Hindawi
Mobile Information Systems
Volume 2022, Article ID 4743605, 16 pages
https://doi.org/10.1155/2022/4743605

Research Article

Steady-State Availability Evaluation for Heterogeneous Edge Computing-Enabled WSNs with Malware Infections

Hong Zhang,[1] Shumin Yang,[1] Guowen Wu,[1] Shigen Shen ,[2] and Qiying Cao[1]

[1]*College of Computer Science and Technology, Donghua University, Shanghai 201620, China*
[2]*Department of Computer Science and Engineering, Shaoxing University, Shaoxing, Zhejiang 312000, China*

Correspondence should be addressed to Shigen Shen; shigens@usx.edu.cn

Received 12 November 2021; Revised 26 February 2022; Accepted 22 March 2022; Published 11 April 2022

Academic Editor: Wenqing Wu

To evaluate the steady-state availability of heterogeneous edge computing-enabled wireless sensor networks (HECWSNs) with malware infections, we first propose a Stackelberg attack-defence game to predict the optimal strategies of malware and intrusion detection systems (IDSs) deployed in heterogeneous sensor nodes (HSNs). Next, we present a new malware infection model—heterogeneous susceptible-threatened-active-recovered-dead (HSTARD) based on epidemic theory. Then, considering the heterogeneity of sink sensor nodes and common sensor nodes and the malware attack correlation, we derive the state transition probability matrix of an HSN based on a semi-Markov process (SMP), as well as the steady-state availability of an HSN. Furthermore, based on a data flow analysis of HSNs, we deduce the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology. Finally, numerical analyses illustrate the influence of the IDS parameters on the optimal infection probability of malware and reveal the effect of multiple factors on the steady-state availability of HSNs, including the initial infection rate, the infection change rate, and the malware attack correlation. In addition, we present data analyses of the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology, which provide a theoretical basis for the design, deployment, and maintenance of high-availability HECWSNs.

1. Introduction

In recent years, edge computing has emerged to address computation-intensive tasks in the 5G architecture [1], for which it can deploy servers at the edges of the network and provide services for the end users. This architecture is also applicable to heterogeneous wireless sensor networks (HWSNs), which enables heterogeneous sensor nodes (HSNs) to offload computation tasks to the deployment servers through the base stations [2]. In this manner, edge computing systems have begun to provide services for HWSNs to improve the performance of HSNs.

With the popularity of low-budget smart sensors, HWSNs have attracted considerable attention from researchers in many fields, including smart transportation, smart grids, the military, and smart homes [3–9]. These applications make our lives more comfortable. However, HWSNs have the same deficiencies as WSNs: HSNs have limited energy, computational capacity, and storage capacity [10, 11]. The other concern is that HWSNs are vulnerable to malware attacks, and malware can damage HWSNs in many ways, which not only affects the performance of HWSNs but also renders HWSNs unable to provide normal services [12]. The steady-state availability of HECWSNs is one of the factor to evaluate its performance, which indicates the probability that HECWSNs are available or reliable when sensing data, transmitting data, and aggregating data during the long-term operation. To address these issues, we evaluate the steady-state availability of heterogeneous edge computing-enabled wireless sensor networks (HECWSNs) with malware infections.

Malware refers to any malicious program with intentional attacks and serious destructive power [13–17], which causes great damage to networks, computer systems, and data [18]. Due to the limited resource allocation of HSNs [19], HECWSNs are vulnerable to malware attacks [20–22]. Once a piece of malware has attacked an HECWSN successfully

through the HSN system's security vulnerabilities, it will eavesdrop on information, block networks, wastefully spend the HSN's energy, or compromise it [23], which can seriously corrupt and damage the HECWSN. This paper will study the service availability of HECWSNs, and it aims to judge the steady-state availability of HECWSNs by evaluating the performance of HSNs. In other words, when HECWSNs are infected by malware, we study whether the HECWSNs can reach a steady state and provide data acquisition, data transmission, and data processing services normally or not.

Because malware infection in HECWSNs is similar to that of epidemic in people, the epidemic model is usually adopted for reference when establishing a malware infection model for HECWSNs [24]. The classical epidemic models include SI, SIS, and SIR [14]. For example, the SIR model classifies all nodes' states into *susceptible* (*S*), *infectious* (*I*), and *recovered* (*R*). When a heterogeneous sensor node (HSN) in HECWSNs is attacked by malware, its state will undergo a series of changes. Based on the classical SIR epidemic model, considering the characteristics of malware hiding and acting, this paper proposes a heterogeneous susceptible-threatened-active-recovered-dead (HSTARD) model to describe the states of HSNs, which includes the states *susceptible* (*S*), *threatened* (*T*), *active* (*A*), *recovered* (*R*), and *dead* (*D*).

In recent years, an increasing number of researchers have studied the problem of network security with game theory [25–28]. During the process of malware infection in HECWSNs, to defend against malware, HECWSNs use a deployed intrusion detection system (IDS) in the system of HSNs to detect malware with a certain probability [29]. The malware, to prevent detection by the IDS, will attack with a certain probability. Clearly, the attack-defence process between malware and the deployed IDS is a game problem. In this attack-defence game, the attack and defence actions initiated by the malware and IDS have priority, so it is appropriate to solve the attack-defence problem based on a Stackelberg game. Therefore, we propose a Stackelberg attack-defence game (SADG) to predict the optimal infection probability of malware, where malware is the leader and the IDS is the follower.

For an HSN, its current state determines the change in the next state, independently of the previous state, which means that the HSN state transition is random. Thus, it is appropriate to describe the state transition of an HSN using a semi-Markov process (SMP). We derive the state transition probability matrix of an HSN based on an SMP and obtain the steady-state availability of an HSN.

The contributions of this paper are summarized as follows:

First, considering that the deployed IDS and malware are two agents, we establish an SADG to predict the optimal infection probability of malware. The SADG can reflect the influence of the real HECWSNs situation when players choose their strategies.

Second, considering the characteristics of malware hiding and acting, we propose an HSTARD model by adding the states *threatened*, *active*, and *dead* to the classical epidemic model SIR. The HSTARD model can not only reflect the latent characteristics of malware but also consider the influence of the optimal defence strategy of the IDS on malware infection.

Third, considering the heterogeneity of HSNs and malware attack correlations, which will jointly affect the vulnerability of HSNs, we derive the state transition probability matrix of an HSN based on an SMP. We also derive the steady-state availability of an HSN, which not only reflects the characteristics of sensor node heterogeneity but also reflects the influence of the number of malware attacks on the state of HSNs.

Fourth, considering the topological heterogeneity of HECWSNs, we derive the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology. The steady-state availability evaluation of HECWSNs can provide a theoretical basis for the design, deployment, and maintenance of high-availability HECWSNs.

The rest of the paper is organized as follows: Section 2 describes related work. Section 3 predicts the optimal infection probability of malware based on an SADG. Section 4 describes the dynamic state transition of an HSN and proposes an HSTARD model. Section 5 derives the steady-state availability of an HSN using an SMP. Section 6 describes the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology. Section 7 describes the analysis of the experimental data. Section 8 summarizes this paper.

To facilitate understanding of this paper, all symbols used are listed in Table 1.

2. Related Work

Edge computing helps improve computing efficiency and reduce the network transmission delay. Therefore, edge computing is widely used in 5G architecture, Internet of Things, and intelligent vehicle networking. Xiao et al. [30] presented reinforcement learning for edge computing to avoid jamming attacks and interference. Xu [31] used mobile edge computing technology to manage digital communities. Rimal et al. [32] discussed mobile edge computing's potential service scenarios and designed scenarios for mobile edge computing over Wi-Fi networks. Corcoran and Soumya [33] discussed edge computing that was suitable for real-time operation and low latency requirements, which extended computing capabilities and services to the edge of the network.

At present, researchers have presented many extended epidemic models for WSNs, some of which consider the heterogeneity of WSNs. Examples include the SEIRSV model, containing states *S*, *E* (*exposed*), *I*, *R*, and *V* (*vaccination*); the susceptible-infected-immunized (SII) model [34]; the worm infection model considering the spatial-temporal perspective [35]; the susceptible-active-dormant-immune (SADI) model considering the hierarchical structure [36]; and the susceptible-exposed-infected-recovered-susceptible with vaccination and quarantine states (SEIRS-QV) model considering user awareness and network delay [37].

To date, many researchers have studied malware infection in HWSNs through game theory. Lalropuia and Gupta

Table 1: Symbols.

