symmetry

MDPI

Article

A Dyna-Q-Based Solution for UAV Networks Against Smart Jamming Attacks

Zhiwei Li [1,2,*], Yu Lu [1,*], Yun Shi [3], Zengguang Wang [1], Wenxin Qiao [1] and Yicen Liu [1]

1 Shijiazhuang Campus, Army Engineering University of PLA, Shijiazhuang 050003, Hebei, China; zengguang_wang@126.com (Z.W.); qiaowenxin1992@foxmail.com (W.Q.); 18419764051@163.com (Y.L.)
2 Aircraft Maintenance Center, 93413 Units, Yongji 044500, Shanxi, China
3 Meteorological Investigation Center, 66199 Units, Beijing 100000, China; peter411@163.com
* Correspondence: arhqs@126.com (Z.L.); ylu@vip.sina.com (Y.L.); Tel.: +86-0311-87994710 (Y.L.)

Received: 31 March 2019; Accepted: 29 April 2019; Published: 2 May 2019

Abstract: Unmanned aerial vehicle (UAV) networks have a wide range of applications, such as in the Internet of Things (IoT), 5G communications, and so forth. However, the communications between UAVs and UAVs to ground control stations mainly use radio channels, and therefore these communications are vulnerable to cyberattacks. With the advent of software-defined radio (SDR), smart attacks that can flexibly select attack strategies according to the defender's state information are gradually attracting the attention of researchers and potential attackers of UAV networks. The smart attack can even induce the defender to take a specific defense strategy, causing even greater damage. Inspired by symmetrical thinking, a solution using a software-defined network (SDN) to combat software-defined radio was proposed. We propose a network architecture which uses dual controllers, including a UAV flight controller and SDN controller, to achieve collaborative decision-making. Built on the top of the SDN, the state information of the whole network converges quickly and is fitted to an environment model used to develop an improved Dyna-Q-based reinforcement learning algorithm. The improved algorithm integrates the power allocation and track planning of UAVs into a unified action space. The simulation data showed that the proposed communication solution can effectively avoid smart jamming attacks and has faster learning efficiency and higher convergence performance than the compared algorithms.

Keywords: UAV networks; SDN; reinforcement learning; Dyna-Q; IoT; cyberattacks

1. Introduction

Applying UAVs in the Internet of Things (IoT) can complete a variety of IoT services, including video surveillance, sensor data collection, disaster relief emergency communications, and intelligent transportation. With the rapid development of IoT applications, the application of UAVs in the IoT has gradually changed from a single service delivery (i.e., Amazon parcel delivery, power line monitoring, etc.) to UAV swarm applications (i.e., urban pollution monitoring, geological disaster prevention, and military 'bee colonies').

The UAV network will foreseeably carry more and more high-value sensitive data, but the UAV network can easily become the target of cyberattacks since the wireless channels are exposed to the air. As the UAV network faces serious security threats, it is imperative to effectively manage possible security threats.

With the advent of software-defined radio (SDR) technology, programmable, intelligent wireless attack devices will pose a greater threat to UAV networks. Smart attack behavior can autonomously perceive the state of the wireless spectrum. By analyzing the characteristics of the defender's behavior, it can carry out various types of attacks, such as eavesdropping, jamming, and spoofing. Among them,

the jamming attack has the lowest technical threshold, the least implementation cost, and the most direct attack effect. The jamming attack directly weakens or blocks the UAV network communication links, so it is urgent to address this kind of smart attack behavior.

Inspired by symmetrical thinking, a solution using a software-defined network (SDN) to combat SDR may be the best choice. Both SDN and SDR are highly flexible and controllable and can deploy intelligence algorithms. Although the SDN architecture has some successful applications in MANETs (Mobile Ad-hoc NETworks) and VANETs (Vehicular Ad-hoc NETworks), due to the mobility of the UAV network and the rapid change of the network topology, it is necessary to design and deploy a new SDN architecture that meets the mission requirements and security requirements. Similarly, there are some successful experiences in deploying intelligent algorithms to the SDN on the ground. However, in deploying intelligent algorithms to UAVs, especially on UAV networks, a series of problems still need to be solved, such as communication process optimization and convergence optimization.

The main contributions of this paper can be summarized as follows:

- We propose a module-level SDN controller design for UAV networks;
- We propose a dual-controller cooperative SDN-based UAV network wireless communication scheme;
- A Dyna-Q-based reinforcement learning algorithm for power allocation and track planning collaborative optimization against smart jamming is proposed.

The remainder of this paper is organized as follows. We review related work in Section 2. In Section 3, we design the topology architecture and functional architecture of the SDN-based UAV network, present the design of the module-level SDN controller, and establish the network model and jamming attack model. We propose an improved Dyna-Q-based smart defense communication solution in Section 4. Simulation results and discussion are presented in Sections 5 and 6.

2. Related Work

Several studies on different approaches have been conducted regarding secure UAV communication, but few studies have focused on UAV network communication. The security problem of the UAV network has remained an open issue until now. In Section 2.1, we present the latest research advances in SDN architectures that have previously received less attention, yet have great potential in addressing UAV network communication problems. In Section 2.2, we highlight the unique security risks in the UAV network. In Section 2.3, we track the security risks and development opportunities that intelligent technology brings to UAV networks.

2.1. SDN-Based UAV Network Control

SDN architecture is essential to enhance the controllability of networks against attackers, since it can calculate an optimal forwarding path according to the approximately real-time network status. This architecture has had many successful applications in ground networks, and therefore many researchers have attempted to replicate this success in the UAV networks. Zhang et al. [1] designed a SDN-based network framework for UAV backbone networks. In this framework, an SDN controller is deployed in the ground control station. They showed that the deployment of an SDN can extend the UAV's battery life due to a load balance algorithm integrated in the SDN controller. White et al. [2] proposed a SDN/NFV (Network Function Virtualization)-based lightweight modular network architecture which meets the high mobility requirement of a UAV swarm. Their main contribution is realizing the highly robust migration of many network services related to UAV networks, such as network monitoring, intrusion detection, and smooth migration of UAVs among different clusters, etc. Zhao et al. [3] developed a SDN architecture-based UAV networks with a single centralized SDN controller to solve the problem about how to replan the location of the relay UAVs, thus improving the QoS (Quality of Service) of a real-time video monitoring service. Alioua et al. [4] implemented SDN-based UAV-aided VANETs and investigated how to realize efficient data processing by offloading of computing tasks

and sharing of state information. They modeled the tradeoff between computational delay and energy expenditure as a two-person sequential game problem. Barritt et al. [5] introduced a network framework called a temporospatial software-defined Network (TS-SDN), which can be applied in UAV networks. Kirichek et al. [6] proposed a software-defined flight ubiquitous sensor network (FUSN) and a set of message interaction rules between a UAV and ground control station. They suggested that sensor modules and routing control modules should be deployed in different UAVs. Rahman et al. [7] conducted a study on how to adjust the location of SDN controller to reduce the communication overhead of the control messages. Ramaprasath et al. [8] claimed that they exploited a SDN-based scheme to control routing. The goal of their work was to maximize throughput, balance traffic, and reduce network delay. Toufga et al. [9] defined a topology-discovering service for SDN-based hybrid VANETs. Their method fully considered flow load balancing of an SDN controller and calculation resources needed. Rahman et al. [10] studied the deployment of an SDN controller. They claimed that SDN controllers should be deployed in the central area of UAV networks to reduce hopping counts of control packets and reduce network delay.

2.2. Cyber Threats Against UAV Networks

Recently, there were some attempts to adopt the idea of machine learning into UAV network defense. Kim et al. [11] evaluated the behavior of attackers targeting UAV nodes, especially automatic dependent surveillance—broadcast (ADS-B) attack and false data injection attack. They designed a set of rules to identify the normal behavior of UAVs, but they did not evaluate the performance of their methods in terms of detection accuracy and resource overhead. Strohmeier et al. [12] summarized the attacks targeting ADS-B components, such as eavesdropping, jamming, and false data injection, but they did not validate their countermeasures via simulation. Strohmeier [12] and Wesson [13] claimed that ADS-B is a component that is vulnerable to cyberattacks because it does not have built-in security mechanisms. They investigated a communication scheme based on confidentiality to protect the privacy of messages broadcasted by ADS-B components. Unfortunately, they did not offer a solution to prevent the detected attacks. Shepard et al. [14] believe that GPS (Global Positioning System) spoofing attacks are the most lethal attacks to UAV networks. They proposed an improved scheme to identify GPS spoofing attacks. Manesh et al. [15] systematically reviewed the security risks and security solutions of ADS-B, and they divided the security solutions into ten categories, namely lightweight PKI (Public Key Infrastructure), message authentication code, μ TESLA (Timed Efficient Stream Loss-tolerant Authentication), multilateration, fingerprinting, spread spectrum, distance bounding, Kalman filtering, data fusion, and traffic modelling. However, they did not consider the situation of air-to-air interference, so their review only focused on interference to ground stations. Brust et al. [16] proposed a method for escorting detected malicious UAVs by transforming the formation of the UAV swarm. However, if the malicious UAV suddenly launches a jamming attack, the large number of UAVs near this malicious UAV will suffer disastrous consequences. Zhao et al. [17] proposed two algorithms including a centralized deployment algorithm and a distributed motion control algorithm for UAV airborne networks, and the functions of these algorithms are to realize on-demand coverage when a disaster occurs; however, looking at it from another angle, the jamming UAV can use these algorithms to achieve maximum jamming efficiency.

