sensors

MDPI

Article

Reliability Evaluation for Clustered WSNs under Malware Propagation

Shigen Shen [1,2,*], Longjun Huang [1], Jianhua Liu [2], Adam C. Champion [3], Shui Yu [4] and Qiying Cao [5]

1 Department of Computer Science and Engineering, Shaoxing University, Shaoxing 312000, China; hlj_jlh@163.com
2 College of Mathematics, Physics and Information Engineering, Jiaxing University, Jiaxing 314001, China; ljh_541@163.com
3 Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210, USA; champion@cse.ohio-state.edu
4 School of Information Technology, Deakin University, Burwood 3125, Australia; syu@deakin.edu.au
5 College of Computer Science and Technology, Donghua University, Shanghai 201620, China; caoqiying@dhu.edu.cn
* Correspondence: shigens@126.com; Tel.: +86-575-8834-2706

Academic Editor: Rongxing Lu
Received: 21 April 2016; Accepted: 6 June 2016; Published: 10 June 2016

Abstract: We consider a clustered wireless sensor network (WSN) under epidemic-malware propagation conditions and solve the problem of how to evaluate its reliability so as to ensure efficient, continuous, and dependable transmission of sensed data from sensor nodes to the sink. Facing the contradiction between malware intention and continuous-time Markov chain (CTMC) randomness, we introduce a strategic game that can predict malware infection in order to model a successful infection as a CTMC state transition. Next, we devise a novel measure to compute the Mean Time to Failure (MTTF) of a sensor node, which represents the reliability of a sensor node continuously performing tasks such as sensing, transmitting, and fusing data. Since clustered WSNs can be regarded as parallel-serial-parallel systems, the reliability of a clustered WSN can be evaluated via classical reliability theory. Numerical results show the influence of parameters such as the true positive rate and the false positive rate on a sensor node's MTTF. Furthermore, we validate the method of reliability evaluation for a clustered WSN according to the number of sensor nodes in a cluster, the number of clusters in a route, and the number of routes in the WSN.

Keywords: wireless sensor network; reliability evaluation; malware propagation; epidemic theory; continuous-time Markov chain; reliability theory

1. Introduction

Wireless Sensor Networks (WSNs) play an important role in daily life, as numerous modern information systems rely on WSNs, which consist of many sensor nodes with limited computation, storage, and communication resources. WSN applications include environmental, highway, and patient health monitoring as well as other commercial uses [1]. To realize these applications, the research community has focused on ensuring the reliability of WSNs. By definition, *reliability* reflects the ability of a system or component to perform its required functions under stated conditions for a specified period of time. Due to the nature of data collection in WSNs, this ability has been a major challenge in applying WSNs for successful monitoring. All sensor nodes need to send their sensed data towards the sink. Hence, packet loss due to transmission errors, packet collisions, interference, node failures, and malicious attacks is common [2]. Therefore, reliability evaluation for WSNs is vital in order to guarantee the delivery of sensed data from sensor nodes to the sink. In addition,

reliability evaluation is crucial for maintaining sensor nodes' functionality as nodes face attacks from self-replicating malicious code (*malware* for short).

In practice, WSNs are prone to malware propagation [3] for two reasons. First, sensor nodes have similar hardware and software, which results in large-scale malware propagation if even one sensor node is compromised. Second, there are several over-the-air reprogramming protocols (such as Trickle, Firecracker, Deluge, and MNP) that reconfigure sensor nodes without physical contact. Such protocols provide an opportunity for malware to spread across WSNs. Hence, WSNs' reliability needs to be evaluated under epidemic-malware propagation conditions. This article focuses on such evaluation.

Epidemic models are borrowed from epidemiology to describe malware propagation in WSNs, since there are strong similarities between biological viruses' self-replication and malware propagation. Classical epidemic models that have attracted much attention in the scientific community are classified among several families. The first family, the Susceptible-Infected (SI) model, is suitable for situations where nodes are either susceptible or infected. In this model, the state transition of any node is only from state *Susceptible* (*S*) to state *Infected* (*I*) and it is assumed that infected nodes remain in state *I* forever. The second family, the Susceptible-Infected-Susceptible (SIS) model, considers a susceptible node's infection with a certain probability when it comes into contact with another infected node. Unlike the SI model, an infected node can be removed with a different probability, becoming susceptible again. The third family, the Susceptible-Infected-Removed (SIR) model, adds the state *Removed* (*R*) by extending the SI model. In this model, a susceptible node can be infected only once, since a node transforming from state *I* to state *R* becomes immune and is hence unable to propagate malware to other susceptible nodes.

A continuous-time Markov chain (CTMC) provides considerable flexibility for modeling state transitions of sensor nodes, which is suitable for illustrating malware propagation. In general, when the CTMC is used, it includes a set of discrete states and is formally described by a state-transition-rate diagram. The diagram indicates possible states of a sensor node along with directed arcs that characterize transition rates between states. Since malware actions lead to state transitions in a CTMC, it shows the process of malware propagation.

However, the CTMC, which is only one type of stochastic model, is insufficient for correctly treating malware infections that often result in security failures for sensor nodes. Malware residing in an infected sensor node always intentionally propagates itself to other susceptible sensor nodes; thus, such infections cannot be modeled as a stochastic process. Even if the time to propagate malware may be randomly distributed, the decision to propagate is not. To solve this problem, we are motivated to leverage a strategic game to obtain the malware's expected propagation probability.

Game theory investigating strategic decision-making among players has been widely employed in the field of WSN security [4–12] such as optimizing intrusion detection strategies [5,6,10–12], securing data aggregation [7], localizing malicious nodes [8], and providing secure defenses for virtual sensor services [9]. This efficient mathematical tool is also suitable to explore different critical decisions during malware propagation. Usually, WSNs guard susceptible sensor nodes from malware attacks using Intrusion Detection Systems (IDSes). However, this method inevitably increases nodes' costs upon launching IDS agents, since sensor nodes have limited computational resources. Malware achieves greater gains by infecting more sensor nodes and eavesdropping on sensed data from infectious nodes, but this infection behavior obviously increases the probability of detection by IDSes. Thus, malware selects an optimal infection strategy to determine its propagation.

In this paper, we study reliability evaluation for clustered WSNs under epidemic-malware propagation. We use a strategic game to predict malware's infection behavior whose consequences we integrate into the transition probability upon infection of a sensor node. As a result, we determine how to relate the intent of malware infection to the CTMC's randomness. Next, we propose a novel measure to reflect the reliability of a sensor node. After considering clustered WSNs as parallel-serial-parallel systems due to their communication modes, we obtain equations to compute the reliability of a cluster, a route, and a clustered WSN, respectively.

Our main contributions are as follows:

(1) We relate the intent of malware infection to the CTMC's randomness by introducing a strategic game that can predict malware's infection behavior. In this manner, state transitions of a sensor node that arise from malware actions can be modeled by the CTMC; and
(2) We propose a novel measure to compute the Mean Time to Failure (MTTF) of a sensor node, which represents the reliability of a sensor node continuously performing tasks such as sensing, transmitting, and fusing data. Thus, we can deduce the reliability of a cluster, a route, and a clustered WSN from the perspective of a parallel-serial-parallel system, respectively. This method of reliability evaluation for clustered WSNs under epidemic-malware propagation can help establish theoretical foundations that guide rules for applying reliability techniques. Consequently, WSNs that guarantee reliable delivery of sensor nodes' sensed data may be realized.