Symbol	Description
H	Number of HSNs in HECWSNs infected by malware.
G	State set of an HSN in HECWSNs infected by malware.
$p^x(t)$	Probability of an HSN being in state x at time t.
$p^{xy}(t)$	Probability of an HSN changing from state x to y at time t.
k	Degree of an HSN.
$\beta(k)$	Initial infection probability of an HSN with degree k.
n	Number of neighbour nodes in *active* state of an HSN.
L	Number of malware attacks on an HSN.
α	Optimal infection probability of malware.
φ	Death probability of an HSN.
θ	Update probability of dead HSNs.
δ	Probability of an HSN changing from state R to S.
ξ	Detection rate of the IDS.
ν_x	Probability of an HSN transitioning into state x.
π_x	Steady-state probability of an HSN staying in state x.
$\mathbf{P}$	State transition matrix of an HSN.
$A_j^h(k)$	Steady-state availability of an HSN with degree k of type h with topology j.
X_s	Total number of common sensor nodes connecting to a sink sensor node in HECWSNs with star topology.
m_s	Minimum number of common sensor nodes connecting to a sink sensor node in HECWSNs with star topology.
N_s	Number of common working sensor nodes connecting to a sink sensor node in HECWSNs with star topology.
$A_{ss}(k)$	Steady-state availability of a sink sensor node with degree k in HECWSNs with star topology.
A_{Star}	Steady-state availability of HECWSNs with star topology.
X_{cs}	Total number of common sensor nodes connecting to a cluster sensor node in HECWSNs with cluster topology.
A_c^c	Steady-state availability of a cluster head node in HECWSNs with cluster topology.
A_c^{cs}	Steady-state availability of a common sensor node in a cluster in HECWSNs with cluster topology.
$A_{\text{cluster}}^{\text{sink}}$	Steady-state availability of a sink sensor node in HECWSNs with cluster topology.
X_c	Total number of cluster head nodes connecting to a sink sensor node in HECWSNs with cluster topology.
m_c	Minimum number of cluster head nodes to which a sink sensor node needs to connect in HECWSNs with cluster topology.
N_c	Number of cluster head working nodes connecting to a sink sensor node in HECWSNs with cluster topology.
A_{Cluster}	Steady-state availability of HECWSNs with cluster topology.
X_{mesh}^c	Total number of common sensor nodes connecting to a sink sensor node in HECWSNs with mesh topology.
m_{mesh}^{nc}	Minimum number of common sensor nodes to which a sink sensor node needs to connect in HECWSNs with mesh topology.
N_{mesh}^c	Number of common sensor nodes connecting to a sink sensor node in HECWSNs with mesh topology.
X_{mesh}^s	Total number of sink sensor nodes in HECWSNs with mesh topology.
m_{mesh}^{ns}	Minimum number of sink sensor nodes that are required for normal work in HECWSNs with mesh topology.
N_{mesh}^s	Number of sink working sensor nodes in HECWSNs with mesh topology.
A_{mesh}^c	Steady-state availability of a common sensor node in HECWSNs with mesh topology.
A_{mesh}^s	Steady-state availability of a sink sensor node in HECWSNs with mesh topology.
A_{Mesh}	Steady-state availability of HECWSNs with mesh topology.

[38] developed an availability model to resolve the problem of the unavailability of a 5G WCN attacked by a denial of service (DoS) attack using a Bayesian game. Jiang et al. [39] established an attack-defence game based on a Stackelberg game to study the reliability of WSNs. Shen et al. [40] set up a dependability assessment mechanism for HWSNs using a noncooperative non-zero-sum game. Shen et al. [41] formulated a malware-defence differential game to

study the decision-making problem between an IDS and malware. Liu et al. [42] proposed a stochastic evolutionary coalition game (SECG) to study the reliable service of virtual-sensor-service nodes. Shen et al. [43] proposed a malware detection strategy for the Internet of Things based on a signalling game. Liu et al. [44] proposed a Bayesian Q-learning game to study the problem of task offloading in sensor edge clouds. Liu et al. [45] also proposed a Stackelberg game to study the problem of malware infection defence in sensor edge clouds based on a deep neural network.

To date, researchers have presented many models for evaluating network stability or availability, which have provided a reference for evaluating the availability of HWSNs. Famila et al. [46] proposed a cluster head selection technique by semi-Markov prediction to enhance WSNs' availability. Kanchana and Ganesan [47] proposed an inspired self-aware cooperative scheme using an SMP to prolong WSNs' lifetimes. Shakya et al. [48] proposed a correlation-based susceptible-infectious-recovered epidemic model to study the stability of WSNs. Tang et al. [49] built an evaluation approach for WSNs' availability based on Markov chains. Gour et al. [50] studied the availability of service function chains in 5G transport networks by designing a slice. Pereira et al. [51] proposed an availability analysis model to evaluate the availability of edge and fog nodes based on Markov chains.

From these references, it can be seen that there are some problems regarding the availability of HECWSNs with malware infections, which need to be resolved. The first issue is how to determine the optimal infection probability of malware. The second issue is how to accurately describe the state of HSNs in HECWSNs with malware infections. The third issue is how to describe the heterogeneity of HSNs and the correlation of malware attacks on HSNs. The fourth issue is how to obtain the steady-state availability of HECWSNs. To address the first issue, we calculate the optimal infection probability of malware through an SADG. To address the second issue, we describe the state of HSNs by adding the states *threatened* and *active* and propose a malware infection HSTARD model. To address the third issue, considering the heterogeneity of security defence policies of HSNs and the malware attack correlation, we deduce the steady-state availability of an HSN based on an SMP. To address the fourth issue, considering the topology heterogeneity of HECWSNs, we deduce the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology.

3. HSTARD Model

We assume an HECWSN scenario, which is composed of HWSNs, a base station (BS), and edge computing servers (ECServers), where the HWSNs are based on various topologies, including the star topology, cluster topology, and mesh topology (see Figure 1). To provide services for the HWSNs, edge computing servers are deployed at the BS. According to the computing tasks, the HWSNs determine whether computation services are offloaded to ECServers via wireless links.

To clearly describe the relationship among HSNs in an HECWSNs with malware infection, the HECWSN is described as an undirected network denoted by $T = (V, B)$. Here, V represents HSNs and B represents the connection between any two HSNs, indicating "connected" by 1 and "not connected" by 0. The total number of HSNs is set as H, and the degree of a heterogeneous sensor node represents the number of other heterogeneous sensor nodes connected to it, which is denoted by k.

All HSNs can be roughly divided into two types by their functions: sink sensor nodes and common sensor nodes. Sink sensor nodes are responsible for pooling and processing the data collected by common sensor nodes. Compared with common sensor nodes, sink sensor nodes are configured with much stronger computing resources, storage resources, energy, and security defence policies. Considering that these two sensor node types differ in their security defence policies, they have different initial infection probabilities. Sink sensor nodes are configured with stronger security defence policies, so they are not easily infected. The probability of initial infection is related to their degree k. The higher the degree is, the easier it is to infect the node, and the higher the initial infection probability is. While common sensor nodes have weaker security defense policies, therefore, they are more likely to be infected. We set the initial infection probability of all common sensor nodes to be the same. Suppose the initial infection probability of an HSN with degree k, denoted by $\beta(k)$, is set as

$$\beta(k) = \begin{cases} \beta_0, & \text{common sensor nodes,} \\ \beta_s(k), & \text{sink sensor nodes.} \end{cases} \tag{1}$$

In HECWSNs with malware infections, when an HSN is infected by malware, its states will undergo a series of changes. We propose an HSTARD model containing five states: S, T, A, R, and D. The state set of HSNs is expressed as $G = \{S, T, A, R, D\}$. An HSN in state S is vulnerable to malware due to its weak security defence policy. An HSN in state T has been infected by malware, but the resident malware is hidden, and the HSN cannot spread the malware to other HSNs. An HSN in state A has been infected by malware, and the resident malware is active, so the HSN will spread the malware to other HSNs. An HSN in state R is immune to the known malware because the system has been patched or the resident malware has been removed by the IDS. An HSN in state D is unable to provide normal services due to malware attacks or energy consumption.

The dynamic state transitions of an HSN are illustrated in Figure 2, where the state transitions are caused by the actions of the HSNs and malware. For an HSN in state S, when it is infected by malware, its state will transition to T; when it is patched by the IDS, its state will transition to R. For an HSN in state T, when the resident malware is activated, its state will transition to A; when it is patched by the IDS, its state will also transition to R. For an HSN in state A, when the resident malware is detected and removed by

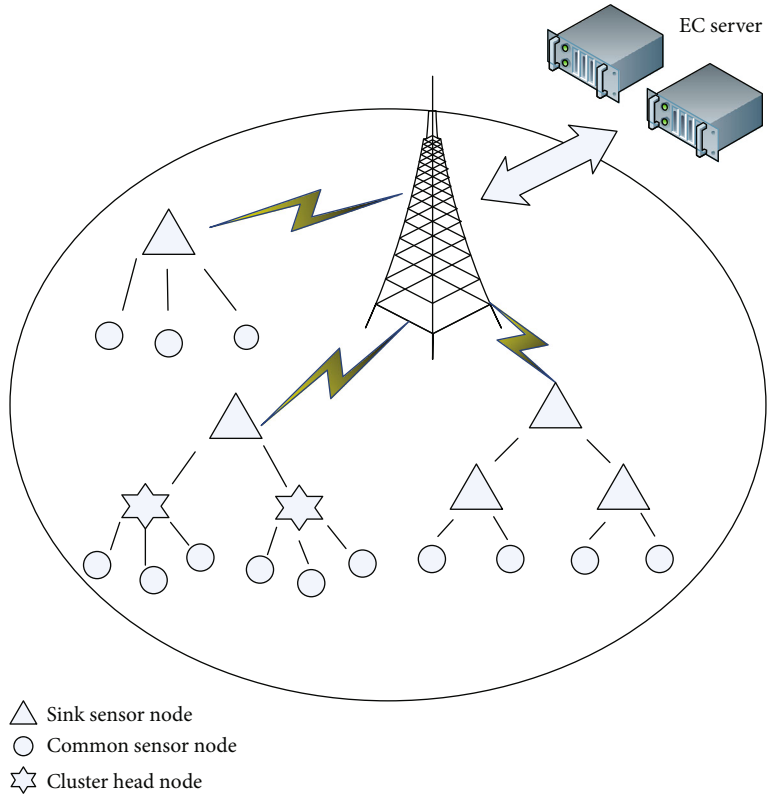

Figure 1: HECWSN model.

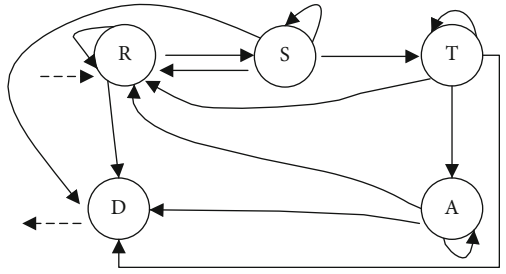

Figure 2: State transition model of an HSN.

the IDS, its state will transition to R. For an HSN in state R, when it is scanned for security vulnerabilities by new malware, its state will transition to S. When an HSN is unable to provide normal services due to malware attacks or energy consumption, its state will transition to D. To ensure that there are enough HSNs in the HECWSNs, the HSNs in state D are updated regularly; at the same time, new HSNs in state R are added.