2.3. Smart Defense Technology for Jamming Attacks

Jamming attacks should be given higher priority than other types of attacks in UAV networks, since available radio channels are the physical basis of any type of UAV communications. Using game theory and reinforcement learning techniques to tackle the smart jamming problem is an emerging academic research hotspot. Wu et al. [18] proposed a Colonel Blotto anti-jamming game-based power allocation scheme which can enhance the anti-jamming performance in cognitive radio networks. Xiao et al. [19] studied the interactions among a source node, a relay node, and a jamming mode using a Stackelberg game, and these nodes chose their transmit power in turn with the premise that they did not

interfere with primary users. Tang et al. [20] proposed a Stackelberg game-based packets transmission scheme to establish a power allocation strategy in order to improve the SINR (Signal to Interference plus Noise Ratio) of radio channels. Xiao et al. [21] investigated the subjective decision process of smart jammers in a time-varying environment based on prospect theory. El-Bardan et al. [22] introduced a stochastic differential game model to solve the power allocation problems under conditions of uncertain channel gains. Wang et al. [23] intended to provide a systematic review of the UAV networks from a cyber-physical system perspective. They summarized the security requirements of UAV networks as sensing security, storage security, communication security, actuation control security, and feedback security. They divided the UAV network into three hierarchies, i.e., the cell level, the system level, and the system of system level, for detailed investigation, and the coupling effects were discussed as well. They outlined several uses of intelligent algorithms for UAV networks, including flight control, path planning, machine vision, and pattern recognition, among others.

In conclusion, most studies related to the application of SDNs in UAV networks focused on the deployment of the SDN controller, routing control, and load balance problems. To the best of our knowledge, no existing works focus on the joint optimization between network performance and UAV track planning. Although some researchers have obtained research progress in anti-jamming by formulating a game between UAVs and jammers, they do not consider the multi-UAV scenario. Those studies on multi-UAV scenarios and smart attacks assume that UAVs are stationary, which is not consistent with reality. In this paper, we focus on the smart defense communication strategy of UAV networks by optimizing the power allocation of each UAV under the action space of radio channel selecting and trace planning.

3. System Model and Network Controller Design

In this section, our goal is to implement the construction of an SDN-based UAV network and model the network and the jamming attacks against the network. In Section 3.1, we propose an SDN-based UAV network architecture. In Section 3.2, a module-level SDN controller design is given in detail. In Section 3.3, we build a hierarchical mesh UAV network model and model UAV location as a spatial grid world. In Section 3.4, we build a model of a UAV network against multiple jammers.

3.1. Network Architecture

In this subsection, we respectively establish the topology architecture and functional architecture of the UAV network. The UAV network topology we studied is a hierarchical mesh network architecture. The functional structure of the dual controller is proposed, and the functional architecture clearly reflects the design concept that the data plane and the control plane are separated from each other.

3.1.1. Topology Architecture

There are six kinds of communication objects in the SDN-based UAV network under a smart jamming environment: the backbone UAV, mission UAV, GPS, ground station, jamming UAV, and jamming station, as illustrated in Figure 1. Among them, the first four belong to the UAV network, and the latter two belong to the jamming source, which will be described in detail in Section 3.4. The UAV network topology studied in this paper is a hierarchical mesh network structure, which is widely used in engineering practice because of its strong scalability. The backbone UAVs, which have wider communication bandwidth, stronger calculation capacity, and larger wireless coverage than the mission UAV, are connected to each other to form the core layer of the UAV network. Each backbone UAV provides a communication relay service for several mission UAVs, which form a cluster as shown by the dotted oval in Figure 1. Each of these clusters is equivalent to a small ad hoc network, and they are the edge layer of UAV network. Mission UAVs can be equipped with many kinds of payload, and they periodically send and receive packets to evaluate the SINR of each channel. All backbone UAVs and some mission UAVs are equipped with GPS modules to obtain their position, speed information, and positioning time. The ground station is connected to at least one backbone UAV.

The backbone UAVs periodically generate a network situation view report and send it to the ground station and other UAVs. The ground station can dispatch the latest mission plan to the UAV network.

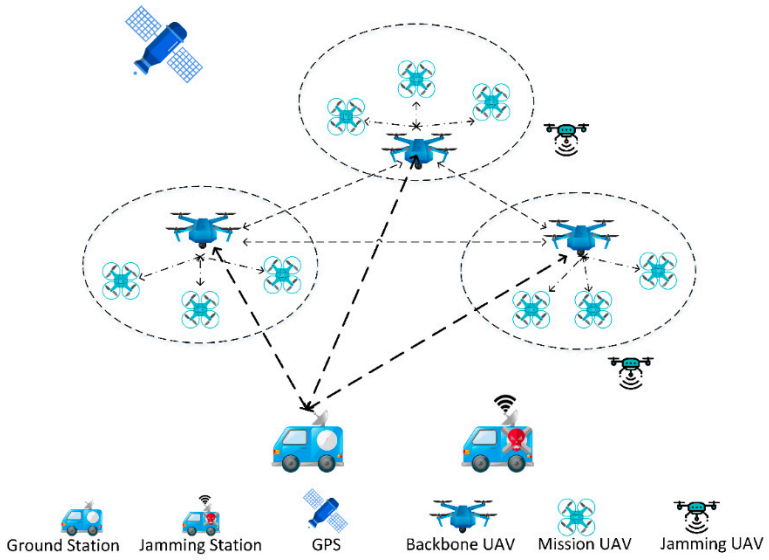


Figure 1. Topology of a software-defined network (SDN)-based unmanned aerial vehicle (UAV) network under a smart jamming environment, and the dotted lines in the figure indicate the wireless channel between the UAVs or between UAVs and Ground Station. GPS: Global Positioning System.

3.1.2. Functional Architecture

In order to achieve the goal of autonomous decision-making, unlike the conventional method, which deploys an SDN controller on the ground station, we deploy a network controller that includes one UAV flight controller and one SDN controller in the core layer of the UAV network. The functional architecture of the UAV network is shown in Figure 2. The separation of the data plane and the control plane of the SDN architecture provides a solid foundation for the autonomous control of the UAV network. The control plane consists of the network controller deployed in a UAV at the core layer, and the data plane consists of all the edge-layer UAVs and other UAVs in the core layer. The network controller chooses the optimal strategy based on the state information, including GPS data, transmission rate data, network delay data, and SINR data collected from the data plane. The UAV flight controller is responsible for UAV track planning and flight attitude control, such as heading angle adjustment, location control, and energy control. The SDN controller is responsible for transmission channel configuration and data packet forwarding control, including routing control, congestion control, and power allocation control. The relationship between the UAV flight controller and the SDN controller is a collaborative relationship. Specifically, the flight of the UAV swarm is usually controlled by the flight controller. However, the SDN controller can fine-tune the UAV track when the network is subjected to a jamming attack.

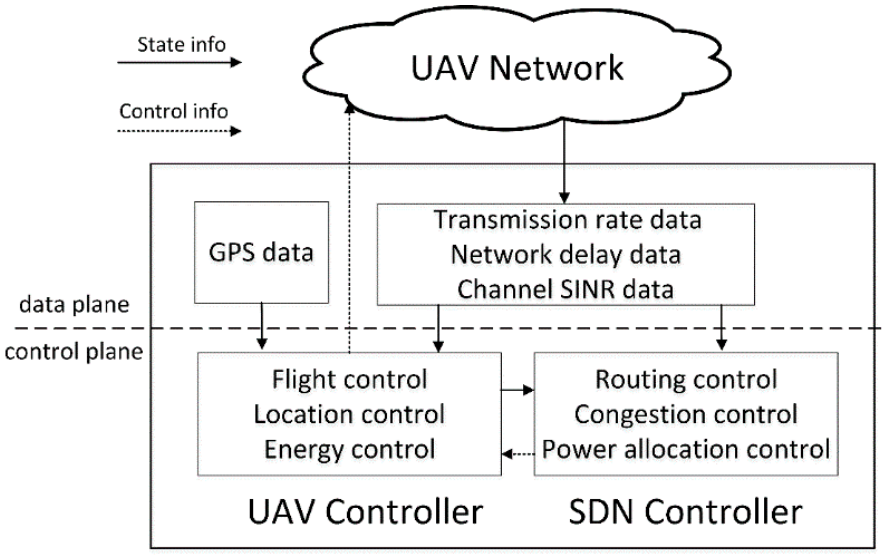


Figure 2. Functional architecture of the SDN-based UAV network. SINR: Signal to Interference plus Noise Ratio. Arrows indicate the flow of state information.