The rest of this article is organized as follows: we first review related work and highlight the salient features of our approach in Section 2. We describe infections as state transitions from the view of a CTMC in Section 3. We obtain the infection probability by introducing a strategic malware-infection game in Section 4. We propose measures of reliability evaluation for clustered WSNs under the scenario of epidemic-malware propagation in Section 5. We validate the proposed measures' efficacy in Section 6. Finally, we conclude the article in Section 7.

2. Related Work

Based on classical epidemic models, many extended studies have been performed to describe the characteristics of malware propagation in WSNs. In a good survey, Yu *et al.* [13] presented current works on modeling malware propagation. Generally, sensor nodes periodically enter sleep mode to save energy. Typical models reflecting nodes sleeping during malware propagation include EiSIRS [14], a modified SI model [15], a modified SIS model [16], and Shen's model [17]. Moreover, sensor nodes "die" due to energy exhaustion or intentional destruction by malware. Thus, a dead state was introduced in iSIRS [18] based on the SIR model and Shen's model [17]. Furthermore, a reaction-diffusion-theoretic model [19], a pulse-differential-equation-based SIR model [20], and a susceptible-exposed-infected-recovered-susceptible model [21] were proposed in order to foresee spatial distribution and the temporal dynamics characteristic of malware propagation. In addition, Yu *et al.* [22] proposed a two-layer malware propagation model that better represents malware propagation in large-scale networks compared with existing single-layer epidemic models. Keshri and Mishra [23] presented a susceptible-exposed-infectious-recovered model with two time delays for charactering the transmission dynamics of malware propagation. Zhu and Zhao [24] explored a SIR-based nonlinear malware propagation model in WSNs. Wang *et al.* [25] presented a survey on modeling malware propagation in networks including WSNs. Other typical models [26–29] address the problem of malware propagation in multi-hop networks, which can help illustrate malware propagation in WSNs.

Several authors have considered decision-making dilemmas arising during malware propagation. Khouzani *et al.* [30] established a zero-sum dynamic game between the network system and the malware, given that malware can dynamically alter infection parameters based on the network system's dynamics. Jin *et al.* [31] used an evolutionary game to construct a malware propagation model under bounded rationality, where the game is to predict the trend of malware's evolutionary infection. Spyridopoulos *et al.* [32] employed a complete information game to obtain the defender's optimal strategy that minimizes the security cost as well as the malware effect. In addition, Trajanovski *et al.* [33] found decentralized optimal protection strategies for the network system by proposing a game-theoretic framework and seeking its Nash equilibria and the Price of Anarchy.

Several methods have been proposed using various techniques to cope with the challenge of evaluating WSNs' reliability. In a pioneering work [34], the authors employed a probabilistic graph to represent WSNs given a failure probability estimation of sensor nodes. Kar *et al.* [35] modeled

WSNs' energy reliability assuming Markovian sensor discharge/recharge periods. Distefano [36] used dynamic reliability block diagrams to represent static structural interactions between sensor nodes, where sleep/wake-up standby policies and interference are considered dynamically. Based on [36], he further gave a reliability evaluation model integrating Petri nets [37]. Silva *et al.* [38] proposed an evaluation methodology supporting arbitrary failure conditions based on automatically generated fault trees. Niyato *et al.* [39] proposed reliability analysis of wireless communications systems in the smart grid, which is also suitable for WSNs. Kamal *et al.* [40] developed a novel framework called Packet-Level Attestation for sensor data reliability evaluation using the spatial relationship among data sensed by nearby sensor nodes. In [41] Dâmaso *et al.* proposed a reliability evaluation model based on routing algorithms in WSNs and sensor nodes' various battery levels. According to MAC protocols adopted in WSNs, Wang *et al.* [42] evaluated a sensor node's reliability under three typical working scenarios including sensor nodes always in active mode, alternating between sleep and active modes on average, and alternating between these modes based on a certain distribution. Zonouz *et al.* [43] evaluated the reliability of energy harvesting sensor nodes and battery-powered sensor nodes as well as develop the corresponding wireless-link-reliability models. Cai *et al.* [44] characterized event-driven WSNs according to limited node battery energy and shadowing under channel fading, obtaining reliable data flows in WSNs via wireless link reliability and node energy availability. Wang *et al.* [45] analyzed a body sensor node's reliability subject to probabilistic competition between propagation effects and probabilistic failure isolation. Yan *et al.* [46] proposed a symbolic ordered-binary-decision-diagram-multicast method to evaluate the reliability of multicast WSNs. Zhu *et al.* [47] proposed mission-oriented and transmission-paths-based models for evaluating WSNs' transmission reliability. Other measures closely related to reliability include dependability and survivability such as the stochastic-activity-network-based dependability measure [48], the epidemic-theory-based survivability measure [49,50], survivability analysis using probabilistic model checking [51], and natural-tenacity-based survivability evaluation for mobile WSNs [52].

Unlike this body of work, to the best of our knowledge this work is the first that concentrates on reliability evaluation for clustered WSNs under epidemic-malware propagation conditions. While our epidemic model is similar to [14,17], we further model all state transitions of a sensor node as a CTMC by integrating the malware's infection probability predicted by our strategic malware-infection game into the transition probability. We apply the approach proposed in [53] to compute the MTTF from a Markov process; however, we further find the equation to compute the reliability of a sensor node under epidemic-malware propagation, which is a novel measure. Considering the topology of a typical clustered WSN, we can thus deduce the reliability of a cluster, a route, and a clustered WSN based on reliability theory. In summary, we propose a unified framework for malware-infection and reliability evaluation that integrates both security and reliability properties of a clustered WSN during the evaluation process.

3. Modeling State Transitions of a Sensor Node as a CTMC

The various epidemic-malware propagation models mentioned above are actually state transition models. These states are mutually exclusive: a sensor node is in exactly one state at any time. During its lifecycle, a sensor node interchanges among different states. Figure 1 illustrates a CTMC indicating state transitions of a sensor node, where p_{ij}, i, $j \in \{S, \widehat{S}, I, \widehat{I}, R, \widehat{R}, D\}$, denotes the transition rate from state i to state j, and S, $\widehat{S}$, I, $\widehat{I}$, R, $\widehat{R}$, and D denote states *Susceptible*, *Susceptible* while *sleeping*, *Infected*, *Infected* while *sleeping*, *Recovered*, *Recovered* while *sleeping*, and *Dead*, respectively. Each sensor node's characteristics determine its state. State S denotes a sensor node that works normally and is not infected by malware, but it is susceptible to malware. $\widehat{S}$ denotes a sleeping sensor node that malware cannot infect although the node is susceptible. I denotes a sensor node that has been infected by malware and may propagate malware to neighboring nodes with which it communicates as it is under

the malware's control. $\widehat{I}$ denotes an infected sensor node that is sleeping; thus, it cannot propagate malware. R denotes a "recovered" node that can be "immunized" to current malware, but not to future malware. $\widehat{R}$ denotes a recovered node that is sleeping. D denotes a node that is unusable because malware exhausted its energy, deliberately destroyed it, or both.