For HSNs in all states, the death probability due to environmental influence, malware infection, and physical damage is set as φ. At time t, the probability that HSN i is in state $x(x \in G)$ is denoted by ${p_i}^x(t)$, and the probability that HSN i transitions from state $x(x \in G)$ to $y\,(y \in G)$ is denoted by ${p_i}^{xy}(t)$. Suppose the initial state of all HSNs is R; thus, for any HSN i, we have $p_i^R(0) = 1$ and $p_i^S(0) = p_i^T(0) = p_i^A(0) = p_i^D(0) = 0$.

For an HSN in state S, when its security vulnerabilities are detected by the IDS, it will be patched, and its state will transition to R with probability ξq. Here, ξq is the probability of successful detection by the IDS, where ξ is the detection probability of the IDS and q is the probability of the IDS choosing the strategy *detection*, which is calculated by the proposed SADG. When it communicates with neighbouring nodes, if it is not infected by any neighbouring node, its state will still be S with probability $\prod_{j=1}^{k}[1 - \alpha p_j^A(t-1)]$, where α is the optimal infection probability of the malware, and it is calculated by the proposed SADG. In addition, it has a death probability φ; therefore, when it communicates with its neighbouring nodes in state A, its state transitions to T with probability $1 - \prod_{j=1}^{k}[1 - \alpha p_j^A(t-1)] - \varphi - \xi q$. Thus, at time t, for HSN i with degree k in state S, its state transition probabilities are expressed as

$$\begin{cases} {p_i}^{SR}(t) = \xi q, \\ {p_i}^{SD}(t) = \varphi, \\ {p_i}^{SS}(t) = \prod_{j=1}^{k}\left[1 - \alpha p_j^A(t-1)\right], \\ {p_i}^{ST}(t) = 1 - \varphi - \xi q - \prod_{j=1}^{k}\left[1 - \alpha p_j^A(t-1)\right]. \end{cases} \tag{2}$$

For an HSN in state T, when its security vulnerabilities are detected by the IDS, it will be patched, and its state will transition to R with probability ξq. When it is continuously attacked by malware, in view of the malware attack correlation, the probability that it is successfully activated is related to the number of malware attacks. For an HSN with degree k, suppose the number of its neighbouring nodes in state A is $n\,(0 \le n \le k)$ and the number of malware attacks is set as $L\,(0 \le L \le n \le k)$. The greater L is, the greater the probability that resident malware will be activated is. Once the resident malware is activated, its state will transition to A with probability $\beta(k) + \Delta\beta \sum_{j=1}^{n} L\,(C_n^j \alpha^j (1-\alpha)^{n-j} / {p_i}^{ST})$, where j is a random variable that follows a binomial distribution [52]. In addition, it has a death probability φ. Thus, at time t, for HSN i with degree k in state T, its state transition probabilities are expressed as

$$\begin{cases} {p_i}^{TR}(t) = \xi q, \\ {p_i}^{TD}(t) = \varphi, \\ {p_i}^{TA}(t) = \beta(k) + \Delta\beta \sum_{j=1}^{n} \left[L \dfrac{C_n^j \alpha^j (1-\alpha)^{n-j}}{{p_i}^{ST}} \right], \\ {p_i}^{TT}(t) = 1 - \xi q - \varphi - \beta(k) - \Delta\beta \sum_{j=1}^{n} \left[L \dfrac{C_n^j \alpha^j (1-\alpha)^{n-j}}{{p_i}^{ST}} \right], \end{cases} \tag{3}$$

where $\beta(k)$ is the initial infection rate and $\Delta\beta$ is the infection change rate. These variables indicate the variation characteristics of the probability of HSNs being infected by malware. In addition, $\Delta\beta$ indicates the sensitivity of the probability of HSN infection by malware to the number of malware attacks. As the number of malware attacks increases, the larger $\Delta\beta$ is, the more sensitive the probability of malware infection is to the number of malware attacks; correspondingly, the greater the probability of HSNs being infected by malware is.

For an HSN in state A, when it is detected by the IDS, the resident malware will be removed, and the system will be patched; its state will transition to R with probability ξq. In addition, it has a death probability of φ, so it remains in state A with a probability of $1 - \xi q - \varphi - \theta$. Thus, at time t, for HSN i in state A, its state transition probabilities are expressed as

$$\begin{cases} {p_i}^{AR}(t) = \xi q, \\ {p_i}^{AD}(t) = \varphi, \\ {p_i}^{AA}(t) = 1 - \xi q - \varphi. \end{cases} \tag{4}$$

For an HSN in state R, its state will transition to S after it is scanned for security vulnerabilities by new malware, and the probability of its state transitioning from R to S is denoted by δ. In addition, it has a death probability of φ, so it remains in state R with probability $1 - \delta - \varphi$. Thus, at time t, for HSN i in state R, its state transition probabilities are expressed as

$$\begin{cases} {p_i}^{RD}(t) = \varphi, \\ {p_i}^{RS}(t) = \delta, \\ {p_i}^{RR}(t) = 1 - \delta - \varphi. \end{cases} \tag{5}$$

For an HSN in state D, when it is updated regularly, the newly added HSNs are in state R, and the update probability of dead HSNs is denoted by θ. Thus, at time t, for HSN i in state D, its state transition probabilities are expressed as

$$\begin{cases} {p_i}^{DD}(t) = 1 - \theta, \\ {p_i}^{DR}(t) = \theta. \end{cases} \tag{6}$$

4. A Stackelberg Attack-Defence Game for Predicting the Optimal Infection Probability of Malware

4.1. Defining a Stackelberg Attack-Defence Game. The Stackelberg game is a noncooperative strategic game, and its strategies have priority. We establish an SADG to predict the optimal infection probability of malware in HECWSNs. Note that the aim of the SADG is not to deduce the detection probability of an IDS. In this SADG, the malware makes the attack strategy first, and then, the IDS chooses the defence strategy. In other words, the malware is the leader, and the IDS is the follower.

Definition 1. The Stackelberg attack-defence game (*SADG*) is expressed by a 3-tuple $<B, E, U>$, where

(i) $B = \{\text{malware}\,(M), \text{IDS}\,(Z)\}$ denotes a set of players

(ii) $E = E_M \times E_Z$, where $E_M = \{\text{Infect}(I), \text{Noninfect}\,(\widetilde{\varnothing})\}$ and $E_Z = \{\text{Detect}\,(D), \text{Nondetect}\,(\varnothing)\}$ denote the pure strategies sets that malware and IDS can choose, respectively

(iii) $U = [U^b_{e_M e_Z}]$, for $b \in B$, $e_M \in E_M$, and $e_Z \in E_Z$, denotes a payoff matrix, where $U^b_{e_M e_Z} : e_M \times e_Z \mapsto \mathbb{R}$ denotes the payoff that player b obtains when player M selects the pure strategy e_M and player Z selects the pure strategy e_Z

The parameters used in the SADG are listed in Table 1. ξ $(0 < \xi < 1)$ and ζ $(0 < \zeta < 1)$ denote the detection rate and the false alarm rate of the IDS, respectively. C_I $(C_I > 0)$ denotes the infection cost of the malware, C_D $(C_D > 0)$ denotes the detection cost of the IDS, ω_I $(\omega_I > 0)$ denotes the utility of the malware for successful infection, and ω_D denotes the utility of the IDS for successful detection.

Definition 1 shows that the SADG has two players: malware (M) and IDS (Z). The two pure strategies for M to choose are Infect (I) and Noninfect$(\widetilde{\varnothing})$, and the two pure strategies for S to choose are Detect (D) and Nondetect$(\varnothing)$.

Therefore, these two players have four combinations of pure strategies in the SADG, and the corresponding utilities form the payoff matrix (see Table 2).

For the combination of pure strategies {Infect, Detect}, which means that malware selects the pure strategy Infect, and IDS selects the pure strategy Detect, the detection rate and the false alarm rate of the IDS are ξ and ζ, so it gains $\xi\omega_D$ and loses $\zeta\omega_D$, while malware loses $\xi\omega_I$ and gains $\zeta\omega_I$. In response, the rate of malware infection is $1-\xi$, so malware gains $(1-\xi)\omega_I$ and IDS loses $(1-\xi)\omega_D$. In addition, the infection cost of malware is C_I, so malware loses C_I. The detection cost of IDS is C_D, so it loses C_D. Therefore, the utilities of malware U_{ID}^M and IDS U_{ID}^Z are, respectively,

$$\begin{aligned} U_{ID}^M &= (\zeta - 2\xi + 1)\omega_I - C_I, \\ U_{ID}^Z &= (2\xi - 1 - \zeta)\omega_D - C_D. \end{aligned} \tag{7}$$

For the combination of pure strategies {Infect, Nondetect}, which means that malware selects the pure strategy Infect and IDS selects the pure strategy Nondetect. Under this case, malware gains ω_I and IDS loses ω_D. In addition, the infection cost of malware is C_I, so malware loses C_I. Therefore, the utilities of malware $U_{I\varnothing}^M$ and IDS $U_{I\varnothing}^Z$ are, respectively,

$$\begin{aligned} U_{I\varnothing}^M &= \omega_I - C_I, \\ U_{I\varnothing}^Z &= -\omega_D. \end{aligned} \tag{8}$$

For the combination of pure strategies {Noninfect, Detect}, which means that malware selects the pure strategy Noninfect and IDS selects the pure strategy Detect, the false alarm rate of the IDS is ζ, so IDS loses $\zeta\omega_D$. In addition, the detection cost of the IDS is C_D, so IDS loses C_D. Therefore, the utilities of malware $U_{\bar{\varnothing}D}^M$ and IDS $U_{\bar{\varnothing}D}^Z$ are, respectively,

$$\begin{aligned} U_{\bar{\varnothing}D}^M &= 0, \\ U_{\bar{\varnothing}D}^Z &= -\zeta\omega_D - C_D. \end{aligned} \tag{9}$$

For the combination of pure strategies {Noninfect, Nondetect}, which means that malware selects the pure strategy Noninfect and IDS selects the pure strategy Nondetect, the utilities of malware $U_{\bar{\varnothing}\varnothing}^M$ and IDS $U_{\bar{\varnothing}\varnothing}^Z$ are both equal to 0.

