Journal of Computers Vol. 35 No. 2, April 2024, pp. 17-36
doi: 10.53106/199115992024043502002

# Malicious Code Propagation Model Based on Sleep Monitoring Mechanism in Wireless Sensor Networks

Si-Yu Hou[1], Jian-Guo Ren[1*], Yong-Hong Xu[2]

[1] School of Computer Science and Technology, Jiangsu Normal University, Xuzhou, Jiangsu, China
[2] Jiangsu Normal University, School of Life Sciences, Key Laboratory of Biotechnology and Medicinal Plants, Xuzhou, Jiangsu, China
{housiyu05, jsnucs1119} @163.com, xyh8810@126.com

*Received 9 July 2023; Revised 8 November 2023; Accepted 11 December 2023*

**Abstract.** Wireless sensor networks (WSNs) are characterized by high node density and finite energy storage. Each node exchanges information with all its neighboring nodes frequently, which makes it easy for nodes to run out of energy, leading to paralysis of the WSNs when facing network malicious code attacks. To address this problem, a malicious code propagation model based on sleep-monitoring technology for WSNs is proposed. As a Multi-compartment propagation, sleep nodes and monitoring nodes are introduced to the conventional SIR model. Sleep nodes can turn the infected node into a dormant state and stop the dissemination of information to save energy. Monitoring nodes can contain malicious codes spread in wireless sensor networks by sharing prevention information in real time. Additionally, by calculating the equilibrium point and propagation threshold of the new feedback model, the corresponding Lyapunov function is constructed to prove the local stability and global stability of the equilibrium point. Finally, the results of numerical simulation experiments show that when the sleep rate is 0.3 and feedback rate is 0.0000005, the number of infected nodes in the wireless sensor network decreases by 43.45%. Therefore, adding sleep-monitoring technology can effectively control the spread of malicious code in the networks.

**Keywords:** malware, wireless sensor network, sleep-monitoring mechanism, system dynamics

## 1 Introduction

Wireless sensor networks (WSNs) are widely utilized in industrial automation and the intelligent Internet of Things (IOT). They play a crucial role in perceiving the working status, monitoring equipment information, optimizing automation control, and improving industrial production efficiency. WSNs consist of numerous small, low-power, wireless communication, and self-organizing sensor nodes. Each sensor node in this system possesses data transmission, processing, and storage capabilities [1]. In comparison to the Internet, WSNs exhibit complex network topology, high node density, low energy storage, and limited communication range. Consequently, the efficient utilization of node energy has become a fundamental distinction between the two [2].

With the increasing adoption of wireless sensor networks in the Internet of Things, network security concerns have gained significant attention. Among various network attacks, malicious code has emerged as a critical issue in WSNs [3-4]. Malicious code refers to a program that leverages wireless transmission technology to autonomously propagate itself, without requiring manual infection. It possesses the ability to self-replicate and propagate, thereby presenting a serious threat. Therefore, it is imperative to employ effective measures to control the dissemination of malicious code within the network [3].

Firstly, in terms of the physical vulnerability of communication methods, Wireless Sensor Networks (WSNs) are prone to wireless attacks and abnormal signal interference because of their heavy reliance on wireless communication for linking a multitude of devices in the Internet of Things [4]. Incidentally, WSNs terminals often utilize resource-constrained devices due to cost considerations. These devices lack adequate protection and possess weak defense capabilities against attacks. Consequently, malicious code can exploit these vulnerabilities to carry out large-scale network attacks. Secondly, in terms of network characteristics, WSNs consist of densely deployed nodes that engage in frequent exchanges. This characteristic accelerates the spread of malicious codes within the network. Simultaneously, each node in the network has extremely finite storage capacity and energy

* Corresponding Author

resources. As nodes deplete their energy and memory, the entire network becomes paralyzed.

In light of these challenges, it is crucial to develop robust security mechanisms and strategies to mitigate the risks associated with malicious code in WSNs.