In practice, behaviors of both sensor nodes and malware trigger each state transition. For each sensor node, installing security patches is a normal method to either prevent susceptible nodes from known malware by fixing bugs or to remedy an infected node and immunize it to known malware. Thus, this action results in the state transitions $S \rightarrow R$ and $I \rightarrow R$. In general, each node is scheduled to sleep, which saves energy, or to awaken, which leads to the state transitions $S \rightarrow \widehat{S}$ or $\widehat{S} \rightarrow S$. The state transitions $I \rightarrow \widehat{I}$ or $\widehat{I} \rightarrow I$ as well as $R \rightarrow \widehat{R}$ or $\widehat{R} \rightarrow R$ follow likewise. When confronting unknown malware, a sensor node usually lacks immunity, and the state transition $R \rightarrow S$ follows. In addition, the state transitions $S \rightarrow D$, $\widehat{S} \rightarrow D$, $I \rightarrow D$, $\widehat{I} \rightarrow D$, $R \rightarrow D$, and $\widehat{R} \rightarrow D$ take place when malware exhausts a sensor node's energy. On the other hand, contamination by malware leads to the state transition $S \rightarrow I$. Moreover, malware can deliberately destroy an infectious sensor node besides the case of death from energy exhaustion; hence, the state transition $I \rightarrow D$ occurs.

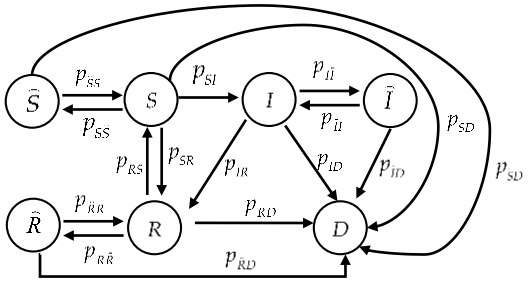

Figure 1. A CTMC that illustrates the state transition of a sensor node.

However, the state transition $S \rightarrow I$ cannot be formulated as a stochastic process because the infection behavior resulting in this transition is deliberate. Even though the time to execute action *Infect* is stochastically distributed, the decision to execute the actual infection is not. Since malware decides whether to infect a susceptible sensor node, there is an infection probability that malware determines to execute infection. We let ρ_I be the probability that the malware will choose action *Infect* to propagate itself and λ be the probability of a successful infection in order to formalize the malware's decision. Thus, the infection rate (*i.e.*, the state transition rate) from state S to state I becomes:

$$p_{SI} = \rho_I \lambda \tag{1}$$

By introducing the infection probability ρ_I, we can model the consequence of a successful infection as one deliberate state transition of the CTMC, which describes the dynamics of a sensor node's behavior. In this manner, our modeling approach concentrates on the higher-level effects of infection on a sensor node rather than the lower-level specific infection procedures.

4. A Strategic Malware-Infection Game to Obtain the Infection Probability

We employ a strategic game to predict the malware's infection behavior, namely obtaining the probability ρ_I. We adopt game-theoretic analysis as the following dilemma arises: the malware attempts to infect as many sensor nodes as possible without detection, whereas the defender attempts to enhance the network robustness by detecting more malware. Facing this dilemma, we introduce players *malware* and *system* to play the strategic game. Even though there are various kinds of malware whose goals may be to eavesdrop on private sensed data or to disable communication among sensor

nodes, it is sufficient to regard all malware as player *malware* due to malware's similar motivations and skills. Player *system*, the opponent of *malware*, actually corresponds to IDSes residing in WSNs.

In practice, iterations of the strategic game can be depicted as follows. There are discrete periods of time in which player *malware* launches infection. Player *system* intends to prevent infection in any of these periods. In each period, each player has two actions: player *malware* can either infect or not infect while player *system* can either defend or not defend. But each player can choose only one action with either pure or mixed strategies. If player *malware* takes no action and player *system* does not defend, then the game enters the next stage. Next, we formally define our strategic game and we explore the game entirely from *malware*'s view, since the target of our strategic game is to predict the infection intention of player *malware* and not to obtain the optimal defense strategies for player *system*.

Definition 1. *The Strategic Malware-Infection Game (SMIG) is formulated by a 4-tuple* $\mathbb{G} = (\mathcal{N}, \mathcal{A}_M, \mathcal{A}_S, \mathcal{U})$, *where:*

- $\mathcal{N} = \{malware,\ system\}$ *is a set of players;*
- $\mathcal{A}_M = \{Infect\ (I),\ Non-infect\ (\phi)\}$ *is a set of actions performed by player* malware*;*
- $\mathcal{A}_S = \{Defend\ (D),\ Non-defend\ (\phi)\}$ *is a set of actions performed by player* system*;*
- $\mathcal{U} : \mathcal{A}_M \times \mathcal{A}_S \mapsto \mathbb{R}$ *is a payoff matrix.*

Let ρ_I and ρ_ϕ be the probabilities that player *malware* adopts actions *Infect* and *Non-infect*, respectively. Let δ_D and δ_ϕ be the probabilities that player *system* adopts actions *Defend* and *Non-defend*, respectively. Accordingly, infection strategy ρ and defense strategy δ are mixed strategies (ρ_I, ρ_ϕ) and (δ_D, δ_ϕ), which represent the probability distributions over action sets $\mathcal{A}_M$ and $\mathcal{A}_S$, respectively. Both of them certainly satisfy $\rho_I + \rho_\phi = 1$ and $\delta_D + \delta_\phi = 1$. Actually, the infection probability ρ_I describes the degree of infection for a sensor node (equivalently, the aggressiveness of player *malware* targeting a sensor node). The larger ρ_I is, the greater the probability of action *Infect* and, hence, the larger the corresponding infection rate for a sensor node.

Now we consider the payoff matrix to explore *malware*'s motivation. For simplicity, we denote the worth of a sensor node by ω, where $\omega > 0$. Actually, ω is equivalent to a degree of damage such as the loss of sensed data, loss due to compromise, and so on. If player *malware* infects successfully, it will obtain payoff ω and its opponent will obtain payoff $-\omega$. On the contrary, if player *system* succeeds in defense, its payoff is ω because it has protected a sensor node worth ω and player *malware* will be penalized by ω. However, no IDS can entirely detect all current and future malware: all IDSes have true positive rates and false positive rates. Next, we consider these two rates as defining the payoffs of *malware* and *system* and we let α and β be the true positive rate and the false positive rate of the WSN IDS, respectively. We also let c_I and c_D be the cost of player *malware* infecting a susceptible sensor node and player *system* detecting the *malware*'s infection, respectively. Obviously, there are four possible payoffs constructing the payoff matrix, since each of *malware* and *system* has two possible actions. For the action profile (*Infect*, *Defend*), player *malware* will obtain gain $(1-\alpha)\omega$ from detection failure as well as loss $\alpha\omega$ from being detected successfully and loss c_I from adopting action *Infect*. Thus, the payoff of player *malware*, u^I_{ID}, is:

$$u^I_{ID} = (1-\alpha)\omega - \alpha\omega - c_I = (1-2\alpha)\omega - c_I \tag{2}$$

On the other hand, player *system* obtains gain $\alpha\omega$ as well as loses $(1-\alpha)\omega$ from detection failure, $\beta\omega$ from false positive detection, and c_D from detecting *malware*'s infection. Thus, the payoff of player *system*, u^S_{ID}, is:

$$u^S_{ID} = \alpha\omega - (1-\alpha)\omega - \beta\omega - c_D = (2\alpha - 1 - \beta)\omega - c_D \tag{3}$$