Let q denote the probability that the follower IDS selects the pure strategy Detect. Correspondingly, the probability that it selects the pure strategy Nondetect is denoted by $1-q$. Let α denote the probability that the leader malware selects the pure strategy Infect. Correspondingly, the probability that it selects the pure strategy Noninfect is denoted by $1-\alpha$.

For IDS, the greater the probability q is, the greater the energy consumed by detection. That is, the detection cost C_D becomes greater as q increases, so we set the detection cost C_D as a linear function of the detection probability q.

Table 2: The payoff matrix of the SADG.

	Detect	Nondetect
Infect	$(\zeta - 2\xi + 1)\omega_I - C_I, (2\xi - 1 - \zeta)\omega_D - C_D$	$\omega_I - C_I, -\omega_D$
Noninfect	$0, -\zeta\omega_D - C_D$	0,0

We obtain the expected payoff of IDS as

$$U^Z = 2q\alpha\omega_D\xi - \alpha\omega_D - q\zeta\omega_D - qC_D, \tag{10}$$

and the detection cost C_D is

$$C_D = q\sigma_D, \tag{11}$$

where σ_D expresses the basic cost of detection of IDS.

Then, using (11) to replace C_D in (10), we can obtain U^Z as

$$U^Z = 2q\alpha\omega_D\xi - \alpha\omega_D - q\zeta\omega_D - q^2\sigma_D. \tag{12}$$

For malware, the greater the probability α is, the greater the attack needs to consume the resources is. That is, the infection cost C_I becomes greater as α increases, so we set the infection cost C_I as a linear function of the infection probability α. We obtain the expected payoff of malware as

$$U^M = (\zeta - 2\xi)q\alpha\omega_I + \alpha\omega_I - \alpha C_I, \tag{13}$$

and the infection cost C_I is

$$C_I = \alpha\sigma_I, \tag{14}$$

where σ_I expresses the basic cost of infection by malware.

Then, using (14) to replace C_I in (13), we can obtain U^M as

$$U^M = (\zeta - 2\xi)q\alpha\omega_I + \alpha\omega_I - \alpha^2\sigma_I. \tag{15}$$

4.2. Predicting the Optimal Infection Probability of Malware. In this section, we predict the optimal infection probability of malware by calculating the Stackelberg equilibrium of the SADG.

Theorem 1. *In the SADG, the optimal infection probability α of malware choosing the pure strategy Infect is*

$$\alpha = \frac{\zeta^2\omega_I\omega_D - 2\xi\zeta\omega_I\omega_D - \omega_I}{2\zeta\xi\omega_I\omega_D - 4\xi^2\omega_I\omega_D - \sigma_I}. \tag{16}$$

Proof. According to the equilibrium solution of the Stackelberg game, we first calculate the reaction of the follower IDS and obtain the optimal detection probability q that maximizes U^S. Then, based on the known optimal detection probability q, we can calculate the reaction of the leader malware and obtain the optimal infection probability α.

First, we calculate the first-order partial derivative of U^Z with respect to q as

$$\frac{\partial U^Z}{\partial q} = 2\alpha\omega_D\xi - \zeta\omega_D - q\sigma_D. \tag{17}$$

The second-order partial derivative of U^Z with respect to q is

$$\frac{\partial^2 U^Z}{\partial^2 q} = -\sigma_D. \tag{18}$$

Since the basic cost of detection σ_D is greater than 0, $-\sigma_D$ is less than 0. Therefore, U^Z has the maximum value. Setting (17) equal to zero, we can obtain

$$q = \frac{2\alpha\omega_D\xi - \zeta\omega_D}{\sigma_D}. \tag{19}$$

Then, we calculate the reaction of the leader malware by introducing the parameter q, and we obtain the optimal infection probability α.

The first-order partial derivative of U^M with respect to α is

$$\frac{\partial U^M}{\partial \alpha} = (\zeta - 2\xi)q\omega_I + \omega_I - \alpha\sigma_I. \tag{20}$$

The second-order partial derivative of U^M with respect to α is

$$\frac{\partial^2 U^M}{\partial^2 \alpha} = -\sigma_I. \tag{21}$$

Since the basic cost of infection σ_I is greater than 0, $-\sigma_I$ is less than 0. Therefore, U^M has the maximum value. Setting (20) equal to zero for maximization, we substitute (19) with the optimal detection probability q and obtain the optimal infection probability α as

$$\alpha = \frac{\zeta^2\omega_I\omega_D - 2\xi\zeta\omega_I\omega_D - \omega_I}{2\zeta\xi\omega_I\omega_D - 4\xi^2\omega_I\omega_D - \sigma_I}. \tag{22}$$

It can be seen from the proof and derivation that α is the optimal infection probability of malware. The proof is complete.

5. Steady-State Availability of a Heterogeneous Sensor Node

The steady state of an HSN is an important factor determining whether it can work normally. It is random, which means that the current state of an HSN determines the change in the next state independently of the previous state. Thus, it is appropriate to describe the state transition of an HSN using an SMP. Comparing the Markov process (MP) and SMP, the MP has strict requirements for the state residence time, which must follow an exponential distribution, while the SMP does not limit the state residence time, which can follow any distribution form. Therefore, modelling the dynamic state transition process of an HSN based on an SMP appears more objective, and it is more in line with practical conditions.

The transition from one state to another based on an SMP can be considered as two logical steps, which are the residence time of an HSN in state x and the transition process from state x to y.

Definition 3. Suppose the residence time of an HSN in state x is expressed as t_x; then, the t_x distribution is expressed as $T(t_x) = [t_S, t_T, t_A, t_R, t_D]$.

Definition 4. Let ν_x represent the probability that an HSN translates to state x, and let the ν_x distribution be expressed as $V(\nu_x) = [\nu_S, \nu_T, \nu_A, \nu_R, \nu_D]$.

Definition 5. The steady-state probability is the share of the residence time of an HSN in each state in the total residence time of all states. Let π_x represent the steady-state probability vector of an HSN; the π_x distribution is expressed as $X(\pi_x) = [\pi_S, \pi_T, \pi_Q, \pi_I, \pi_R, \pi_D]$, where

$$\pi_x = \frac{\nu_x t_x}{\sum_{y\in G}\nu_y t_y}, x, y \in G. \tag{23}$$

According to the proposed HSTARD model, the state transition matrix $\mathbf{P}$ of an HSN is constructed as

$$\mathbf{P} = \begin{bmatrix} p^{SS} & p^{ST} & 0 & p^{SR} & p^{SD} \\ 0 & p^{TT} & p^{TA} & p^{TR} & p^{TD} \\ 0 & 0 & p^{AA} & p^{AR} & p^{AD} \\ p^{RS} & 0 & 0 & p^{RR} & p^{RD} \\ 0 & 0 & 0 & p^{DR} & p^{DD} \end{bmatrix}. \tag{24}$$

Here, $p^{SS} = p_i^{SS}(t)$, $p^{ST} = p_i^{ST}(t)$, $p^{SR} = p_i^{SR}(t)$, $p^{SD} = p_i^{SD}(t)$, $p^{TT} = p_i^{TT}(t)$,$p^{TA} = p_i^{TA}(t)$, $p^{TR} = p_i^{TR}(t)$, $p^{TD} = p_i^{TD}(t)$, $p^{AA} = p_i^{AA}(t)$, $p^{AR} = p_i^{AR}(t)$, $p^{AD} = p_i^{AD}(t)$, $p^{RS} = p_i^{RS}(t)$, $p^{RR} = p_i^{RR}(t)$, $p^{RD} = p_i^{RD}(t)$, $p^{DR} = p_i^{DR}(t)$, and $p^{DD} = p_i^{DD}(t)$.

According to the discrete Markov properties, we obtain

$$\begin{cases} V = V\mathbf{P}, \\ \sum_{x\in G}\nu_x = 1. \end{cases} \tag{25}$$

Combining and solving (24) and (25), we can obtain

$$\begin{aligned}
v_S &= \frac{p^{RS}p^{\tilde{T}}p^{\tilde{A}}Z}{p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)},\\
v_T &= \frac{p^{RS}p^{ST}p^{\tilde{A}}Z}{p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)},\\
v_A &= \frac{p^{RS}p^{ST}p^{TA}Z}{p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)},\\
v_R &= \frac{p^{\tilde{S}}p^{\tilde{T}}p^{\tilde{A}}Z}{p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)},\\
v_D &= 1-Z.
\end{aligned} \tag{26}$$

Substituting v_S, v_T, v_A, v_R, and v_D into (25), we can obtain

$$\begin{aligned}
\pi_S &= \frac{p^{RS}p^{\tilde{T}}p^{\tilde{A}}t_SZ}{\left(p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)\right)\pi_{\text{sum}}},\\
\pi_T &= \frac{p^{RS}p^{ST}p^{\tilde{A}}t_TZ}{\left(p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)\right)\pi_{\text{sum}}},\\
\pi_A &= \frac{p^{RS}p^{ST}p^{TA}t_AZ}{\left(p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)\right)\pi_{\text{sum}}},\\
\pi_R &= \frac{p^{\tilde{S}}p^{\tilde{T}}p^{\tilde{A}}t_RZ}{\left(p^{\tilde{T}}p^{\tilde{A}}\left(p^{RS}+p^{\tilde{S}}\right)+p^{ST}p^{RS}\left(p^{\tilde{A}}+p^{TA}\right)\right)\pi_{\text{sum}}},\\
\pi_D &= \frac{(1-Z)t_D}{\pi_{\text{sum}}},
\end{aligned} \tag{27}$$

where $\pi_{\text{sum}} = \pi_s t_s + \pi_T t_T + \pi_A t_A + \pi_R t_R + \pi_D t_D$, $Z = 1-(\varphi/1+\varphi+P^{DD})$, $p^{\tilde{S}} = 1-p^{SS}$, $p^{\tilde{T}} = 1-p^{TT}$, and $p^{\tilde{A}} = 1-p^{AA}$.

