, W.; Hou, W.; Liu, W.; Xu, A.; Jiang, Y. Internet of Things Based Smart Grids Supported by Intelligent Edge Computing. *IEEE Access* **2019**, *7*, 74089–74102. [CrossRef]
27. Alcaraz, C.; Lopez, J. Wide-Area Situational Awareness for Critical Infrastructure Protection. *Computer* **2013**, *46*, 30–37. [CrossRef]
28. Xie, Y.; Wen, H.; Wu, B.; Jiang, Y.; Meng, J. A Modified Hierarchical Attribute-Based Encryption Access Control Method for Mobile Cloud Computing. *IEEE Trans. Cloud Comput.* **2019**, *7*, 383–391. [CrossRef]
29. Hou, W.; Jiang, Y.; Lei, W.; Xu, A.; Wen, H.; Chen, S. A P2P network based edge computing smart grid model for efficient resources coordination. *Peer-To-Peer Netw. Appl.* **2020**, *13*, 1026–1037. [CrossRef]
30. Khan, W.Z.; Ahmed, E.; Hakak, S.; Yaqoob, I.; Ahmed, A. Edge computing: A survey. *Future Gener. Comput. Syst.* **2019**, *97*, 219–235. [CrossRef]
31. Ponzio, F.; Urgese, G.; Ficarra, E.; Di Cataldo, S. Dealing with Lack of Training Data for Convolutional Neural Networks: The Case of Digital Pathology. *Electronics* **2019**, *8*, 256. [CrossRef]
32. Alom, M.Z.; Taha, T.M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M.S.; Hasan, M.; Van Essen, B.C.; Awwal, A.A.S.; Asari, V.K. A State-of-the-Art Survey on Deep Learning Theory and Architectures. *Electronics* **2019**, *8*, 292. [CrossRef]
33. Liang, G.; Zhao, J.; Luo, F.; Weller, S.R.; Dong, Z.Y. A Review of False Data Injection Attacks Against Modern Power Systems. *IEEE Trans. Smart Grid* **2017**, *8*, 1630–1638. [CrossRef]