3.2. SDN Controller Design

The control of the SDN-based UAV network adopts a large and small dual-loop design. The large loop is the loop formed by each UAV and the network controller. Each UAV periodically collects attack features, including the SINR of each channel, GPS coordinates, flight velocity, and positioning time, and sends them to the network controller, thus generating a network attack situation map. The SDN controller generates the power allocation scheme and the new flow table according to the attack situation map and sends the commands and new flow table to the UAV involved for execution. The small loop is the control loop of each UAV itself. Each UAV periodically receives a global network attack situation map from the network controller. Each UAV combines the full network situation map with its newly received situation information as the basis for its next action.

Specifically, we propose a design scheme for an SDN controller, as illustrated in Figure 3. The design diagram consists of three parts. The lower area is the information collection area of the UAV network, the middle area is the SDN controller area, and the upper area is the monitoring area.

1. The information collection area generates state information. The information collection area collects various types of information to help the SDN controller make decisions. For example, the msg/packet/byte count module counts the number of messages, packets, and bytes transmitted in the network, respectively, and the *Src/Dst (Source/Destination) Address Module* records the address information of the packets forwarded by each node. The *Flow Table Module* stores a flow table currently executed by each substrate node, the SINR value of each wireless channel is collected by the channel *SINR Module* periodically, and the *GPS Coordinate Module* periodically reports the position and velocity information to the controller or the ground station.
2. The SDN controller area maps state information. The SDN controller acts like a function that maps the state information provided by the data plane to control instructions such as the *Optimal Flow Table, Power Allocation Policy*, and *UAV Location Policy*, as shown by the three black horizontal arrows in Figure 2. At the same time, the SDN controller also periodically transmits the summarized important information to the ground station. The flow of state information is as indicated by the arrows in Figure 2.
3. The monitoring area generates a network situation view. The function of this area is to convert the received status information into a network situation report suitable for human reading to assist the ground operator in mastering the latest network situation.

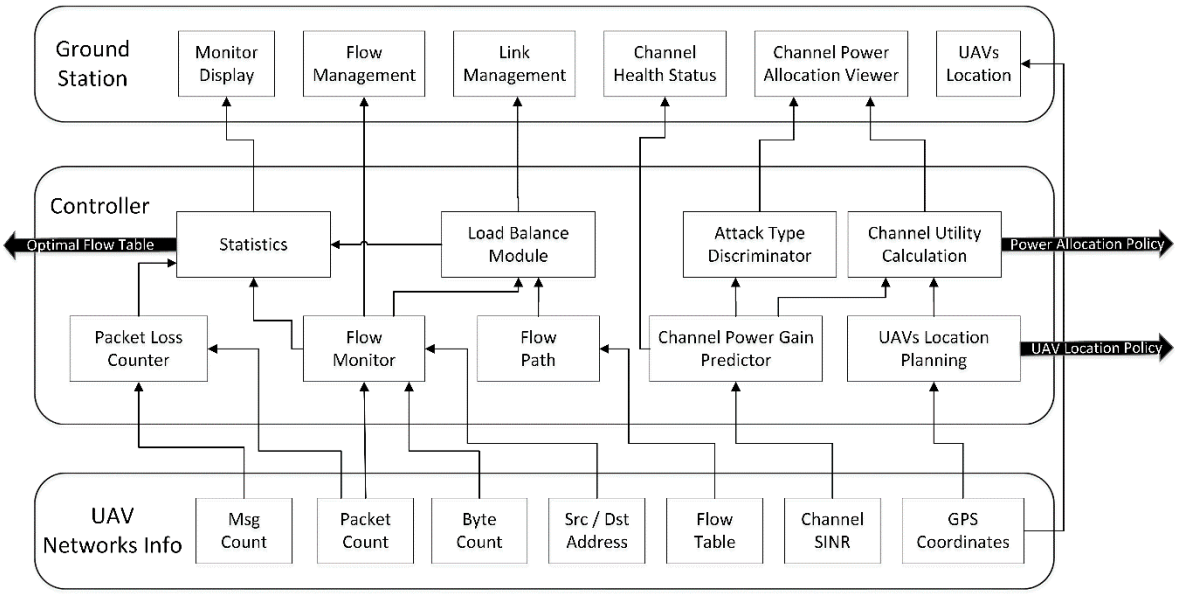


Figure 3. Design of functional modules in the SDN controller and flow of state information. Msg: Message, Src: Source, Dst: Destination.

3.3. Network Model

In the hierarchical mesh network chosen for its flexible scalability, the communication process between the core layer and the edge layer is similar. For simplicity, we only model the core layer of the network, and the modeling method of the edge layer is similar. The notations used in our model are summarized in Table 1.

Table 1. List of main notations used in this work.

Notations	Description
N_U	Number of UAVs
N_J	Number of jamming UAVs
h_u	Flying height of the u-th UAV
N	Number of radio channels
ϖ	Number of frequency patterns
C_ψ	The ψ-th frequency pattern
$f^{(k)}$	The chosen channel at time slot k
$P_u^{(k)}$	Transmit power of UAV u at time slot k
$P_{T/J}$	Total power constraints of the UAV/jammer
$d_{u_iu_j}^{(k)}$	The distance between UAV u_i and u_j at time slot k
$L_{st}^{u_i(k)}$	UAV u_i's destination at time slot k, $s, t \in \{-1, 0, 1\}$
ξ	Number of SINR quantization levels
$\mathbf{h}_{u_iu_j}^{(k)}$	Channel power gains between UAV u_i and u_j at time slot k
C_h	Cost of frequency hopping
C_p	Cost of data transmitting
C_m	Cost of UAV path replanning

There are N_U UAVs in the network, and each UAV flies at a certain height h_u to avoid collisions. The flying height can be adjusted when receiving commands from the network controller or its own flight controller, but the height should be maintained after adjustment. UAV nodes transmit messages over N radio channels. All the UAV nodes follow the same frequency pattern sets denoted by $C = \left[C_\psi\right]_{1\le\psi\le\varpi}$, where ϖ is the number of frequency patterns and the ψ-th pattern C_ψ consists of κ

time slots pattern modes, where the i-th channel is denoted by $C_\psi = \left[c_\psi^{(i)}\right]_{1\le i\le \kappa}$, and the chosen channel at time slot k can be denoted by $f^{(k)} = c_\psi^{k \bmod \kappa+1}$.

The transmit power of the u-th UAV at time slot k is denoted by $P_u^{(k)}$, and the total transmit power P_r is quantized into $L+1$ levels. The position of the u-th UAV at time slot k can be denoted by $L_u^{(k)}$, and $\left(x_u^{(k)}, y_u^{(k)}, z_u^{(k)}\right)$ are the converted Cartesian coordinates. The distance between UAV u_i and UAV u_j at time slot k is $d_{u_iu_j}^{(k)}$. The UAV swarm needs to maintain a relatively stable topology during flight. For simplicity, we assume that only one UAV is allowed to relocate at each time slot and that the other UAVs maintain uniform motion. Within each time slot, the position of the UAV can be relocated to one of the surrounding eight spatial grids, which in the clockwise direction are the north, the northeast, the east, the southeast, the south, the southwest, the western, and the northwest, as illustrated in Figure 4. Since the height difference of the UAV network is much smaller than the UAV's flight range, the spatial grid approximates the planar grid. The side length of the space grid is determined by the maneuverability of the UAV. Specifically, the side length of UAV u_i at time slot k denoted by $d_{u_i}^{(k)}$ is equal to the minimum displacement of the UAV that flies in eight directions, according to the Dubins path. The entire UAV swarm can be regarded as one large virtual UAV, and the front direction of the eight spatial grids coincides with the flight direction of the virtual drone at time slot k. UAV u_i's eight relocation spatial grids at time slot k are denoted by $\mathbf{L}^{u_i(k)} = L_{st}^{u_i(k)}$, where $s, t \in \{-1, 0, 1\}$ represents the coordinates of the spatial grids. Specifically, s represents the left and right direction, where -1 means leftward, 0 means no motion, and 1 means rightward; and t represents the front–rear directions, where -1 means forward, 0 means no motion, and 1 means backward. For example, $L_{-1,-1}^{u_i(k)}$ represents the spatial grid of UAV u_i in the front left at time slot k. When receiving a message, the UAV evaluates the signal-to-interference-plus-noise ratio (SINR) of the channel based on the bit error rate (BER) of the message. For simplicity, the value of SINR is quantized to ξ levels. Each UAV broadcasts its quantized value of the SINR and the index of transmit frequency chosen according to C_ψ at time slot k to its neighbor nodes. Let the vector $\mathbf{h}_u^{(k)} = \left[h_{u,i}^{(k)}\right]_{1\le i\le N}$ denote the channel power gains of UAV u's N channels, C_h the cost of frequency hopping, C_p the cost of data transmission, and C_m the cost of track replanning.