For the action profile (*Infect*, *Non-defend*), player *malware* obtains gain $\lambda\omega$ from successful infection and loses c_I from adopting action *Infect*. Thus, the payoff of player *malware*, $u^I_{I\phi}$, is:

$$u_{I\phi}^{I} = \lambda\omega - c_I \tag{4}$$

whereas the payoff of player *system*, $u_{I\phi}^{S}$, is:

$$u_{I\phi}^{S} = -\lambda\omega \tag{5}$$

For the action profile (*Non-infect*, *Defend*), the payoff of player *malware*, $u_{\phi D}^{I}$, is:

$$u_{\phi D}^{I} = 0 \tag{6}$$

whereas the payoff of player *system*, $u_{\phi D}^{S}$, is:

$$u_{\phi D}^{S} = -\beta\omega - c_D \tag{7}$$

Finally, for the action profile (*Non-infect*, *Non-defend*), since neither player *malware* nor player *system* can obtain any gain or produce any loss, the payoff of player *malware*, $u_{\phi\phi}^{I}$, is:

$$u_{\phi\phi}^{I} = 0 \tag{8}$$

and the payoff of player *system*, $u_{\phi\phi}^{S}$, is:

$$u_{\phi\phi}^{S} = 0 \tag{9}$$

Table 1 summizes our defined payoff matrix.

Table 1. The payoff matrix of the SMIG.

	Defend	***Non-Defend***
Infect	$(1-2\alpha)\omega - c_I, (2\alpha-1-\beta)\omega - c_D$	$\lambda\omega - c_I, -\lambda\omega$
Non-infect	$0, -\beta\omega - c_D$	$0, 0$

The objective of player *malware* is to maximize its expected infection utility, whereas the objective of player *system* is to minimize its expected defense utility. This objective can be achieved by solving the mixed-strategy Nash Equilibrium (NE) of the strategic game.

Theorem 1. *In the SMIG, the optimal probability of player malware choosing action Infect is:*

$$\rho_I^* = \frac{\beta\omega + c_D}{(\lambda + 2\alpha - 1)\omega} \tag{10}$$

Proof: Under player *malware*'s mixed strategy, player *system*'s expected payoffs for choosing actions *Defend* and *Non-defend* are:

$$E_S(Defend) = \rho_I((2\alpha - 1 - \beta)\omega - c_D) + (1 - \rho_I)(-\beta\omega - c_D) \tag{11}$$

and:

$$E_S(Non - defend) = \rho_I(-\lambda\omega) + (1 - \rho_I) \cdot 0 = -\rho_I\lambda\omega \tag{12}$$

respectively. From the indifference between actions *Defend* and *Non-defend* under the optimal mixed strategy of player *system*, we obtain:

$$E_S(Defend) = E_S(Non - defend) \tag{13}$$

Therefore, the optimal probability of player *malware* choosing action *Infect* is:

$$\rho_I^* = \frac{\beta\omega + c_D}{(\lambda + 2\alpha - 1)\omega}$$

□

Obtaining the optimal infection probability ρ_I^* means that we acquire the indication of the expected infection behavior of player *malware* for WSNs under epidemic-malware propagation. In other words, malware will choose action *Infect* with probability ρ_I^*. When following ρ_I^*, player *malware* has no reason to adjust its strategy as it has maximized its expected utility from the infection regardless of the success of its actions.

5. Reliability Evaluation Method

5.1. Evaluating the Reliability of a Sensor Node

The reliability of WSNs represents the probability that sensor nodes continue to perform tasks such as data sensing, transmission, and fusion over a particular period of time under stated conditions. In general, MTTF and MTBF (Mean Time between Failures) are typical ways to evaluate the reliability of pieces of hardware or other technology. Here, MTTF refers to the length of time that a device is expected to last in operation, whereas MTBF is the average elapsed time between a device's failures in operation. The difference between two terms is that MTTF is used for non-repairable devices, whereas MTBF is used for devices that can be repaired and returned to operation. In this work, we concentrate on WSNs where sensor nodes can hardly be repaired upon failure and we use MTTF for evaluating the reliability of a sensor node.

We denote $\mathcal{E}$ as the discrete state space where:

$$\mathcal{E} = \{S,\ \widehat{S},\ I,\ \widehat{I},\ R,\ \widehat{R},\ D\} \tag{14}$$

as illustrated in Figure 1. Let:

$$\mathbf{Y}(t) = [Y_S(t)\ \ Y_{\widehat{S}}(t)\ \ Y_I(t)\ \ Y_{\widehat{I}}(t)\ \ Y_R(t)\ \ Y_{\widehat{R}}(t)\ \ Y_D(t)] \tag{15}$$

be the state probability vector, where $Y_x(t)$ denotes the probability that a sensor node is in state x, $x \in \mathcal{E}$, at time t. Let $\mathbf{P}$ be the 7×7 state transition matrix consisting of element p_{ij}, $i,\ j \in \mathcal{E}$. We find the state equation of a sensor node as:

$$\frac{d\mathbf{Y}(t)}{dt} = \mathbf{Y}(t)\mathbf{P} \tag{16}$$

We find the steady-state probability vector that is independent of $\mathbf{Y}(0)$ (*i.e.*, the initial state):

$$\mathbf{Y} = [Y_S\ \ Y_{\widehat{S}}\ \ Y_I\ \ Y_{\widehat{I}}\ \ Y_R\ \ Y_{\widehat{R}}\ \ Y_D] \tag{17}$$

by solving the system of seven equations where six of the seven equations are from:

$$\mathbf{YP} = 0 \tag{18}$$

and the seventh equation is:

$$\sum_{i \in \mathcal{E}} Y_i = 1 \tag{19}$$

Next, we compute a sensor node's MTTF from the steady-state probability vector. The discrete state space $\mathcal{E}$ can be split into two disjoint sets $\mathcal{E}_{Use}$ and $\mathcal{E}_{Disuse}$, where:

$$\mathcal{E}_{Use} = \{S,\ R\} \tag{20}$$

and:

$$\mathcal{E}_{Disuse} = \{\widehat{S},\ I,\ \widehat{I},\ \widehat{R},\ D\} \tag{21}$$

denote the set of usable and unusable states, respectively. Correspondingly, the state transition matrix $\mathbf{P}$ can be rewritten as:

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_1 & \mathbf{P}_2 \\ \mathbf{P}_3 & \mathbf{P}_4 \end{bmatrix} \tag{22}$$

where:

$$\mathbf{P}_1 = \begin{bmatrix} p_{SS} & p_{SR} \\ p_{RS} & p_{RR} \end{bmatrix} \tag{23}$$

$$\mathbf{P}_2 = \begin{bmatrix} p_{S\widehat{S}} & p_{SI} & p_{S\widehat{I}} & p_{S\widehat{R}} & p_{SD} \\ p_{R\widehat{S}} & p_{RI} & p_{R\widehat{I}} & p_{R\widehat{R}} & p_{RD} \end{bmatrix} \tag{24}$$

$$\mathbf{P}_3 = \begin{bmatrix} p_{\widehat{S}S} & p_{\widehat{S}R} \\ p_{IS} & p_{IR} \\ p_{\widehat{I}S} & p_{\widehat{I}R} \\ p_{\widehat{R}S} & p_{\widehat{R}R} \\ p_{DS} & p_{DR} \end{bmatrix} \tag{25}$$

and:

$$\mathbf{P}_4 = \begin{bmatrix} p_{\widehat{S}\widehat{S}} & p_{\widehat{S}I} & p_{\widehat{S}\widehat{I}} & p_{\widehat{S}\widehat{R}} & p_{\widehat{S}D} \\ p_{I\widehat{S}} & p_{II} & p_{I\widehat{I}} & p_{I\widehat{R}} & p_{ID} \\ p_{\widehat{I}\widehat{S}} & p_{\widehat{I}I} & p_{\widehat{I}\widehat{I}} & p_{\widehat{I}\widehat{R}} & p_{\widehat{I}D} \\ p_{\widehat{R}\widehat{S}} & p_{\widehat{R}I} & p_{\widehat{R}\widehat{I}} & p_{\widehat{R}\widehat{R}} & p_{\widehat{R}D} \\ p_{D\widehat{S}} & p_{DI} & p_{D\widehat{I}} & p_{D\widehat{R}} & p_{DD} \end{bmatrix} \tag{26}$$