Since the steady-state availability of an HSN describes its ability to continue working after being damaged or attacked by malware, an HSN in state A or D cannot work normally. Therefore, the steady-state availability of an HSN of type h in HECWSNs with topology j, denoted by A_j^h, is

$$A_j^h = 1-\pi_A-\pi_D. \tag{28}$$

6. Steady-State Availability of HECWSNs

6.1. Steady-State Availability of HECWSNs with a Star Topology. HECWSNs with a star topology are composed of a single sink sensor node and some common sensor nodes (see Figure 3). The common sensor nodes are responsible for collecting data and forwarding them to the sink sensor node, which is responsible for processing data and transferring them to the base station. Common sensor nodes work in parallel without affecting each other and are classified into a parallel system. Thus, to enable HECWSNs to work normally, two conditions should be met: (1) the sink sensor node is able to work normally, and (2) a certain number of common sensor nodes can work normally.

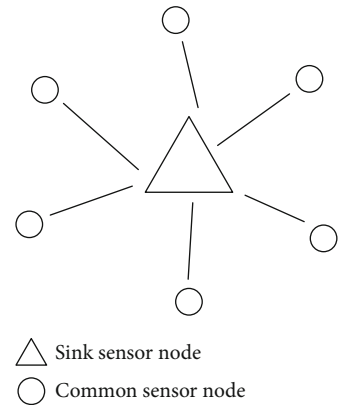

Figure 3: HECWSNs with a star topology.

Assuming that the HECWSNs with a star topology contain X_s common sensor nodes, the steady-state availability of each common sensor node is denoted by A_{star}^c. It is required that there be at least m_s common sensor nodes working normally in the HECWSNs. When the HECWSNs can work normally, the number of common working sensor nodes is denoted by N_s. In the HECWSNs, the steady-state availability of the sink sensor node is denoted by $A_{\text{star}}^{\text{sink}}(X_s)$, which is expressed as $\sum_{N_s=m_s}^{X_s} C_{X_s}^{N_s} (A_{\text{star}}^c)^{N_s}(1-A_{\text{star}}^c)^{X_s-N_s}$. The steady-state availability of the HECWSNs with a star topology, denoted by A_{Star}, is

$$A_{\text{Star}} = A_{\text{star}}^{\text{sink}}(X_s)\sum_{N_s=m_s}^{X_s} C_{X_s}^{N_s}(A_{\text{star}}^c)^{N_s}(1-A_{\text{star}}^c)^{X_s-N_s}. \tag{29}$$

6.2. Steady-State Availability of HECWSNs with a Cluster Topology. HECWSNs with a cluster topology are composed of three node types: sink sensor nodes, cluster head nodes [53], and common sensor nodes (see Figure 4). Cluster head nodes are selected from common sensor nodes [54], so the availability of cluster head nodes is equal to that of common sensor nodes. A cluster is composed of a cluster head node and some common sensor nodes. When the HECWSNs work, the common sensor nodes first communicate with the cluster head node, and then, the cluster head node communicates with the sink sensor node. As long as there is one common working sensor node in a cluster, the cluster is available.

In HECWSNs with a cluster topology, suppose that the steady-state availability of a common sensor node in a cluster is denoted by A_c^{cs} and that the number of common sensor nodes in a cluster is expressed by X_{cs}. The steady-state availability of a cluster in the HECWSNs, denoted by A_c^c, is

$$A_c^c = 1-(1-A_c^{cs})^{X_{cs}}. \tag{30}$$

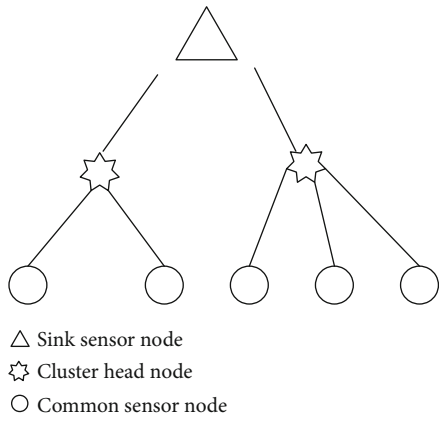

Figure 4: HECWSNs with a cluster topology.

For HECWSNs with a cluster topology, if a cluster is regarded as one unit, it can be regarded as having a star structure. Suppose that there are X_c clusters and that the steady-state availability of the sink sensor nodes is denoted by $A^{\text{sink}}_{\text{cluster}}(X_c)$. Suppose that there are at least m_c clusters required to work normally in the HECWSNs and that the number of working clusters is denoted by N_c. The steady-state availability of the HECWSNs with a cluster topology, denoted by A_{Cluster}, is

$$A_{\text{Cluster}} = A^{\text{sink}}_{\text{cluster}}(X_c) \sum_{N_c=m_c}^{X_c} \left[C^{N_c}_{X_c} \cdot (A^c_c)^{N_c} \cdot (1 - A^c_c)^{X_c - N_c} \right]. \tag{31}$$

6.3. Steady-State Availability of HECWSNs with a Mesh Topology. HECWSNs with a mesh topology usually contain several sink sensor nodes, and each sink sensor node manages some common sensor nodes (see Figure 5). The common sensor nodes are responsible for translating the collected data to the corresponding sink sensor node, and then, the sink sensor node sends data to the base station [55].

In HECWSNs with a mesh topology, communication between a sink sensor node and its common sensor nodes occurs in a parallel system (see Figure 5), so the HECWSNs can be regarded as a combination of multiple HECWSNs with star topologies. Assume that the total number of common sensor nodes connecting to the sink sensor node is X^c_{mesh}, the minimum number of common sensor nodes required to work normally is m^c_{mesh}, the steady-state availability of a common sensor node is denoted by A^c_{mesh}, and the number of common working sensor nodes connecting to the sink sensor node is denoted by N^c_{mesh}. Then, the steady-state availability of the sink sensor node, denoted by A^s_{mesh}, is

$$A^s_{\text{mesh}} = A^{\text{sink}}_m(X_m) \sum_{N^c_{\text{mesh}}=m^c_{\text{mesh}}}^{X^c_{\text{mesh}}} C^{k_m}_{X^c_{\text{mesh}}} (A^c_{\text{mesh}})^{N^c_{\text{mesh}}} (1 - A^c_{\text{mesh}})^{X^c_{\text{mesh}} - N^c_{\text{mesh}}}. \tag{32}$$

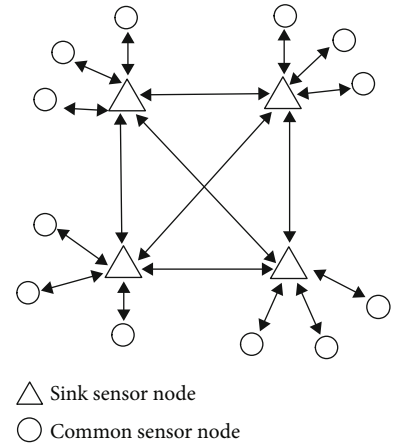

Figure 5: HECWSNs with a mesh topology.

To enable HECWSNs with a mesh topology to work normally, at least a certain number of sink sensor nodes are required to work normally. Assume that the total number of sink sensor nodes is X^s_{mesh}, the minimum number of sink sensor nodes required to work normally is m^s_{mesh}, and the number of working sink sensor nodes is denoted by N^s_{mesh}. Then, the steady-state availability of the HECWSNs with a mesh topology is denoted by A_{Mesh}, which can be expressed as

$$A_{\text{Mesh}} = \sum_{N^s_{\text{mesh}}=m^s_{\text{mesh}}}^{X^s_{\text{mesh}}} C^{X^c_{\text{mesh}}}_{X^s_{\text{mesh}}} (A^s_{\text{mesh}})^{X^c_{\text{mesh}}} (1 - A^s_{\text{mesh}})^{X^s_{\text{mesh}} - X^c_{\text{mesh}}}. \tag{33}$$

7. Experimental Simulations and Data Analysis

In this paper, simulation experiments to verify the steady-state availability of HECWSNs are completed based on MATLAB 2015. The steady-state availability of HECWSNs with malware infections is affected by many factors, including the initial infection probability $\beta(k)$, the infection change rate $\Delta\beta$, and the infection probability of malware. First, we analyse the influence of the detection rate ξ and false alarm rate ζ of the IDS on the optimal infection probability of malware. Second, we analyse the impact of the initial infection probability $\beta(k)$ and infection change rate $\Delta\beta$ on the steady-state availability of HSNs. Third, we evaluate the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology.

7.1. Optimal Infection Probability of Malware under the Influence of IDS Parameters. In this section, the experimental data show the influence of the detection rate ξ and false alarm rate ζ of the IDS on the optimal infection probability of malware. In the experiment, the relevant parameters are set as $\sigma_I = 20$, $\sigma_D = 10$, $\omega_I = 50$, and $\omega_D = 40$.