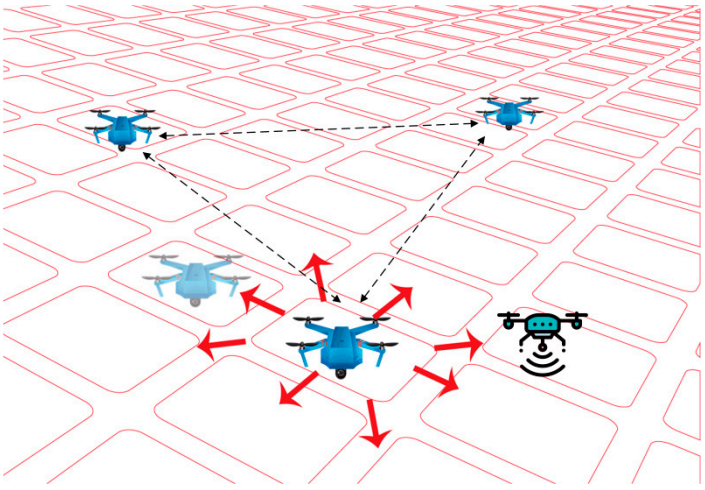


Figure 4. Schematic diagram of the relocation of a UAV. The black dotted line in the figure represents the wireless channel between the UAVs, and the red grid represents the grid world of UAVS, and the red arrows represent the directions in which the UAV may move after the jammer adjusts the jamming strategy.

3.4. Jamming Attack Model

There are two types of interference sources: one is a low-mobility jamming station and the other is a high-mobility jamming UAV, as illustrated in Figure 1. When the velocity of the jammer is not much different from the velocity of UAVs, continuous jamming can occur. When the UAV swarm is flying at high speed, the impact of the jamming station is negligible. It can be observed from the analysis mentioned above that the relative velocity of the jammers which have a great influence on the UAV network will be low. Therefore, we can assume that the state information of the whole network can converge in the SDN controller in one time slot.

Let $\mathbf{H}_{u_iu_j}^{T(k)} = \left[h_{u_iu_j,i}^{T(k)}\right]_{1\le i\le N}$ denote the channel power gains between UAV u_i and UAV u_j at time slot k, and let $\mathbf{H}_{u_iJ_j}^{J(k)} = \left[h_{u_iJ_j,i}^{J(k)}\right]_{1\le i\le N}$ denote the channel power gains between UAV u_i and jammer J_j. Let d_0 denote a reference distance and ρ the path loss exponent; the value of path loss PL at distance d can thus be modeled by

$$PL(\mathrm{dB}) = v(\mathrm{dB}) + 10\rho \lg\left(\frac{d}{d_0}\right), d > d_0 \tag{1}$$

where v is a constant indicating the antenna gain and the path loss exponent ρ characterizes the radio environment, where $\rho = 2$ describes a free-space propagation and $\rho = 4$ describes a two-ray model. Assuming $r_i \sim N(0,1)$ follow the standard normal distribution, we have $h_i^{g(k)} = r_i^{(k)}/PL_g$, where $g = T/J$.

4. Improved Dyna-Q-Based Smart Defense Communication Solution

Artificial intelligence can make the UAV network smarter, thus resisting SDR-based smart attacks. The Dyna-Q-based algorithm is a model-based reinforcement learning technique. If we do not consider the limitations of calculation resources, a deep reinforcement learning technique (e.g., DQN (Deep Q Network), fast-DQN) can be adopted to tackle large state space and action space problems, as convolutional neural networks (CNNs) have powerful feature extraction capacity. Unfortunately, it is very uneconomical to deploy the large amount of calculation resources that CNNs need on UAVs. A practical method is to use a grading method that divides continuous values into several quantitative levels to compress the state space and the action space, and then attempt to speed up the convergence of the algorithm. The SDN-based UAV network has strong environmental awareness and control flexibility, and the error between radio channel models and the real radio environment is easily eliminated by multiple iterations. Therefore, we chose the Dyna-Q technique to calculate a better smart defense strategy of the UAV network.

The interaction between UAVs and jammers can be formulated as a multistage repeated game. To derive the optimal communication, Dyna-Q-based reinforcement learning techniques can be used to control both the UAV power allocation policy and UAV relocation policy. In the proposed algorithm, the main function of the SDN controller is to aggregate the SINR values of the UAV nodes of the whole network, fit a quadric equation with these SINR values and their GPS coordinates, and use the surface equation to predict SINR values of the eight spatial grids that surround the UAV. Compared to the conventional Dyna-Q algorithm, the proposed algorithm uses global information to predict the SINR values in eight directions around each UAV and uses the predicted values to optimize the Q function.

Upon receiving packets from other UAVs, the receiving UAV node extracts the SINR estimated by the sending UAV node and formulates the state as $\mathbf{s}^{(k)} = \left[\mathrm{SINR}^{(k)}, L_u^{(k)}\right] \in S$, where $L_u^{(k)} = \left(x_u^{(k)}, y_u^{(k)}\right)$ represents the position coordinates of the sending UAV required by GPS and S is the state set. The receiving UAV node adopts a Dyna-Q-based reinforcement learning algorithm to choose the transmit power $P_s^{(k)}$ and determines whether to launch the track relocation action with the communication action denoted by $a^{(k)} = \left[P_s^{(k)}, L_{st}^{u(k)}\right] \in \mathcal{A}$, where $L_{st}^{u(k)}$ is UAV u's flight direction approximated by eight spatial grids and $\mathcal{A}$ is the action space.

In the process of communication, the sending UAV node evaluates the SINR from the feedback packets and calculates the utility based on the SINR and the communication cost, including the cost of frequency hopping C_h, the cost of data transmitting C_p, and the cost of UAV path replanning C_m; the utility at time slot k $r^{(k)}\left(\mathbf{s}^{(k)}, \mathbf{a}^{(k)}\right)$ is given by

$$r^{(k)}\left(\mathbf{s}^{(k)}, \mathbf{a}^{(k)}\right) = \hat{\mathrm{SINR}}^{(k)} - C_p P_s^{(k)} - C_m \mathcal{F}\left(L^{(k)} - L_{00}^{(k)}\right) - C_h \mathcal{F}\left(f^{(k)} - f^{(k-1)}\right) \tag{2}$$

where $\mathcal{F}(.)$ is an bool function that equals 0 if the argument of $\mathcal{F}(.)$ equals 0; otherwise, $\mathcal{F}(.)$ equals 1.

The value function denoted by $V(\mathbf{s})$ stands for the maximum value of the Q function. The UAVs update their Q function and value function at time slot k as follows:

$$Q\left(\mathbf{s}^{(k)}, \mathbf{a}^{(k)}\right) \leftarrow (1-\alpha) Q\left(\mathbf{s}^{(k)}, \mathbf{a}^{(k)}\right) + \alpha\left(r\left(\mathbf{s}^{(k)}, \mathbf{a}^{(k)}\right) + \gamma V\left(\mathbf{s}^{(k+1)}\right)\right) \tag{3}$$

$$V\left(\mathbf{s}^{(k)}\right) = \max_{\mathbf{a}^{(k)} \in A} Q\left(\mathbf{s}^{(k)}, \mathbf{a}^{(k)}\right) \tag{4}$$

where the learning factor α adjusts the UAV's learning speed and the discount factor γ adjusts the importance of future rewards. In order to balance the exploit and exploration, the ε – greedy strategy is often used. $\varepsilon \in (0, 1]$ is a small positive value, which represents the likelihood of choosing the explore strategy, and $1 - \varepsilon$ means the likelihood of choosing the exploit strategy.