Likewise, we split the steady-state probability vector into two disjoint parts $\mathbf{Y}_{Use}$ and $\mathbf{Y}_{Disuse}$, where:

$$\mathbf{Y}_{Use} = [Y_S\ \ Y_R] \tag{27}$$

and:

$$\mathbf{Y}_{Disuse} = [Y_{\widehat{S}}\ \ Y_I\ \ Y_{\widehat{I}}\ \ Y_{\widehat{R}}\ \ Y_D] \tag{28}$$

According to the method provided in [53] to compute the MTTF from a Markov process, we find the MTTF of a sensor node η as:

$$\eta = \mathbf{Y}_{Use}(0)(-\mathbf{P_1})^{-1}\mathbf{I} \tag{29}$$

where $\mathbf{Y}_{Use}(0)$ denotes the initial usable state probability vector (*i.e.*, with $t = 0$) computed as:

$$\mathbf{Y}_{Use}(0) = \frac{\mathbf{Y}_{Use}}{\mathbf{Y}_{Use}\mathbf{I}} \tag{30}$$

and $\mathbf{I}$ denotes a column vector of two ones:

$$\mathbf{I} = [1\ \ 1]^{-1} \tag{31}$$

Let $Reliability_{node}(t)$ be the reliability of a sensor node at time t. We assume that all sensor nodes, due to their similarity, have the same MTTF. From reliability theory, we find that:

$$Reliability_{node}(t) = \exp(-\frac{1}{\eta}t) \tag{32}$$

5.2. Evaluating Reliability of a Clustered WSN

We aim to perform reliability evaluations for clustered WSNs due to their popularity. Figure 2 illustrates the topology of a clustered WSN. Coordinating cluster heads (CHs) control the topology where each CH guides a different cluster of sensor nodes. Such an architecture leads to a two-tier hierarchy where the upper tier comprises CHs and the lower tier comprises sensor nodes. Sensor nodes in specific regions transmit their sensed data to the responsible CH that manages the nodes. The responsible CH collects the data, which are transmitted to the single base station via other CHs. In this way, we can relate each cluster to a parallel system and each set of clusters to a serial system. Therefore, any route from a sensor node to the base station can be naturally regarded as a serial-parallel path. Since there are various routes through which sensed data can be transferred, a clustered WSN can be mapped correspondingly to a parallel-serial-parallel system where each sensor node fails independently. Each cluster operates if at least one of its sensor nodes works normally. However, for any route, all of its clusters must work normally for proper operation. As a result, we can evaluate the reliability of a clustered WSN based on the MTTF of a sensor node from the perspective of classical reliability theory.

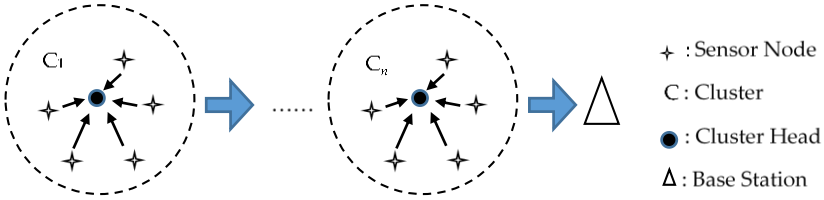

Figure 2. Topology of a clustered WSN, where n denotes the number of clusters in a route.

Since each cluster is a parallel system, the reliability of a cluster at time t, $Reliability_{cluster}(t)$, can be computed as:

$$Reliability_{cluster}(t) = 1 - \prod_{node \in cluster} (1 - Reliability_{node}(t)) \tag{33}$$

Moreover, any route composed of clusters is a serial system, and thus the reliability of a route at time t, $Reliability_{route}(t)$, can be computed as:

$$Reliability_{route}(t) = \prod_{cluster \in route} Reliability_{cluster}(t) \tag{34}$$

Finally, a clustered WSN composed of available routes is a parallel system. Therefore, the reliability of a clustered WSN at time t, $Reliability(t)$, is:

$$Reliability(t) = 1 - \prod_{route} (1 - Reliability_{route}(t)) \tag{35}$$

So far, we have developed a method of reliability evaluation for clustered WSNs under epidemic-malware propagation conditions. In practice, we suggest using the proposed method in off-line mode, since performing on-line reliability evaluation for clustered WSNs is very difficult. First, we define our strategic game mathematically and solve it analytically. We integrate the results of the game with the transition probability upon infection of a sensor node from which we compute the MTTF of a sensor node. As soon as the topology of a clustered WSN is determined based on the actual requirements, we can find the number of sensor nodes in a cluster, the number of clusters in a route, and the number of routes in a clustered WSN. Based on these numbers, we can compute the reliability of a clustered WSN from Equation (35). In fact, our way is popular in the field of using the game theoretical approaches, which is easy to realize.

6. Numerical Results

6.1. Illustrating Influence of α and β

With MATLAB R2010b, we explore how the optimal infection probability and the MTTF of a sensor node depend on the parameters of the true positive rate (*i.e.*, the detection rate) and the false positive rate (*i.e.*, the false alarm rate). The parameters of the strategic game are $\omega = 50$, $c_I = 20$, $c_D = 10$, and $\lambda = 0.5$. Note that we can attain similar trends if the parameters are changed. However, specific values will be correspondingly changed.

Figures 3 and 4 demonstrate the changing optimal infection probabilities that player *malware* adopts according to α and β, which reveals the *malware*'s intention. Obviously, a higher detection rate and a lower false-alarm rate can help a WSN IDS detect malware. Therefore, player *malware* will choose its optimal strategy to lower the infection probability in order to minimize the loss arising from IDS detection. As shown in Figure 3, the optimal infection probability decreases gradually when the true positive rate increases slowly from 0.7 to 0.98. Moreover, a lower false positive rate results in a lower infection probability. For example, when $\alpha = 0.88$ in Figure 3, the optimal infection probabilities are ~0.1667, ~0.1984, and ~0.2381 for $\beta = 0.01$, $\beta = 0.05$, and $\beta = 0.1$, respectively. We observe in Figure 4 the optimal infection probability increases gradually when the false positive rate increases slowly from 0.01 to 0.15. Furthermore, a higher detection rate leads to a lower infection probability. For example, when $\beta = 0.1$ in Figure 4, the optimal infection probabilities are ~0.2727, ~0.2308, and ~0.2055 for $\alpha = 0.8$, $\alpha = 0.9$, and $\alpha = 0.98$, respectively. These experimental results indicate that the true positive rate should be increased and the false positive rate should be decreased in order to decrease the infection probability adopted by player *malware* and to enhance a sensor node's reliability.