With the increase of the detection rate ξ of the IDS, the optimal infection probability of malware will decrease (see

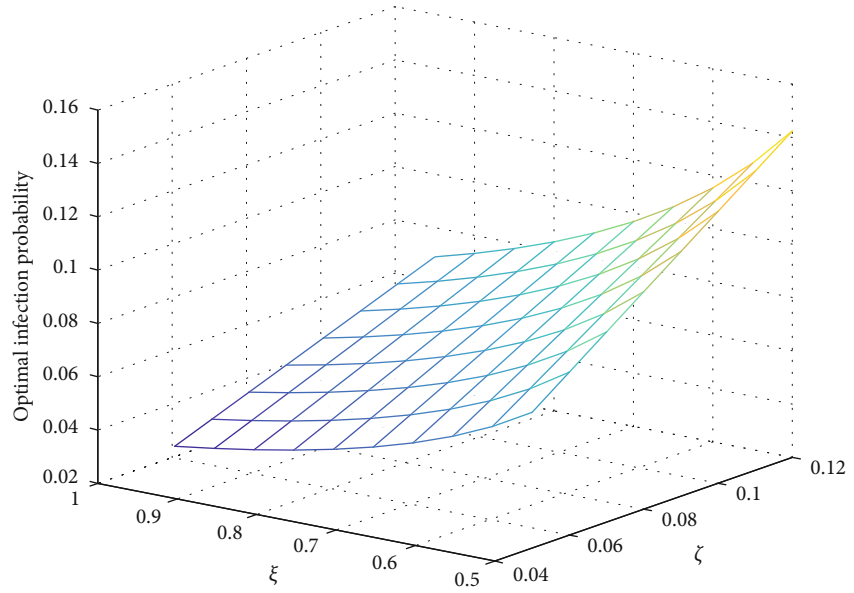

Figure 6: Influence of the IDS parameters on the optimal infection probability of malware.

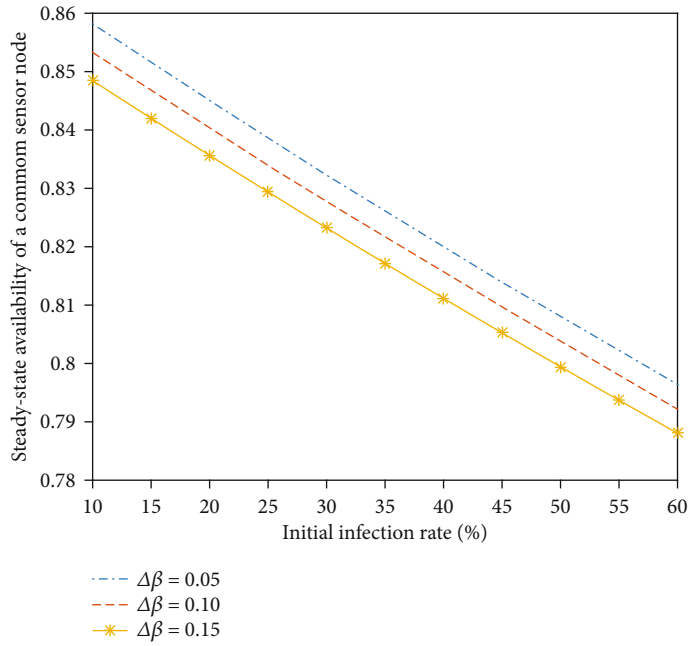

Figure 7: Influence of the initial infection rate and the infection change rate on the steady-state availability of an HSN.

Figure 6), but with the increase of the false alarm rate ζ of the IDS, the optimal infection probability of malware will also increase. For example, when the detection rate ξ of the IDS increases from 0.5 to 0.95, the optimal infection probability of malware decreases from 0.06 to 0.01 when $\zeta = 0.06$. When the false alarm rate ζ of the IDS increases from 0.04 to 0.12, the optimal infection probability of malware increases from 0.04 to 0.14 when $\xi = 0.6$.

According to the analysis of the experimental data, to decrease the optimal infection probability of malware, we should prioritize reducing the false alarm rate to improve the detection rate.

7.2. Effect of the Initial Infection Rate and the Infection Change Rate on the Steady-State Availability of an HSN. The steady-state availability of an HSN is related to the initial infection rate and the infection change rate (see Figure 7). It can be seen from the experimental data curves that with the increase of the initial infection rate and the infection change rate, the steady-state availability of an HSN decreases gradually.

For example, when the initial infection rate increases from 0.1 to 0.6, the steady-state availability of the HSN decreases from 0.8596, 0.8543, and 0.8483 to 0.8021, 0.7953, and 0.7886, corresponding to three values of the

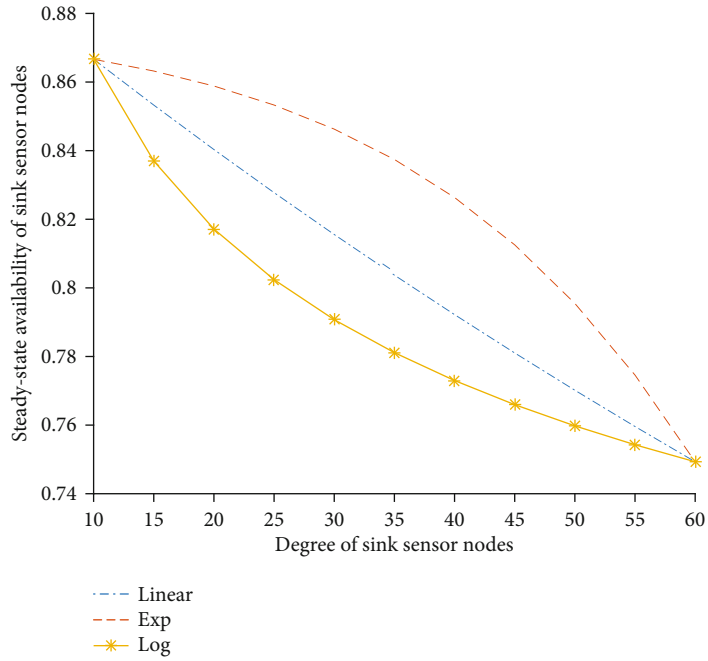

Figure 8: Influence of the degree of sink sensor nodes on the steady-state availability of the sink sensor nodes.

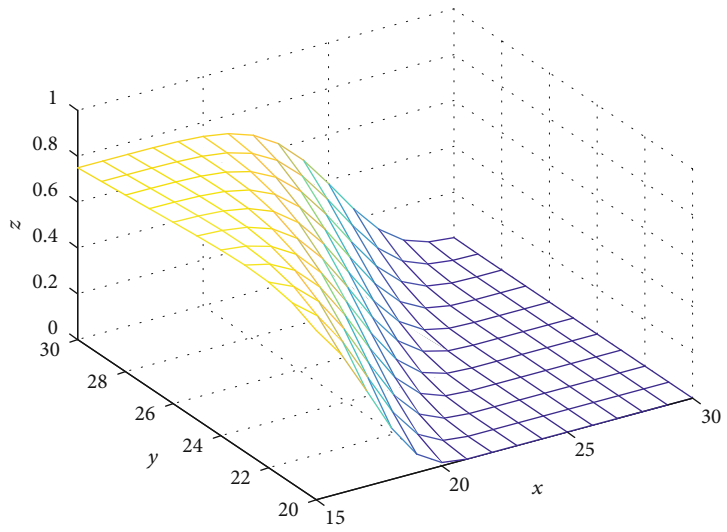

Figure 9: Steady-state availability of HECWSNs with a star topology.

infection change rate, $\Delta\beta = 0.05$, $\Delta\beta = 0.1$, and $\Delta\beta = 0.15$, respectively. The data analysis indicates that the higher the initial infection rate is, the greater the probability of HSNs being infected by malware is, and the lower the steady-state availability of the HSNs is. The higher the infection change rate is, the more difficult it is for HSNs to suppress malware infection. Therefore, to improve the steady-state availability of HSNs, we should reduce the initial infection rate and the infection change rate.

The initial infection rate and the infection change rate influence the steady-state availability of a sink sensor node (see Figure 8). It can be seen from the experimental data curves that with the increase of the sink sensor node's degree k, the steady-state availability of the sink sensor node decreases gradually. Three functions of the initial infection rate are considered in the experiment: $\beta(k) = ck$, $\beta(k) = c \log k$, and $\beta(k) = ce^k$. Figure 7 shows that when the degree k of the sink sensor node increases from 10 to 60, the steady-state availability of the sink sensor node decreases from 0.8673 to 0.7535. When the initial infection rate of the sink sensor node satisfies the power function, the steady-state availability of the sink sensor node is least affected by its degree k, followed by the linear function and then the exponential function.

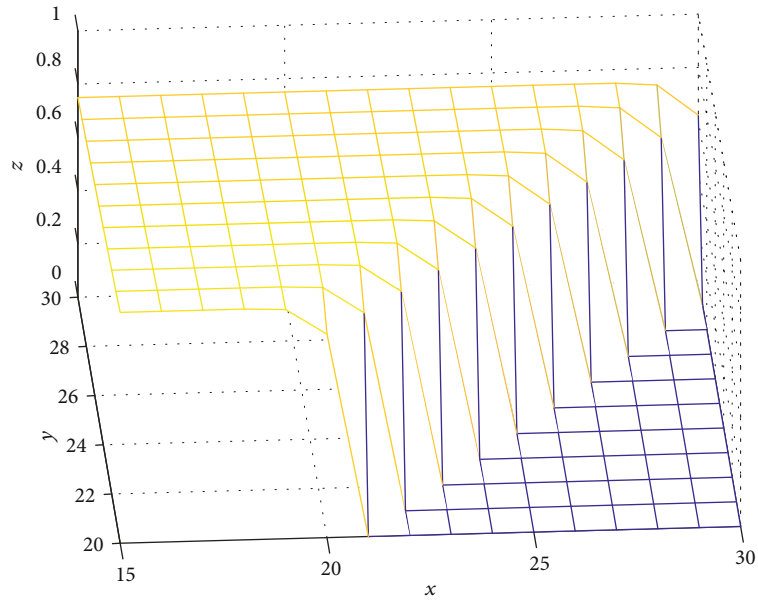

Figure 10: Steady-state availability of HECWSNs with a cluster topology.