When all the UAVs send their states $\mathbf{s}_{u_i}^{(k)} = \left[x_{u_i}^{(k)}, y_{u_i}^{(k)}, \mathrm{SINR}_{u_i}^{(k)}\right]$ at time slot k to the SDN controller, the SDN controller can fit a quadratic equation (Equation (5)); that is:

$$q_0 + q_1 x + q_2 y + q_3 x^2 + q_4 xy + q_5 y^2 = v \tag{5}$$

where (x, y) means the position coordinates of UAV, v is the SINR at (x, y), and $q_0 - q_5$ are the parameters of the quadratic equation. Let $N^{(k)}$ be the number of states received by the SDN controller at time slot k, and let δ be the functional error of least squares fitting as follows:

$$\delta = \sum_{i=1}^{N} \sum_{j=1}^{N} \left[v_{i,j} - \left(q_0 + q_1 x_{i,j} + q_2 y_{i,j} + q_3 x_{i,j}^2 + q_4 x_{i,j} y_{i,j} + q_5 y_{i,j}^2\right)\right] \tag{6}$$

Let the partial derivatives of δ to $q_0 - q_5$ be 0, and the value of $q_0 - q_5$ can be obtained by solving these equations, which is given by

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix}_{6\times 1} = \begin{bmatrix} \sum_{i=1}^{N}\sum_{j=1}^{N} 1 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j}^2 \\ \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^3 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j}^2 \\ \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j}^3 \\ \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^3 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^4 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^3 y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j}^2 \\ \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^3 y_{i,j} & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j}^3 \\ \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j}^3 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j}^2 y_{i,j}^2 & \sum_{i=1}^{N}\sum_{j=1}^{N} x_{i,j} y_{i,j}^3 & \sum_{i=1}^{N}\sum_{j=1}^{N} y_{i,j}^4 \end{bmatrix}_{6\times 6}^{-1} \begin{bmatrix} \sum_{i=1}^{N}\sum_{j=1}^{N} v_{i,j} \\ \sum_{i=1}^{N}\sum_{j=1}^{N} v_{i,j} x_{i,j} \\ \sum_{i=1}^{N}\sum_{j=1}^{N} v_{i,j} y_{i,j} \\ \sum_{i=1}^{N}\sum_{j=1}^{N} v_{i,j} x_{i,j}^2 \\ \sum_{i=1}^{N}\sum_{j=1}^{N} v_{i,j} x_{i,j} y_{i,j} \\ \sum_{i=1}^{N}\sum_{j=1}^{N} v_{i,j} y_{i,j}^2 \end{bmatrix}_{6\times 1} \tag{7}$$

Using Equations (5) and (7), we can estimate the SINR of the spatial grids surrounding each UAV. In each episode of the Dyna-Q-based learning algorithm, agents in UAVs will learn n steps from model $(\mathbf{s}, a)$ additionally, and eight experiences from the SDN global model created by the fitted quadratic surface via Equations (5) and (7) will speed up the convergence. The details of the algorithm are shown in Table 2.

Table 2. The algorithm of Dyna-Q-based UAV networks' smart defense communication.

Algorithm: Dyna-Q-based UAV Networks' Smart Defense Communication	
1	Initialize γ, α, ε, $\mathrm{SINR}^{(0)}$, T and model $(\mathbf{s}, a)$ for all $\mathbf{s} \in S$ and $\mathbf{a} \in \mathbf{A}(\mathbf{s})$
2	**for** $t = 1, \cdots, T$ **do**
3	$L_u^{(0)} \leftarrow$ get the coordinates (x_u, y_u) of UAV u by GPS
4	$\mathbf{s}^{(1)} = \left[\mathrm{SINR}^{(0)}, L_u^{(0)}\right]$
5	**if** $t = 1$, **then**
6	$\mathbf{s} \leftarrow \mathbf{s}^{(1)}$,
7	**or else**
8	$\mathbf{s} \leftarrow$ current state
9	**end**
10	$A \leftarrow \varepsilon$−greedy$(\mathbf{s}, Q)$
11	Execute action A, observe reward R, and state $\mathbf{s}'$
12	**if** $L_{st} \neq L_{00}$, **then**
13	Report SINR and reward R to SDN controller
14	Update local cyber attack overall map
15	**end**
16	$Q(\mathbf{s}, A) \leftarrow Q(\mathbf{s}, A) + \alpha\left[r(\mathbf{s}, \mathbf{a}) + \gamma \max_{\mathbf{a}} Q\left(\mathbf{s}', \mathbf{a}\right) - Q(\mathbf{s}, A)\right]$
17	Model $(\mathbf{s}, A) \leftarrow r, \mathbf{s}'$
18	**for** $k = 1, \cdots, n$ **do**
19	$\mathbf{s} \leftarrow$ random previously observed state
20	$A \leftarrow$ random action previously taken in S
21	$R, \mathbf{s}' \leftarrow$ Model $(\mathbf{s}, A)$
22	$Q(\mathbf{s}, A) \leftarrow Q(\mathbf{s}, A) + \alpha\left[r(\mathbf{s}, \mathbf{a}) + \gamma \max_{\mathbf{a}} Q\left(\mathbf{s}', \mathbf{a}\right) - Q(\mathbf{s}, A)\right]$
23	**end**
24	Receive overall cyberattack map from SDN controller
25	Calculate the parameters of the fitted SINR quadratic surface as SINR_{SDN} via Equations (5) and (7)
26	**for** $s, t = -1, 0, 1$, **do**
27	$\mathbf{s} \leftarrow [\mathrm{SINR}_{SDN}, L_{st}]$
28	$A \leftarrow \left[L_{st}, \underset{\mathbf{a}}{\mathrm{argmax}} V(\mathbf{s})\right]$ via Equation (4)
29	$R, \mathbf{s}' \leftarrow r(\mathbf{s}, A)$
30	$Q(\mathbf{s}, A) \leftarrow Q(\mathbf{s}, A) + \alpha\left[r(\mathbf{s}, \mathbf{a}) + \gamma \max_{\mathbf{a}} Q\left(\mathbf{s}', \mathbf{a}\right) - Q(\mathbf{s}, A)\right]$
31	**end**
32	**end**

5. Simulation Results

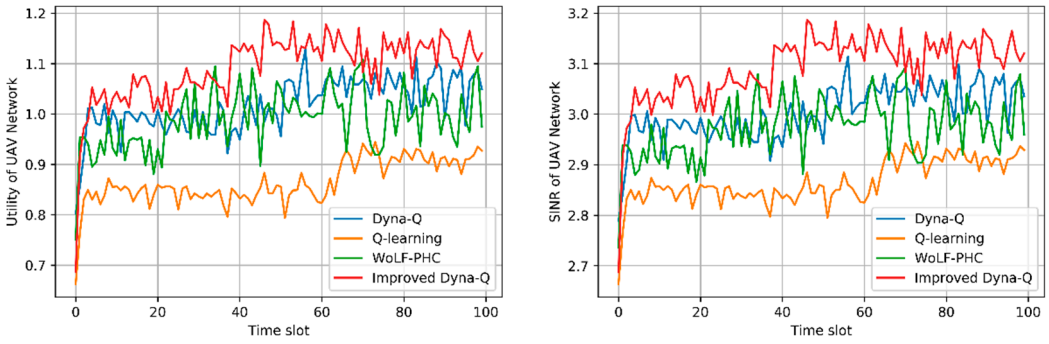
Simulations are carried out to appraise the performance of the proposed power allocation policy and trace relocation policy against a smart jammer with a Dyna-Q-based reinforcement learning algorithm. Simulation parameters similar to those used previously [21,24] are chosen, with $\alpha = 0.95$, $\gamma = 0.7$, $d_0 = 10m$, $\varepsilon = 0.9$, $C_m = 0.8$, $C_p = 0.2$, $C_h = 0.4$, $\rho_{free-space} = 2$, $\rho_{two-way} = 4$, $T = 300$, $N_U = 3$, $N_J = 2$, $N = 3$, and $\varpi = 5$. The simulated flight area is 1000 km × 800 km. The horizontal and vertical position coordinates of the moving objects are the remainders devided by 1000 km and 800 km, respectively. The initial position coordinates of the three UAVs are (260,610), (790,110), and (520,270), respectively. The initial position coordinates of the two jamming UAVs are (320,360) and (450,100). The mobility model of the jammer is a random waypoint model. The length of all spatial grids is 1 km. The speed of all the UAVs is 50 km/h. The maximum number of relocating UAVs in one time slot is 1. When calculating the channel gains via Equation (1), the distance d is calculated from the position coordinates after the remainder operation.

We use four algorithms to form the smart defense algorithm of the UAV network. The four algorithms are the WoLF-PHC (Win or Learn Fast-Policy Hill-Climbing) algorithm [21], Q-learning algorithm [25], Dyna-Q algorithm [26], and our improved Dyna-Q algorithm. We performed 100 time

slot simulations for each algorithm on a computer with 3.6 GHz Intel Core i7-4790 and 8 GB of RAM, and each time slot contains 30 episodes. Since the state space and the action space are all quantified into discretized levels, all the four algorithms end in 10 s.

We conducted three simulation experiments. The first experiment verified the performance of the four algorithms under the fixed jamming attack strategy. The second experiment verified the performance of the four algorithms under the smart jamming strategy, which means that jammers can use SDR to change their jamming strategy. The third experiment verified the influence of several key parameters on a utility metric.