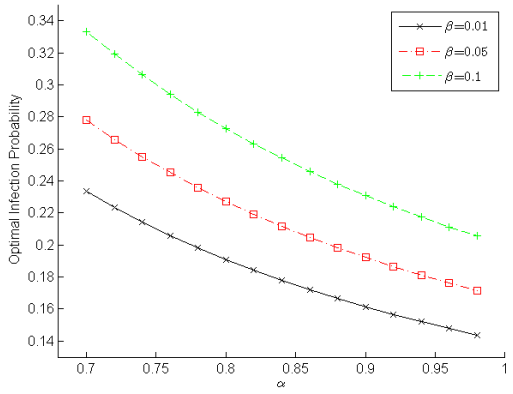

Figure 3. Optimal infection probability in terms of the true positive rate.

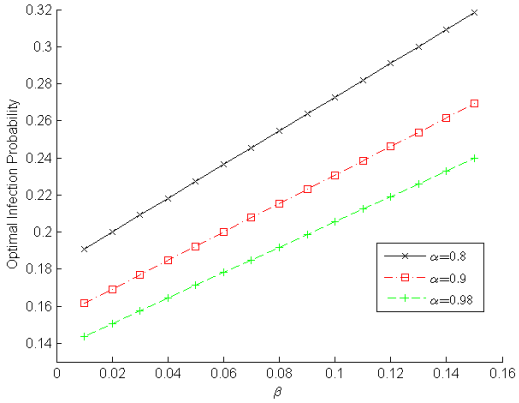

Figure 4. Optimal infection probability in terms of the false positive rate.

Figure 5 shows the MTTF of a sensor node under the epidemic-malware propagation scenario in terms of α and β. As the true positive rate increases and the false positive rate decreases, it is easy for player *system* to detect infected sensor nodes, increasing their MTTFs. From Figure 5, as we expect, the MTTF increases slowly when the true positive rate increases gradually from 70% to 98%. There is a similar tendency when the false positive rate decreases from 15% to 1%. Smaller decreases of the false positive rate increase the MTTF for a sensor node more than the same decreases of the true positive rate. For example, the MTTF of a sensor node increases from ~4.6366 to ~5.7219 (an increase of ~23.41%) as β drops from 15% to 1% when $\alpha = 88\%$. However, when $\beta = 10\%$, the MTTF of a sensor node increases from ~4.2487 to ~5.0350 (an increase of ~18.51%) as α increases from 70% to 90%. These results indicate that we should further reduce the false positive rate while improving IDSes for WSNs in order to increase a sensor node's MTTF.

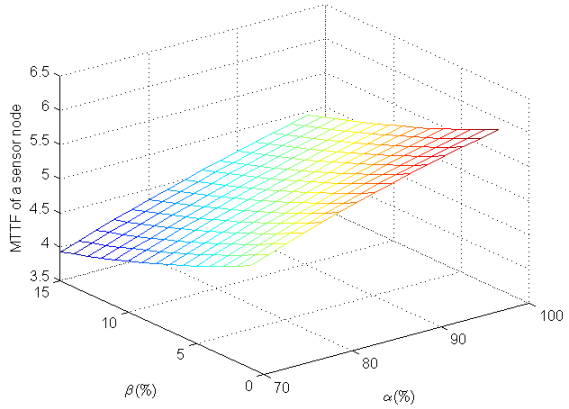

Figure 5. A sensor node's MTTF in terms of the true positive rate and the false positive rate.

6.2. Validating the Method of Reliability Evaluation for a Clustered WSN

Next, from the perspective of a clustered WSN, we evaluate its reliability according to the number of sensor nodes in a cluster, the number of clusters in a route, and the number of routes in a clustered WSN, as illustrated in Figures 6–8, respectively.

Figure 6 illustrates varying degrees of reliability for a clustered WSN when there are two, four, and six sensor nodes in a cluster, respectively. When both the number of clusters in a route and the number of routes in a clustered WSN are static, the reliability of the WSN increases with the number of sensor nodes in a cluster. With two, four, and six nodes in a cluster, it takes about six, nine, and eleven days, respectively, to reduce the reliability of a clustered WSN to 0.5 under the epidemic-malware propagation scenario.

Figure 7 illustrates varying degrees of reliability for a clustered WSN when there are two, four, and six clusters in a route, respectively. When both the number of sensor nodes in a cluster and the number of routes in a clustered WSN are static, the reliability of the WSN decreases with the number of clusters in a route. With two, four, and six clusters in a route, it takes about eleven, eight, and seven days, respectively, to reduce the reliability of a clustered WSN to 0.5 under the epidemic-malware propagation scenario.

Figure 8 illustrates varying degrees of reliability for a clustered WSN when there are two, four, and six routes in a clustered WSN, respectively. When both the number of sensor nodes in a cluster and the number of clusters in a route are static, the reliability of the WSN increases with the number of routes in the clustered WSN. With two, four, and six routes in the clustered WSN, it takes about seven, eight, and nine days, respectively, to reduce the reliability of the clustered WSN to 0.5 under the epidemic-malware propagation scenario.

In summary, the experimental results shown in Figures 6–8 indicate that deploying more redundant sensor nodes in a cluster, deducing the clusters along constructed routes, and providing

more available routes all help improve the reliability of a clustered WSN, which accords with our expectations.

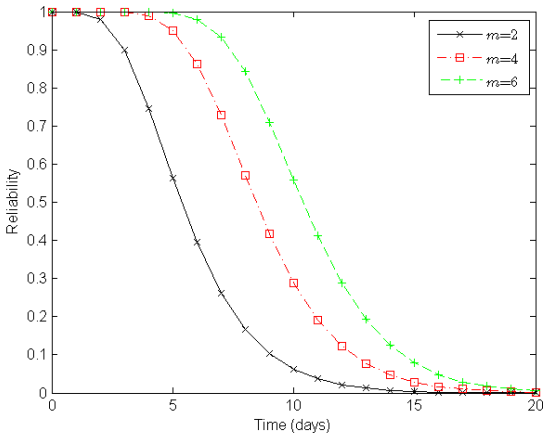

Figure 6. Reliability of a clustered WSN when $m = 2$, $m = 4$, and $m = 6$, respectively. Here, m denotes the number of sensor nodes in a cluster. There are four clusters in a route and four routes in a clustered WSN.

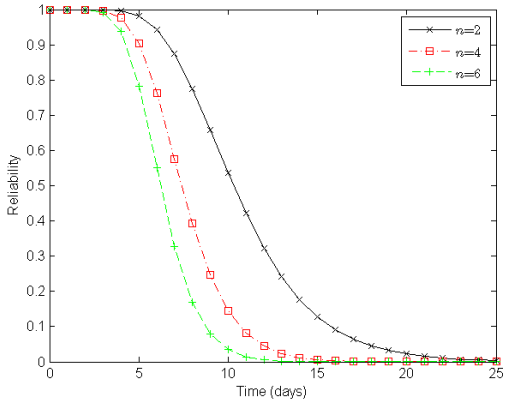

Figure 7. Reliability of a clustered WSN when $n = 2$, $n = 4$, and $n = 6$, respectively. Here, n denotes the number of clusters in a route. There are four sensor nodes in a cluster and four routes in a clustered WSN.