For example, when the degree k of the sink sensor node is set as 35, the steady-state availability of the sink sensor node is 0.8565, 0.8214, and 0.7946, corresponding to three functions of the initial infection rate $\beta_S(k) = ce^k$, $\beta_S(k) = ck$, and $\beta_S(k) = c \log k$, respectively. Experimental data analysis indicates that the function of the initial infection $\beta_S(k) = ce^k$ is the most appropriate for HECWSNs.

7.3. Steady-State Availability of HECWSNs with a Star Topology. Figure 9 shows the evaluation results of the steady-state availability of HECWSNs with a star topology. The x-axis shows the number of common working sensor nodes connecting to the sink sensor node, which ranges from 15 to 30. The y-axis shows the total number of common sensor nodes connecting to the sink sensor node, which ranges from 20 to 30. The z-axis shows the steady-state availability of HECWSNs with a star topology, which ranges from 0 to 1.

With the increase in the total number of common sensor nodes connecting to the sink sensor node, the steady-state availability of the HECWSNs will increase (see Figure 9). However, with the increase in the number of common working sensor nodes connecting to the sink sensor node, the steady-state availability of the HECWSNs will decrease. For example, when the number of common working sensor nodes connecting to the sink sensor node is 20, the steady-state availability of HECWSNs with a star topology will gradually increase from 0 to 0.75 as the total number of common sensor nodes connecting to the sink sensor node increases from 20 to 30. When the total number of common sensor nodes connecting to the sink sensor node is 30, the steady-state availability of HECWSNs with a star topology will gradually decrease from 0.75 to 0 as the number of common working sensor nodes connecting to the sink sensor node increases from 15 to 30. Therefore, when constructing HECWSNs with a star topology, it is necessary to increase the number of redundant common sensor nodes according to the actual situation, which helps to increase the steady-state availability of HECWSNs with a star topology.

7.4. Steady-State Availability of HECWSNs with a Cluster Topology. Figure 10 shows the evaluation results for the steady-state availability of HECWSNs with a cluster topology. The x-axis shows the number of working clusters, which ranges from 15 to 30. The y-axis shows the total number of clusters, which ranges from 20 to 30. The z-axis shows the steady-state availability of HECWSNs with a cluster topology, which ranges from 0 to 1.

With the increase in the total number of clusters, the steady-state availability of HECWSNs with a cluster topology will increase (see Figure 10). However, with the increase in the number of working clusters, the steady-state availability of HECWSNs with a cluster topology will decrease. For example, when the number of working clusters is 25, the steady-state availability of HECWSNs with a cluster topology will gradually increase from 0 to 0.75 as the total number of clusters increases from 24 to 30. When the total number of clusters is 26, the steady-state availability of HECWSNs with a cluster topology will gradually decrease from 0.75 to 0 as the number of working clusters increases from 15 to 27. Therefore, when constructing HECWSNs with a cluster topology, it is necessary to increase the number of redundant clusters according to the actual situation, which helps to increase the steady-state availability of HECWSNs with a cluster topology.

7.5. Steady-State Availability of HECWSNs with a Mesh Topology. Figure 11 shows the evaluation results of the steady-state availability of HECWSNs with a mesh topology. The x-axis shows the number of common working sensor nodes in a subnet, which ranges from 2 to 6. The y-axis

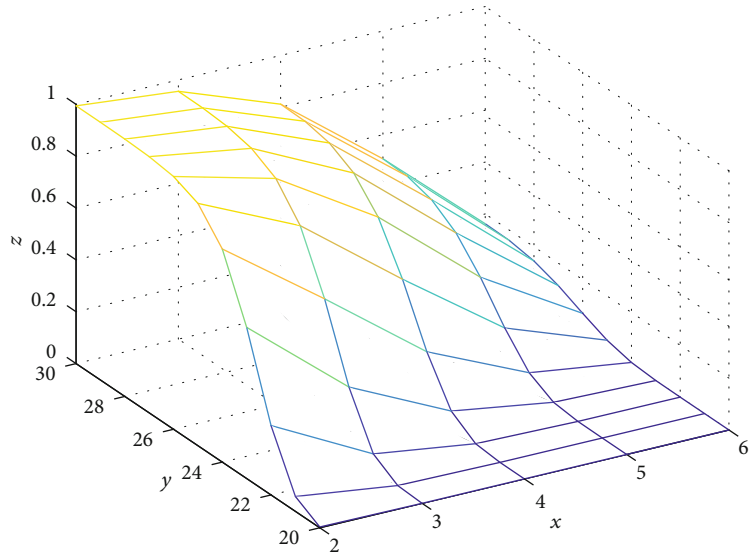

Figure 11: Steady-state availability of HECWSNs with a mesh topology.

shows the total number of common sensor nodes in a subnet, which ranges from 20 to 30. The z-axis shows the steady-state availability of HECWSNs with a mesh topology, which ranges from 0 to 1.

With the increase in the number of common sensor nodes in a subnet, the steady-state availability of HECWSNs with a mesh topology will increase (see Figure 11). However, with the increase in the number of common working sensor nodes in a subnet, the steady-state availability of HECWSNs with a mesh topology will decrease. For example, when the number of common working sensor nodes in a subnet is 2, the steady-state availability of HECWSNs with a mesh topology will increase from 0 to 1 as the total number of common sensor nodes in a subnet increases from 20 to 30. When the total number of common sensor nodes in a subnet is 30, the steady-state availability of HECWSNs with a mesh topology will decrease from 1 to 0.18 as the number of common working sensor nodes in a subnet increases from 2 to 6. Therefore, when constructing HECWSNs with a mesh topology, it is necessary to increase the number of redundant common sensor nodes in a subnet and the number of sink sensor nodes, which helps to increase the steady-state availability of HECWSNs with a mesh topology.

8. Conclusions

To suppress the spread of malware and improve the steady-state availability of HECWSNs, considering the heterogeneity of common sensor nodes and sink sensor nodes in HECWSNs with malware infections, we evaluated the steady-state availability of HECWSNs with malware infections based on an SMP and a Stackelberg game. First, we established the SADG to predict the optimal infection probability of malware. Second, extending the classical epidemic SIR model by adding states T, A, and D, we proposed an HSTARD model to describe the states of HSNs. Third, considering the influence of the malware attack correlation on the steady-state availability of HECWSNs, we obtained the steady-state availability of HECWSNs with various topologies, including the star topology, cluster topology, and mesh topology. The steady-state availability evaluation of HECWSNs provides a theoretical basis for the design, deployment, and maintenance of high-availability HECWSNs.

In future work, an even more interesting direction to consider is the heterogeneity of infection change rate of HSNs. When the influences of neighbour sensor nodes are considered, the infection change rate of HSNs is effect by their degree. In addition, another consideration is to evaluate the steady-state availability of HECWSNs with multiple topologies, in which HSNs are deployed in a variety of topologies, including star topology, cluster topology, and mesh topology.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare no conflicts of interest in this paper.

Acknowledgments

This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LZ22F020002.

References

[1] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: a survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.

[2] S. Feng, C. Wu, Y. Zhang, and G. Oliva, "WSN deployment and localization using a mobile agent," *Wireless Personal Communications*, vol. 97, no. 4, pp. 4921–4931, 2017.

[3] R. Herrero, "Analytical model of IoT CoAP traffic," *Digital Communications and Networks*, vol. 5, no. 2, pp. 63–68, 2019.

[4] P. Park, H. S. Ghadikolaei, and C. Fischione, "Proactive fault-tolerant wireless mesh networks for mission-critical control systems," *Journal of Network and Computer Applications*, vol. 186, article 103082, 2021.

[5] M. S. Haghighi, S. Wen, Y. Xiang, B. Quinn, and W. Zhou, "On the race of worms and patches: modeling the spread of information in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2854–2865, 2016.

[6] R. Shahzadi, M. Ali, H. Z. Khan, and M. Naeem, "UAV assisted 5G and beyond wireless networks: a survey," *Journal of Network and Computer Applications*, vol. 189, article 103114, 2021.

[7] M. B. Dowlatshahi, M. Kuchaki Rafsanjani, and B. B. Gupta, "An energy aware grouping memetic algorithm to schedule the sensing activity in WSNs-based IoT for smart cities," *Applied Soft Computing*, vol. 108, article 107473, 2021.

[8] I. Kitouni, D. Benmerzoug, and F. Lezzar, "Smart agricultural enterprise system based on integration of internet of things and agent technology," *Journal of Organizational and End User Computing*, vol. 4, no. 30, pp. 64–82, 2018.

[9] K. G. Srinivasa, B. J. Sowmya, A. Shikhar, R. Utkarsha, and A. Singh, "Data analytics assisted Internet of Things towards building intelligent healthcare monitoring systems," *Journal of Organizational and End User Computing*, vol. 3, no. 4, pp. 83–103, 2018.

[10] Z. W. A. S. Yu, "Binomial distribution based reputation for WSNs: a comprehensive survey," *KSII Transactions on Internet and Information Systems TIIS*, vol. 15, no. 10, pp. 3793–3814, 2021.

[11] L. Fan, E. Fan, C. Yuan, and K. Hu, "Weighted fuzzy track association method based on Dempster–Shafer theory in distributed sensor networks," *International Journal of Distributed Sensor Networks*, vol. 12, no. 7, Article ID 812018627, 2016.