In the first experiment, we compared the performance of the above four reinforcement learning algorithms under a certain jamming strategy randomly selected by jammers in 100 time slots. In this performance analysis, two most important metrics are selected, which are the utility and the SINR of the UAV network. The utility metric can be calculated by Equation (2), and the SINR metric is just the first item on the right side of Equation (2). As shown in Figure 5, the utility and SINR of the UAV network increases over time and gradually converge with different amplitudes. The improved Dyna-Q based strategy has the highest utility, followed by the Dyna-Q, WoLF-PHC, and Q-learning-based strategies. After 100 time slots, the utility of the improved Dyna-Q-based strategy is 1.1252, which is 6.9%, 13.8%, and 18.6% higher than that of the Dyna-Q, WoLF-PHC, and Q-learning-based strategies, respectively. The performance of the WoLF-PHC algorithm has large fluctuations. After 40 time slots, the range of utility of the WoLF-PHC-based strategy varies from 0.88 to 1.11, which is wider than that of the other three strategies. Besides, the SINR values calculated by the four algorithms have a similar trend compared to their utilities. The utility values are calculated by subtracting each cost from the SINR values. However, although three costs (path planning, data transmission, and frequency hopping) vary randomly, the sum of the three will offset a large part of the change, resulting in a stable trend of the sum of the three, which causes the trend of SINR to be very similar to that of utility. By the end of the simulation, the SINR with improved Dyna-Q is 3.1141, which is 2.8% higher than that of Dyna-Q, 5.4% higher than that of WoLF-PHC, and 6.6% higher than that of the Q-learning strategy.

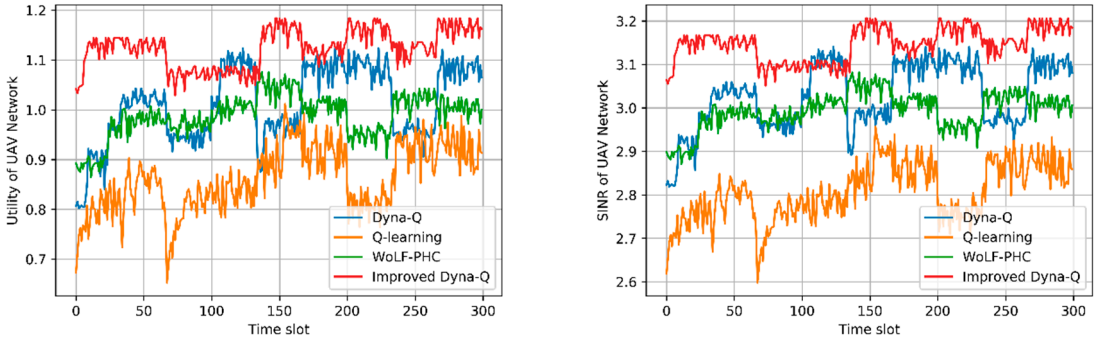


(**a**) Utility (**b**) SINR

Figure 5. Performance of the SDN-based learning algorithms using power allocation and path re-planning strategy against smart attacks with $P_T = P_J = 0.4$, $C_m = 0.8$, $C_p = 0.2$, and $C_h = 0.4$ where $P_{T/J}$ means total power constraints of the UAV/jammer, C_m means the cost of UAV path replanning, C_p means the cost of data transmitting, and C_h means the cost of frequency hopping. (**a**) shows the utility of the UAV network and (**b**) shows the SINR of the UAV network.

The second experiment was used to test the performance of the four algorithms when the jammers used a SDR-based smart jamming strategy. Specifically, the jammers can continuously adjust their attack strategies using the Q-learning algorithm, including reallocating the jamming power and replanning the location of the jammers, but can only adjust the position of one jammer to its surrounding eight grids at one time slot, with $\varepsilon = 0.3$, $\mu = 0.7$, $P_T = P_J = 0.4$, $C_m = 0.8$, $C_p = 0.2$, and $C_h = 0.4$. The UAV network selects and performs actions from the action space according to the four algorithms, respectively, and finally calculates the utility and SINR. Similarly, at most one UAV can be allowed

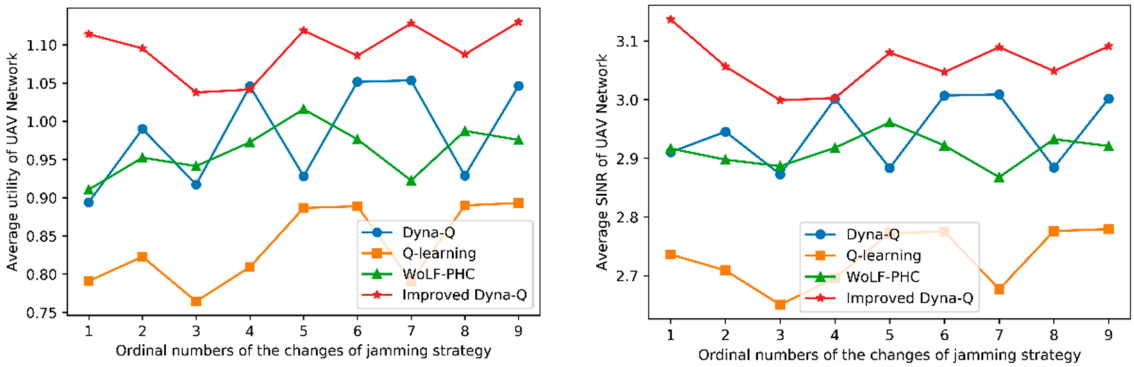
to move to one of the surrounding eight grids at one time slot. The jammers change their jamming strategy nine times at equal intervals in 300 time slots. The experimental results are shown in Figure 5. It can be seen from Figure 6 that the four algorithms bring about different degrees of performance hopping for each change of the attack strategy. The improved Dyna-Q-based strategy has the highest utility, followed by the Dyna-Q, WoLF-PHC, and Q-learning-based strategies. For instance, at the 300th time slot, the utility of improved Dyna-Q-based strategy is 1.1626, which is 8.7%, 14.9%, and 22.5% higher than that of the Dyna-Q, WoLF-PHC, and Q-learning-based strategies, respectively. The trend of SINR of the UAV network is basically the same as that of utility when facing smart jamming. At the 300th time slot, the SINR with improved Dyna-Q is 3.1928, which is 3.4% higher than that of Dyna, 5.7% higher than that of WoLF-PHC, and 10.4% higher than that of the Q-learning strategy.



(a) Utility **(b)** SINR

Figure 6. Performance of the SDN-based learning algorithms using power allocation and path re-planning strategy against smart attacks with $P_T = P_J = 0.4$, $C_m = 0.8$, $C_p = 0.2$, and $C_h = 0.4$ under the condition that jammers change their jamming strategy, including changing jamming channels and changing the positions of jammers, nine times at equal intervals in 300 time slots. (**a**) shows the utility of the UAV network and (**b**) shows the SINR of the UAV network.

To better observe the impact of changes in the jamming strategy, we averaged the utility and SINR within the same jamming strategy. The results are shown in Figure 7, Table 3, and Table 4. The average performance of the UAV network with the improved Dyna-Q strategy is optimal, the average performance of Dyna-Q and WoLF-PHC are lower and similar, and the average performance of the Q-learning algorithm is the worst. For instance, the mean value of nine average utilities in Table 3 with improved Dyna-Q is 1.0968, 0.9861 with Dyna, 0.9618 with WoLF-PHC, and 0.8380 with Q-learning. The performance advantage of the improved Dyna-Q algorithm is obvious, which is closely related to the situation awareness capabilities of the SDN architecture. The reason for the low performance of the Q-learning algorithm is that the fast-paced attack and defense strategy changes make the algorithm unable to converge in time. It is well known that the Q-learning algorithm can only update the Q value of one state per episode if the delay update mechanism is not used. This slow learning will inevitably lead to slow convergence. For example, in the 60–80th time slots of Figure 6a, the utility of the Q-learning algorithm shows a continuous but slow growth. The phenomenon of nonconvergence after 20 time slots is not uncommon in the other three algorithms. The advantage of the Dyna-Q algorithm over Q-learning is that it can be learned from models built with historical experience to speed up convergence. The advantage of the improved Dyna-Q algorithm over the Dyna-Q algorithm is that it can be learned from the SINR model fitted by Equation (7) to speed up convergence. The advantage of WoLF-PHC is that it can dynamically adjust the learning rate parameters according to the learning effect to speed up the convergence. Due to the smart jamming attack, Dyna-Q has almost no advantage over WoLF-PHC. The reason is that the changed attack strategy invalidates many experiences in the Dyna model. Learning the wrong experience in the Dyna model leads to the learning effect not rising, but falling.



(a) Average utility **(b)** Average SINR

Figure 7. Average performance of the SDN-based learning algorithms using power allocation and path replanning strategy against smart attacks with $P_T = P_J = 0.4$, $C_m = 0.8$, $C_p = 0.2$, and $C_h = 0.4$ under the condition that jammers change their jamming strategy, including changing jamming channels and changing the positions of jammers, nine times at equal intervals over 300 time slots. (**a**) shows the average utility of the UAV network and (**b**) shows the average SINR of the UAV network.