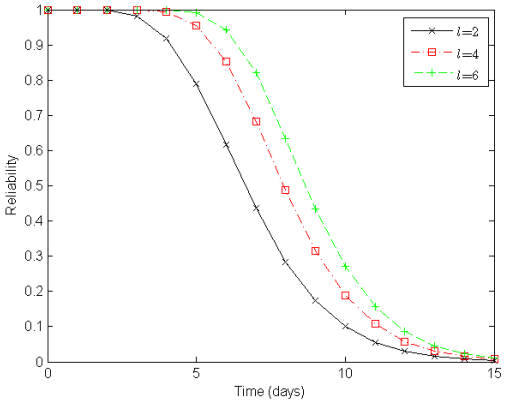

Figure 8. Reliability of a clustered WSN when $l = 2$, $l = 4$, and $l = 6$, respectively. Here, l denotes the number of routes in the clustered WSN. There are four sensor nodes in a cluster and four clusters in a route.

7. Conclusions

We have performed reliability analysis on clustered WSNs under the epidemic-malware propagation scenario and developed a corresponding measure of reliability evaluation in order to establish a kind of highly reliable WSN. We have determined how to relate the intent of malware infection to the randomness of CTMCs using a strategic game that can predict malware's infection behavior. We have proposed the MTTF to reflect the reliability of a sensor node and we have regarded clustered WSNs as a parallel-serial-parallel system. Using this approach, we have obtained equations to compute the reliability of a cluster, a route, and a clustered WSN, respectively. As a result, we have provided a foundation for the mechanism of reliability evaluation for susceptible WSNs. Our experiments have shown the importance of reducing the false positive rate rather than the true positive rate in order to increase MTTFs for susceptible sensor nodes. We have also validated the efficacy of our proposed measure of reliability evaluation for susceptible WSNs.

We have assumed that the topology of a clustered WSN only consists of a single sink node; however, actual clustered WSNs may have several sink nodes. Furthermore, the topology of a clustered WSN may be changed once mobile sensor nodes are introduced. Under these circumstances, the equation to compute the reliability of the clustered WSN will be more complicated. Providing such a reliability evaluation method is an interesting research direction when the assumption is relaxed. Moreover, providing measures of availability, dependability, and survivability for WSNs under malware propagation is another interesting direction.

Acknowledgments: This work was supported by National Natural Science Foundation of China under Grants Nos. 61272034 and 61572014 and by Zhejiang Provincial Natural Science Foundation of China under Grant No. LY16F020028.

Author Contributions: S.S, L.H., and J.L. constructed the CTMC, formulated the game, proved the theorem, and deduced the reliability evaluation. S.S also drafted the initial version of the manuscript. L.H. also performed the experiments. A.C.C. polished the English grammar. Y.S. revised and improved the paper. Q.C. gave the initial idea and supervised the development of the work.

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

References

1. Borges, L.M.; Velez, F.J.; Lebres, A.S. Survey on the characterization and classification of wireless sensor network applications. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1860–1890. [CrossRef]
2. Mahmood, M.A.; Seah, W.K.G.; Welch, I. Reliability in wireless sensor networks: A survey and challenges ahead. *Comput. Netw.* **2015**, *79*, 166–187. [CrossRef]
3. Illiano, V.P.; Lupu, E.C. Detecting malicious data injections in wireless sensor networks: A survey. *ACM Comput. Surv.* **2015**, *48*, 24. [CrossRef]
4. Manshaei, M.H.; Zhu, Q.; Alpcan, T.; Başar, T.; Hubaux, J. Game theory meets network security and privacy. *ACM Comput. Surv.* **2013**, *45*, 25. [CrossRef]
5. Moosavi, H.; Bui, F.M. A game-theoretic framework for robust optimal intrusion detection in wireless sensor networks. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 1367–1379. [CrossRef]
6. Shamshirband, S.; Patel, A.; Anuar, N.B.; Kiah, M.L.M.; Abraham, A. Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks. *Eng. Appl. Artif. Int.* **2014**, *32*, 228–241. [CrossRef]
7. Engouang, T.D.; Liu, Y.; Zhang, Z. GABs: A game-based secure and energy efficient data aggregation for wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2015**, *2015*. [CrossRef]
8. Basilico, N.; Gatti, N.; Monga, M.; Sicari, S. Security games for node localization through verifiable multilateration. *IEEE Trans. Depend. Secur.* **2014**, *11*, 72–85. [CrossRef]
9. Liu, J.; Shen, S.; Yue, G.; Han, R.; Li, H. A stochastic evolutionary coalition game model of secure and dependable virtual service in sensor-cloud. *Appl. Soft Comput.* **2015**, *30*, 123–135. [CrossRef]
10. Shen, S.; Hu, K.; Huang, L.; Li, H.; Han, R.; Cao, Q. Quantal response equilibrium-based strategies for intrusion detection in WSNs. *Mob. Inf. Syst.* **2015**, *2015*. [CrossRef]