[12] L. Fabisiak, "Web service usability analysis based on user preferences," *Journal of Organizational and End User Computing*, vol. 30, no. 4, pp. 1–13, 2018.

[13] S. Hosseini, M. A. Azgomi, and A. T. Rahmani, "Malware propagation modeling considering software diversity and immunization," *Journal of Computational Science*, vol. 13, pp. 49–67, 2016.

[14] Y. Wang, S. Wen, Y. Xiang, and W. Zhou, "Modeling the propagation of worms in networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 942–960, 2014.

[15] T. Wang, M. Z. A. Bhuiyan, G. Wang, L. Qi, J. Wu, and T. Hayajneh, "Preserving balance between privacy and data integrity in edge-assisted internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2679–2689, 2020.

[16] B. Asvija, R. Eswari, and M. B. Bijoy, "Security threat modelling with Bayesian networks and sensitivity analysis for IAAS virtualization stack," *Journal of Organizational and End User Computing*, vol. 33, no. 4, pp. 44–66, 2021.

[17] Q. Li, Q. Zhang, H. Huang, W. Zhang, W. Chen, and H. Wang, *Secure, Efficient and Weighted Access Control for Cloud-Assisted Industrial IoT*, IEEE Internet of Things Journal, Early Access, 2022.

[18] M. T. Amin, F. Khan, and S. Imtiaz, "Dynamic availability assessment of safety critical systems using a dynamic Bayesian network," *Reliability Engineering & System Safety*, vol. 178, pp. 108–117, 2018.

[19] D. K. Kotary, S. J. Nanda, and R. Gupta, "A many-objective whale optimization algorithm to perform robust distributed clustering in wireless sensor network," *Applied Soft Computing*, vol. 110, article 107650, 2021.

[20] V. P. Illiano and E. C. Lupu, "Detecting malicious data injections in wireless sensor networks," *ACM Computing Surveys*, vol. 48, no. 2, 2015.

[21] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of Things security: a survey," *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, 2017.

[22] A. S. Nandan, S. Singh, and L. K. Awasthi, "An efficient cluster head election based on optimized genetic algorithm for movable sinks in IoT enabled HWSNs," *Applied Soft Computing*, vol. 107, article 107318, 2021.

[23] D. Acarali, M. Rajarajan, N. Komninos, and B. B. Zarpelão, "Modelling the spread of botnet malware in IoT-based wireless sensor networks," *Security and Communication Networks*, vol. 2019, Article ID 3745619, 2019.

[24] A. Mahboubi, S. Camtepe, and H. Morarji, "A study on formal methods to generalize heterogeneous mobile malware propagation and their impacts," *IEEE Access*, vol. 5, pp. 27740–27756, 2017.

[25] S. Madadi, B. Mohammadi-Ivatloo, and S. Tohidi, "Probabilistic available transfer capability evaluation considering dynamic line rating based on a sequential game-theoretic approach," *IEEE Systems Journal*, vol. 16, 2021.

[26] M. Dibaei, X. Zheng, K. Jiang et al., "Attacks and defences on intelligent connected vehicles: a survey," *Digital Communications and Networks*, vol. 6, no. 4, pp. 399–421, 2020.

[27] Q. Jia, R. Xie, T. Huang, J. Liu, and Y. Liu, "Caching resource sharing for network slicing in 5G core network: a game theoretic approach," *Journal of Organizational and End User Computing*, vol. 31, no. 4, pp. 1–18, 2019.

[28] T. Li, H. Wang, D. He, and J. Yu, *Blockchain-Based Privacy-Preserving and Rewarding Private Data Sharing for IoT*, IEEE Internet of Things Journal, Early Access, 2022.

[29] S. Shen, K. Hu, L. Huang, H. Li, R. Han, and Q. Cao, "Quantal response equilibrium-based strategies for intrusion detection in WSNs," *Mobile Information Systems*, vol. 2015, Article ID 179839, 2015.

[30] L. Xiao, X. Lu, T. Xu, X. Wan, W. Ji, and Y. Zhang, "Reinforcement learning-based mobile offloading for edge computing against jamming and interference," *IEEE Transactions on Communications*, vol. 68, no. 10, pp. 6114–6126, 2020.

[31] J. Xu, "Digital community management mobile information system based on edge computing," *Mobile Information Systems*, vol. 2021, 11 pages, 2021.

[32] B. P. Rimal, D. P. Van, and M. Maier, "Mobile-edge computing empowered fiber-wireless access networks in the 5G era," *IEEE Communications Magazine*, vol. 11, no. 2, pp. 192–200, 2016.

[33] P. Corcoran and K. Soumya, "Mobile-edge computing and the Internet of Things for consumers: extending cloud computing and services to the edge of the network," *IEEE Consumer Electronics Magazine*, vol. 5, no. 4, pp. 73-74, 2016.

[34] S. Wen, W. Zhou, J. Zhang et al., "Modeling and analysis on the propagation dynamics of modern email malware," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 4, pp. 361–374, 2014.

[35] B. K. Mishra and N. Keshri, "Mathematical model on the transmission of worms in wireless sensor network," *Applied Mathematical Modelling*, vol. 37, no. 6, pp. 4103–4111, 2013.

[36] M. H. R. Khouzani, S. Sarkar, and E. Altman, "Saddle-point strategies in malware attack," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 1, pp. 31–43, 2012.

[37] S. Hosseini and M. A. Azgomi, "The dynamics of an SEIRS-QV malware propagation model in heterogeneous networks," *Physica A: Statistical Mechanics and its Applications*, vol. 512, pp. 803–817, 2018.

[38] K. C. Lalropuia and V. Gupta, "A Bayesian game model and network availability model for small cells under denial of service (DoS) attack in 5G wireless communication network," *Wireless Networks*, vol. 26, no. 1, pp. 557–572, 2020.

[39] W. Jiang, Z. Ma, and X. Deng, "An attack-defense game based reliability analysis approach for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 15, no. 4, Article ID 812336897, 2019.

[40] S. Shen, H. Ma, E. Fan et al., "A non-cooperative non-zero-sum game-based dependability assessment of heterogeneous WSNs with malware diffusion," *Journal of Network and Computer Applications*, vol. 91, pp. 26–35, 2017.

[41] S. Shen, H. Li, R. Han, A. V. Vasilakos, Y. Wang, and Q. Cao, "Differential game-based strategies for preventing malware propagation in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 11, pp. 1962–1973, 2014.

[42] J. Liu, S. Shen, G. Yue, R. Han, and H. Li, "A stochastic evolutionary coalition game model of secure and dependable virtual service in sensor-cloud," *Applied Soft Computing*, vol. 30, pp. 123–135, 2015.

[43] S. Shen, L. Huang, H. Zhou, S. Yu, E. Fan, and Q. Cao, "Multistage signaling game-based optimal detection strategies for suppressing malware diffusion in fog-cloud-based IoT networks," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1043–1054, 2018.

[44] J. Liu, X. Wang, S. Shen, G. Yue, S. Yu, and M. Li, "A Bayesian Q-learning game for dependable task offloading against DDoS attacks in sensor edge cloud," *IEEE Internet of Things Journal*, vol. 8, no. 9, pp. 7546–7561, 2020.

[45] J. Liu, X. Wang, S. Shen et al., "Intelligent jamming defense using DNN Stackelberg game in sensor edge cloud," *IEEE Internet of Things Journal*, vol. 9, 2021.

[46] S. Famila, A. Jawahar, S. Vimalraj, and J. Lydia, "Integrated energy and trust-based semi-Markov prediction for lifetime maximization in wireless sensor networks," *Wireless Personal Communications*, vol. 118, no. 1, pp. 505–522, 2021.

[47] D. V. Kanchana and R. Ganesan, "Semi Markov process inspired selfish aware co-operative scheme for wireless sensor networks (SMPISCS)," *Cybersecurity*, vol. 2, no. 1, 2019.

[48] R. K. Shakya, K. Rana, A. Gaurav, P. Mamoria, and P. K. Srivastava, "Stability analysis of epidemic modeling based on spatial correlation for wireless sensor networks," *Wireless Personal Communications*, vol. 108, no. 3, pp. 1363–1377, 2019.

[49] J. Tang, C. Ma, and P. Tian, "Network availability evaluation based on Markov chain of QoS-aware," *Wireless Personal Communications*, vol. 113, no. 4, pp. 1673–1689, 2020.

[50] R. Gour, G. Ishigaki, J. Kong, and J. P. Jue, "Availability-guaranteed slice composition for service function chains in 5G transport networks," *Journal of Optical Communications and Networking*, vol. 13, no. 3, pp. 14–24, 2021.

[51] P. Pereira, J. Araujo, C. Melo, V. Santos, and P. Maciel, "Analytical models for availability evaluation of edge and fog computing nodes," *Journal of Supercomputing*, vol. 77, pp. 9905–9933, 2021.

[52] A. Jafarabadi and M. A. Azgomi, "A stochastic epidemiological model for the propagation of active worms considering the dynamicity of network topology," *Peer-to-peer Networking and Applications*, vol. 8, no. 6, pp. 1008–1022, 2015.

[53] S. A. Sert and A. Yazici, "Increasing energy efficiency of rule-based fuzzy clustering algorithms using CLONALG-M for wireless sensor networks," *Applied Soft Computing*, vol. 109, article 107510, 2021.

[54] D. N. Kanellopoulos, "Recent progress on QoS scheduling for mobile ad hoc networks," *Journal of Organizational and End User Computing*, vol. 31, no. 3, pp. 37–66, 2019.

[55] B. Baranidharan and B. Santhi, "DUCF: distributed load balancing unequal clustering in wireless sensor networks using fuzzy approach," *Applied Soft Computing*, vol. 40, pp. 495–506, 2016.