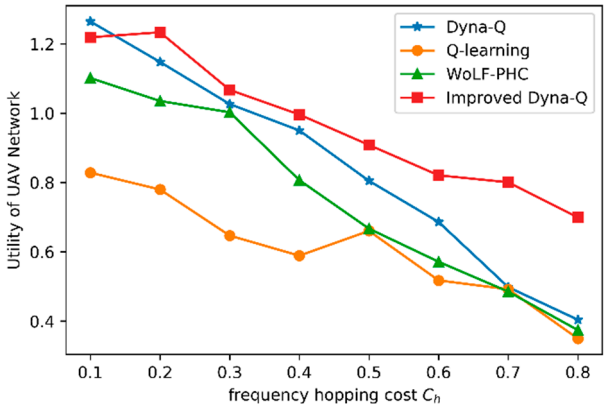
Table 3. Average utility values of the UAV network over nine rounds of smart jamming attack.

	Dyna-Q	Q-Learning	WoLF-PHC	Improved Dyna-Q
1	0.89552	0.79123	0.91045	1.11774
2	0.99201	0.82354	0.95167	1.09899
3	0.91891	0.76560	0.94276	1.04108
4	1.05217	0.81017	0.97284	1.04489
5	0.92983	0.88816	1.01630	1.12242
6	1.05396	0.88928	0.97618	1.08949
7	1.05576	0.78900	0.92270	1.13215
8	0.93050	0.88928	0.98733	1.09115
9	1.04638	0.89596	0.97618	1.13371

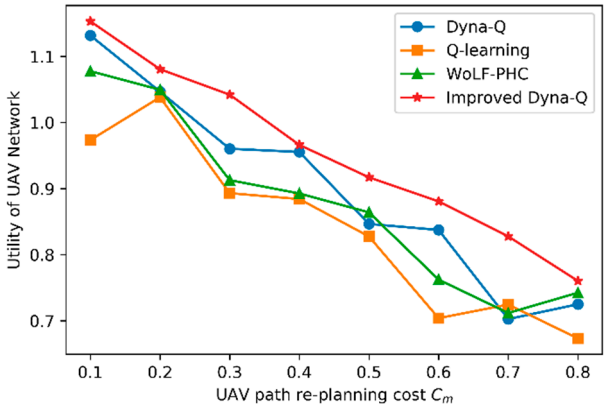
Table 4. Average SINR values of the UAV network over nine rounds of smart jamming attack.

	Dyna-Q	Q-Learning	WoLF-PHC	Improved Dyna-Q
1	2.89319	2.71596	2.89776	3.12693
2	2.92946	2.68807	2.87904	3.04405
3	2.85423	2.62658	2.86797	2.98408
4	2.98676	2.67052	2.90120	2.98828
5	2.86721	2.75111	2.94359	3.06887
6	2.99363	2.75530	2.90655	3.03335
7	2.99477	2.65255	2.84658	3.07880
8	2.86605	2.75300	2.91342	3.03412
9	2.98484	2.75873	2.89929	3.07651

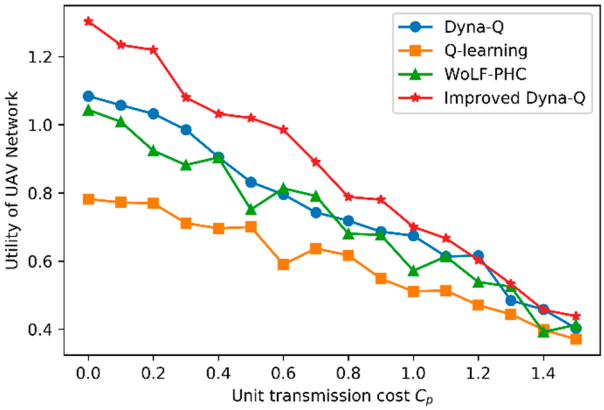
The last experiment was used to test the effect of changes in key parameters on UAV network performance with $P_T = P_J = 0.4$, $C_p \in [0, 1.5]$, $C_h \in [0.1, 0.8]$, and $C_m \in [0.1, 0.8]$. The key parameters we chose are three costs, which are frequency hopping cost, C_h; UAV path replanning cost, C_m; and unit transmission cost, C_p. The simulation results are shown in Figure 8 and Tables 5–7. The same effect of the three cost parameters on network performance is that the utility decreases almost linearly as the cost increases. For instance, the utility of the UAV network decreases 68.0% with Dyna-Q if the frequency hopping cost C_h changes from 0.1 to 0.8. The utility of the UAV network decreases 34.1% w improved Dyna-Q if the UAV path re-planning cost C_m changes from 0.1 to 0.8. The utility of UAV network decreases 60.4% with WoLF-PHC if the unit transmission cost C_p changes from 0.1 to 1.5.



(**a**) Average utility of UAV network with $P_T = P_J = 0.4$, $C_m = 0.2$, $C_p = 0.2$, and $C_h \in [0.1, 0.8]$.



(**b**) Average utility of UAV network with $P_T = P_J = 0.4$, $C_h = 0.2$, $C_p = 0.4$, and $C_m \in [0.1, 0.8]$.



(**c**) Average utility of UAV network with $P_T = P_J = 0.4$, $C_h = 0.2$, $C_m = 0.2$, and $C_p \in [0, 1.5]$.

Figure 8. Average performance of the SDN-based communication scheme with $C_h \in [0.1, 0.8]$, $C_m \in [0.1, 0.8]$, and $C_p \in [0, 1, 5]$, respectively.

Table 5. Average utility values with different frequency hopping costs.

C_h	Dyna-Q	Q-Learning	WoLF-PHC	Improved Dyna-Q
0.1	1.26438	0.82832	1.10132	1.21888
0.2	1.14760	0.77978	1.03504	1.23324
0.3	1.02622	0.64694	1.00254	1.06713
0.4	0.95013	0.58889	0.80681	0.99601
0.5	0.80474	0.66041	0.66686	0.90830
0.6	0.68610	0.51764	0.57135	0.82070
0.7	0.49812	0.49204	0.48520	0.80065
0.8	0.40412	0.34969	0.37415	0.69923

Table 6. Average utility values with different UAV path replanning costs.

C_m	Dyna-Q	Q-Learning	WoLF-PHC	Improved Dyna-Q
0.1	1.13170	0.97344	1.07771	1.15310
0.2	1.04668	1.03824	1.04958	1.08062
0.3	0.96030	0.89309	0.91283	1.04212
0.4	0.95543	0.88420	0.89263	0.96616
0.5	0.84655	0.82759	0.86404	0.91704
0.6	0.83745	0.70395	0.76227	0.88063
0.7	0.70258	0.72456	0.71178	0.82772
0.8	0.72501	0.67328	0.74244	0.76028

Table 7. Average utility values with different unit transmission costs.

C_p	Dyna-Q	Q-Learning	WoLF-PHC	Improved Dyna-Q
0	1.08415	0.78196	1.04347	1.30321
0.1	1.05738	0.77221	1.00889	1.23439
0.2	1.03222	0.76924	0.92380	1.21956
0.3	0.98544	0.71159	0.88192	1.07993
0.4	0.90506	0.69605	0.90408	1.03215
0.5	0.83172	0.70019	0.75123	1.01979
0.6	0.79568	0.59038	0.81342	0.98556
0.7	0.74280	0.63702	0.79098	0.88988
0.8	0.71846	0.61700	0.68024	0.78838
0.9	0.68653	0.54869	0.67719	0.78033
1.0	0.67443	0.51113	0.57168	0.70036
1.1	0.61374	0.51350	0.61359	0.66690
1.2	0.61592	0.47066	0.53846	0.60285
1.3	0.48464	0.44424	0.52505	0.53345
1.4	0.45818	0.39884	0.39196	0.45725
1.5	0.40313	0.37066	0.41293	0.43857

It can be seen from the simulation results that each of the three cost parameters has a particular pattern. The pattern of the frequency hopping cost C_h is that when the cost increases, the other three algorithms decrease rapidly, except for the slow decline of the improved Dyna-Q. This is because when the frequency jump cost increases, path replanning becomes the main strategy to avoid jamming. At this time, the SDN-based improved Dyna-Q can use the whole network situation information to better avoid jamming. The characteristic of UAV path replanning cost C_m is that when it increases, the UAV will use path planning less to avoid jamming. At this time, the problem gradually turns into the traditional problem of using a power allocation method to avoid jamming. In this case, the advantages of improved Dyna-Q will gradually diminish. As unit transmission cost C_p grows, any form of communication cost in the network will increase, because any type of network defense strategy relies on packet transmission. Consistent with expectations, as unit transmission cost C_p grows, the utility declines in roughly the same proportion, regardless of the algorithm used.