11. Shen, S.; Hu, K.; Huang, L.; Li, H.; Han, R.; Cao, Q. Optimal report strategies for WBANs using a cloud-assisted IDS. *Int. J. Distrib. Sens. Netw.* **2015**, *2015*. [CrossRef]
12. Shen, S.; Li, Y.; Xu, H.; Cao, Q. Signaling game based strategy of intrusion detection in wireless sensor networks. *Comput. Math. Appl.* **2011**, *62*, 2404–2416. [CrossRef]
13. Yu, S.; Wang, G.; Zhou, W. Modeling malicious activities in cyber space. *IEEE Netw.* **2015**, *29*, 83–87. [CrossRef]
14. Wang, X.; Li, Q.; Li, Y. EiSIRS: A formal model to analyze the dynamics of worm propagation in wireless sensor networks. *J. Comb. Optim.* **2010**, *20*, 47–62. [CrossRef]
15. Tang, S. A modified SI epidemic model for combating virus spread in wireless sensor networks. *Int. J. Wirel. Inf. Netw.* **2011**, *18*, 319–326. [CrossRef]
16. Tang, S.; Myers, D.; Yuan, J. Modified SIS epidemic model for analysis of virus spread in wireless sensor networks. *Int. J. Wirel. Mob. Comput.* **2013**, *6*, 99–108. [CrossRef]
17. Shen, S.; Li, H.; Han, R.; Vasilakos, A.V.; Wang, Y.; Cao, Q. Differential game-based strategies for preventing malware propagation in wireless sensor networks. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 1962–1973. [CrossRef]
18. Wang, X.; Li, Y. An improved SIR model for analyzing the dynamics of worm propagation in wireless sensor networks. *Chin. J. Electron.* **2009**, *18*, 8–12.
19. Wang, X.; He, Z.; Zhao, X.; Lin, C.; Pan, Y.; Cai, Z. Reaction-diffusion modeling of malware propagation in mobile wireless sensor networks. *Sci. China Inf. Sci.* **2013**, *56*, 1–18. [CrossRef]
20. Wang, X.; He, Z.; Zhang, L. A pulse immunization model for inhibiting malware propagation in mobile wireless sensor networks. *Chin. J. Electron.* **2014**, *23*, 810–815.
21. Mishra, B.K.; Keshri, N. Mathematical model on the transmission of worms in wireless sensor network. *Appl. Math. Model.* **2013**, *37*, 4103–4111. [CrossRef]
22. Yu, S.; Gu, G.; Barnawi, A.; Guo, S.; Stojmenovic, I. Malware propagation in large-scale networks. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 170–179. [CrossRef]
23. Keshri, N.; Mishra, B.K. Two time-delay dynamic model on the transmission of malicious signals in wireless sensor network. *Chaos Solitons Fractals* **2014**, *68*, 151–158. [CrossRef]
24. Zhu, L.; Zhao, H. Dynamical analysis and optimal control for a malware propagation model in an information network. *Neurocomputing* **2015**, *149*, 1370–1386. [CrossRef]
25. Wang, Y.; Wen, S.; Xiang, Y.; Zhou, W. Modeling the propagation of worms in networks: A survey. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 942–960. [CrossRef]
26. Zou, C.C.; Towsley, D.; Gong, W. Modeling and simulation study of the propagation and defense of internet e-mail worms. *IEEE Trans. Depend. Secur.* **2007**, *4*, 106–118. [CrossRef]
27. Wen, S.; Zhou, W.; Zhang, J.; Xiang, Y.; Zhou, W.; Jia, W.; Zou, C.C. Modeling and analysis on the propagation dynamics of modern email malware. *IEEE Trans. Depend. Secur.* **2014**, *11*, 361–374. [CrossRef]
28. Wen, S.; Zhou, W.; Zhang, J.; Xiang, Y.; Zhou, W.; Jia, W. Modeling propagation dynamics of social network worms. *IEEE Trans. Parall. Distr.* **2013**, *24*, 1633–1643. [CrossRef]
29. Karyotis, V. Markov Random Fields for malware propagation: The case of chain networks. *IEEE Commun. Lett.* **2010**, *14*, 875–877. [CrossRef]
30. Khouzani, M.H.R.; Sarkar, S.; Altman, E. Saddle-point strategies in malware attack. *IEEE J. Sel. Area Commun.* **2012**, *30*, 31–43. [CrossRef]
31. Jin, C.; Jin, S.W.; Tan, H.Y. Computer virus propagation model based on bounded rationality evolutionary game theory. *Secur. Commun. Netw.* **2013**, *6*, 210–218. [CrossRef]
32. Spyridopoulos, T.; Maraslis, K.; Mylonas, A.; Tryfonas, T.; Oikonomou, G. A game theoretical method for cost-benefit analysis of malware dissemination prevention. *Inform. Secur. J.* **2015**, *24*, 164–176. [CrossRef]
33. Trajanovski, S.; Hayel, Y.; Altman, E.; Wang, H.; Van Mieghem, P. Decentralized protection strategies against SIS epidemics in networks. *IEEE Trnas. Control Netw. Syst.* **2015**, *2*, 406–419. [CrossRef]
34. AboElFotoh, H.M.F.; Iyengar, S.S.; Chakrabarty, K. Computing reliability and message delay for cooperative wireless distributed sensor networks subject to random failures. *IEEE Trans. Reliab.* **2005**, *54*, 145–155. [CrossRef]
35. Kar, K.; Krishnamurthy, A.; Jaggi, N. Dynamic node activation in networks of rechargeable sensors. *IEEE/ACM Trans. Netw.* **2006**, *14*, 15–26. [CrossRef]

36. Distefano, S. Reliability evaluation of WSN with dynamic-dependent nodes. *Int. J. Reliab. Qual. Saf. Eng.* **2011**, *18*, 515–530. [CrossRef]
37. Distefano, S. Evaluating reliability of WSN with sleep/wake-up interfering nodes. *Int. J. Syst. Sci.* **2013**, *44*, 1793–1806. [CrossRef]
38. Silva, I.; Guedes, L.A.; Portugal, P.; Vasques, F. Reliability and availability evaluation of wireless sensor networks for industrial applications. *Sensors* **2012**, *12*, 806–838. [CrossRef] [PubMed]
39. Niyato, D.; Wang, P.; Hossain, E. Reliability analysis and redundancy design of smart grid wireless communications system for demand side management. *IEEE Wirel. Commun.* **2012**, *19*, 38–46. [CrossRef]
40. Kamal, A.R.M.; Bleakley, C.; Dobson, S. Packet-level attestation (PLA): A framework for in-network sensor data reliability. *ACM Trans. Sens. Netw.* **2013**, *9*. [CrossRef]
41. Dâmaso, A.; Rosa, N.; Maciel, P. Reliability of wireless sensor networks. *Sensors* **2014**, *14*, 15760–15785. [CrossRef] [PubMed]
42. Wang, C.; Xing, L.; Vokkarane, V.M.; Sun, Y. Reliability and lifetime modeling of wireless sensor nodes. *Microelectron. Reliab.* **2014**, *54*, 160–166. [CrossRef]
43. Zonouz, A.E.; Xing, L.; Vokkarane, V.M.; Sun, Y.L. Reliability-oriented single-path routing protocols in wireless sensor networks. *IEEE Sens. J.* **2014**, *14*, 4059–4068. [CrossRef]
44. Cai, J.; Song, X.; Wang, J.; Gu, M. Reliability analysis for a data flow in event-driven wireless sensor networks. *Wirel. Pers. Commun.* **2014**, *78*, 151–169. [CrossRef]
45. Wang, Y.; Xing, L.; Wang, H.; Levitin, G. Combinatorial analysis of body sensor networks subject to probabilistic competing failures. *Reliab. Eng. Syst. Saf.* **2015**, *142*, 388–398. [CrossRef]
46. Yan, Z.; Nie, C.; Dong, R.; Gao, X.; Liu, J. A novel OBDD-based reliability evaluation algorithm for wireless sensor networks on the multicast model. *Math. Probl. Eng.* **2015**, *2015*. [CrossRef]
47. Zhu, X.; Lu, Y.; Han, J.; Shi, L. Transmission Reliability Evaluation for Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2016**, *2016*. [CrossRef]
48. Di Martino, C.; Cinque, M.; Cotroneo, D. Automated generation of performance and dependability models for the assessment of wireless sensor networks. *IEEE Trans. Comput.* **2012**, *61*, 870–884. [CrossRef]
49. Di Pietro, R.; Verde, N.V. Epidemic theory and data survivability in unattended wireless sensor networks: Models and gaps. *Pervasive Mob. Comput.* **2013**, *9*, 588–597. [CrossRef]
50. Bahi, J.M.; Guyeux, C.; Hakem, M.; Makhoul, A. Epidemiological approach for data survivability in unattended wireless sensor networks. *J. Netw. Comput. Appl.* **2014**, *46*, 374–383. [CrossRef]
51. Petridou, S.; Basagiannis, S.; Roumeliotis, M. Survivability analysis using probabilistic model checking: A study on wireless sensor networks. *IEEE Syst. J.* **2013**, *7*, 4–12. [CrossRef]
52. Xu, L.; Zhang, J.; Tsai, P.W.; Wu, W.; Wang, D.J. Uncertain random spectra: A new metric for assessing the survivability of mobile wireless sensor networks. *Soft Comput.* **2015**. [CrossRef]
53. Buzacott, J.A. Markov approach to finding failure times of repairable systems. *IEEE Trans. Reliab.* **1970**, *R-19*, 128–134. [CrossRef]