6. Discussion and Conclusions

In this paper, we propose a dual-controller cooperative SDN-based UAV network wireless communication scheme and design a Dyna-Q-based reinforcement learning algorithm using power allocation and track planning collaborative optimization against smart jamming. The proposed Dyna-Q algorithm has faster convergence speed and more stable performance than other three algorithms. Researchers have applied many algorithms, such as WoLF-PHC, Q-learning, DQN, and fast-DQN, to study the interactions between smart attackers and smart defenders. The DQN and fast-DQN algorithms belong to deep reinforcement learning algorithms, which need a large amount of calculation resources and have high energy consumption. Although these kinds of algorithms can reach higher performance, it is neither economical nor realistic to deploy such a large amount of calculation resources on the UAV platform. WoLF-PHC is a practically decentralized learning algorithm. It is a simple and practical algorithm for mixed-strategies learning. It does not need to know the recent behaviors of the agent and the current strategy of the opponent. However, the algorithm does not prove that it can converge to the Nash equilibrium strategy, and the stability of the algorithm is insufficient. The Q-learning algorithm is the most basic model-free reinforcement learning algorithm. The temporal-difference (TD) learning idea is the basis of many reinforcement learning algorithms. However, the UAV network has too much action space. The Q-learning algorithm can only update one Q value in an episode, and the learning efficiency is low. The method is not based on any model, which makes the SDN-based UAV network unable to use its cooperative sensing ability. The Dyna-Q algorithm is often overlooked because its environment model is difficult to build. The SDN-based UAV network has a large number of sensors and its state information can quickly converge to the network controller, which can be used to build an environment model. Therefore, the Dyna-Q-based reinforcement learning algorithm is more suitable to solve the smart defense problem of UAV networks.

Author Contributions: Writing—original draft preparation, Z.L.; supervision, Y.L. (Yu Lu); data curation, Y.S.; validation, Z.W., W.Q., Y.L. (Yicen Liu).

Funding: This research received no external funding.

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

References

1. Zhang, X.; Wang, H.J.; Zhao, H.T. An SDN Framework for UAV Backbone Network towards Knowledge Centric Networking. In Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops, Honolulu, HI, USA, 15–19 April 2018; IEEE: New York, NY, USA, 2018; pp. 456–461.
2. White, K.J.S.; Denney, E.; Knudson, M.D.; Mamerides, A.K.; Pezaros, D.P. A Programmable SDN Plus NFV-Based Architecture for UAV Telemetry Monitoring. In Proceedings of the 2017 14th IEEE Annual Consumer Communications & Networking Conference, Las Vegas, NV, USA, 8–11 January 2017; pp. 522–527.
3. Zhao, Z.L.; Cumino, P.; Souza, A.; Rosário, D.; Braun, T.; Cerqueira, E.; Gerla, M. Software-defined unmanned aerial vehicles networking for video dissemination services. *Ad Hoc Netw.* **2019**, *83*, 68–77. [CrossRef]
4. Alioua, A.; Senouci, S.-M.; Moussaoui, S.; Sedjelmaci, H.; Messous, M.-A. Efficient Data Processing in Software-Defined UAV-Assisted Vehicular Networks: A Sequential Game Approach. *Wirel. Pers. Commun.* **2018**, *101*, 2255–2286. [CrossRef]
5. Barritt, B.; Kichkaylo, T.; Mandke, K.; Zalcman, A.; Lin, V. Operating a UAV Mesh & Internet Backhaul Network Using Temporospatial SDN. In Proceedings of the 2017 IEEE Aerospace Conference, Big Sky, MT, USA, 4–11 March 2017; IEEE: Piscataway, NJ, USA, 2017.
6. Kirichek, R.; Vladyko, A.; Paramonov, A. Software-Defined Architecture for Flying Ubiquitous Sensor Networking. In Proceedings of the 2017 19th International Conference on Advanced Communication Technology (ICACT), Bongpyeong, Korea, 19–22 February 2017; IEEE: Piscataway, NJ, USA, 2018.

7. Rahman, S.U.; Kim, G.-H.; Cho, Y.-Z.; Khan, A. Deployment of an SDN-based UAV Network: Controller Placement and Tradeoff Between Control Overhead and Delay. In Proceedings of the 2017 International Conference on Information and Communication Technology Convergence, Jeju, Korea, 18–20 October 2017; pp. 1290–1292.
8. Ramaprasath, A.; Srinivasan, A.; Lung, C.-H.; St-Hilaire, M. Intelligent Wireless Ad Hoc Routing Protocol and Controller for UAV Networks. In *Ad Hoc Networks, Proceedings of the 8th International Conference, ADHOCNETS 2016, Ottawa, ON, Canada, 26–27 September 2016*; Zhou, Y., Kunz, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2017; pp. 92–104.
9. Toufga, S.; Abdellatif, S.; Owezarski, P.; Villemur, T. OpenFlow Based Topology Discovery Service in Software Defined Vehicular Networks: Limitations and Future Approaches. In Proceedings of the 2018 IEEE Vehicular Networking Conference (VNC), 2018; IEEE: Taipei, Taiwan, China, 5–7 December 2018.
10. Ur Rahman, S.; Cho, Y.Z. UAV positioning for throughput maximization. *EURASIP J. Wirel. Commun. Netw.* **2018**, *1*, 31. [CrossRef]
11. Kim, A.; Wampler, B.; Goppert, J. Cyber Attack Vulnerabilities Analysis for Unmanned Aerial Vehicles. In Proceedings of the Infotech@ Aerospace, Hilton London Olympia, London, UK, 19—20 June 2013.
12. Strohmeier, M.; Lenders, V.; Martinovic, I. On the Security of the Automatic Dependent Surveillance-Broadcast Protocol. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 1066–1087. [CrossRef]
13. Wesson, K.D.; Evans, B.L.; Humphreys, T.E. A combined symmetric difference and power monitoring GNSS anti-spoofing technique. In Proceedings of the 2013 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Austin, TX, USA, 3–5 December 2013.
14. Shepard, D.P.; Bhatti, J.A.; Humphreys, T.E.; Fansler, A.A. Evaluation of Smart Grid and Civilian UAV Vulnerability to GPS Spoofing Attacks. In Proceedings of the Radionavigation Laboratory Conference Proceedings, Nashville, TN, USA, 19–21 September 2012.
15. Manesh, M.R.; Kaabouch, N. Analysis of vulnerabilities, attacks, countermeasures and overall risk of the Automatic Dependent Surveillance-Broadcast (ADS-B) system. *Int. J. Crit. Infrastruct. Prot.* **2017**, *19*, 16–31. [CrossRef]
16. Brust, M.R.; Danoy, G.; Bouvry, P.; Gashi, D.; Pathak, H.; Gonçalves, M.P. Defending Against Intrusion of Malicious Uavs with Networked Uav Defense Swarms. In Proceedings of the 2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops), Singapore, 9 October 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 103–111.
17. Zhao, H.; Wang, H.; Wu, W.; Wei, J. Deployment algorithms for uav airborne networks toward on-demand coverage. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2015–2031. [CrossRef]
18. Wu, Y.; Wang, B.; Liu, K.R.; Clancy, T.C. Anti-jamming games in multi-channel cognitive radio networks. *IEEE J. Sel. Areas Commun.* **2012**, *30*, 4–15. [CrossRef]
19. Xiao, L.; Li, Y.; Liu, J.; Zhao, Y.F. Power control with reinforcement learning in cooperative cognitive radio networks against jamming. *J. Supercomput.* **2015**, *71*, 3237–3257. [CrossRef]
20. Tang, X.; Ren, P.; Wang, Y.; Du, Q.H.; Sun, L. Securing Wireless Transmission Against Reactive Jamming: A Stackelberg Game Framework. In Proceedings of the 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, 6–10 December 2015; IEEE: Piscataway, NJ, USA, 2015.
21. Xiao, L.; Xie, C.X.; Min, M.H.; Zhuang, W.H. User-Centric View of Unmanned Aerial Vehicle Transmission Against Smart Attacks. *IEEE Trans. Veh. Technol.* **2018**, *67*, 3420–3430. [CrossRef]
22. El-Bardan, R.; Sharma, V.; Varshney, P.K. Learning Equilibria for Power Allocation Games in Cognitive Radio Networks with a Jammer. In Proceedings of the 2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Washington, DC, USA, 7–9 December 2016; IEEE: Piscataway, NJ, USA, 2016.
23. Wang, H.; Zhao, H.; Zhang, J.; Ma, D.; Li, J.; Wei, J. Survey on Unmanned Aerial Vehicle Networks: A Cyber Physical System Perspective. *arXiv* **2018**, arXiv:1812.06821.
24. Garnaev, A.; Trappe, W. The Eavesdropping and Jamming Dilemma in Multi-Channel Communications. In Proceedings of the 2013 IEEE International Conference on Communications (ICC), Budapest, Hungary, 9–13 June 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 2160–2164.

25. Xiao, L.; Jiang, D.; Xu, D.; Zhu, H.; Zhang, Y.; Poo, H.V. Two-dimensional antijamming mobile communication based on reinforcement learning. *IEEE Trans. Veh. Technol.* **2018**, *67*, 9499–9512. [CrossRef]
26. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 2018.