In previous research on virus transmission in wireless sensor networks, two primary research directions emerged. In the initial stages, researchers integrated the network topology of WSNs and studied network virus transmission based on biological infectious disease models (SIR, SI) [6-8]. However, as the research progressed, scholars increasingly recognized that while the dynamics of biological epidemics bear some resemblance to the propagation of network viruses, the unique configurability of network virus transmission and the transferability between nodes cannot be reflected by these typical models. Therefore, on this basis, they proposed some infectious disease models that conform to the characteristics of network virus transmission [9-10].

To sum up, the previous research does not take into account the node characteristics of the WSN network, and ignores the characteristics of node state transitions. In order to solve this problem, this paper proposes a model using WSNs sleep-monitoring technology. In this model, after the susceptible nodes are infected, some of them enter the sleep state to reduce the energy loss, nodes in sleep state are unable to receive and transform any information. At the same time, the infected nodes can also be transformed into a monitoring state to monitor the status of virus transmission in the network, and then generate prevention and control information feedback and back to the network to enhance the prevention and control capabilities of the entire network.

**Algorithm 1.** The process of worm attack and propagation

**Algorithm 0.1** The Process of **Worm Attack and propagation**.

1. generate an IP address
2. send TCP/SYN packets to machines randomly
3. **if** a TCP SYN-ACK packet is received
4. **then** complete the three-way handshake and establish a connection with the target machine
5. **else if** without receiving packet
6. **then** return line.2
7. send malicious code to connected machines
8. inducing connected machines to execute hostile code
9. the new infected machines start at line.1

**return** $S$

Major contributions of this paper are summarized as follows:

(1) Based on the node dormancy characteristics of the WSN networks, add sleep-monitoring technology, and add two new cabins on the basis of the SIR model: sleep ($S_2$) and monitoring (M). the infected node can enter the monitoring state, monitor malicious codes, generate virus prevention files, and send them to surrounding nodes. It makes the whole network system After the infected nodes have reached the threshold, it will trigger events and enter the sleep state.

(2) Carry out dynamic analysis on the $S_1S_2IMR$ model, and calculate the existence and uniqueness of its disease-free equilibrium point and endemic equilibrium point, as well as the transmission threshold, and perform stability analysis on the equilibrium point.

(3) Verify the correctness of the theoretical results through numerical simulation, and simulate the influence of each parameter in the model on the change of the number of infected nodes.

This paper is organized as follows: presents a novel epidemic model with latent by using the theory of epidemic models; analyzes the dynamical features of this model; by performing numerical simulations to verifies theoretic analysis. Finally, we end our investigations with brief conclusions.

This paper is organized as follows. Section 2 introduces the related work in the malicious model in WSNs. Section 3 presents the $S_1S_2IMR$ models. The basic reproduction number and the global stability of the worm-free equilibrium are investigated in Section 3. In Section 4, numerical simulations and suggestions are presented. Finally, Section 5 concludes the paper.

## 2 Related Work

Sleep technology in wireless sensor networks is an energy-saving technology, which can prolong the service life of nodes and reduce network energy consumption. The implementation of this technology is to set the nodes in the wireless sensor network to sleep when they do not need to send or receive data, so as to save energy. After the node enters the sleep state, its energy consumption will be greatly reduced. Additionally, incorporating monitoring technology can further enhance the network's security. This technology involves storing information about infected viruses and generating preventive measures to be added to the network feedback. This approach helps to enhance the network's resilience against malicious code and improve overall security.

By combining sleep technology for saving nodes energy and monitoring technology for enhancing security of systems, it can effectively extend the life of the node, reduce energy consumption, and improve the stability and reliability of the wireless sensor network in which a large number of nodes are randomly deployed. The Schematic diagram of Wireless Sensor Networks structure is shown in Fig. 1.

In recent years, researchers have been primarily focused on two research directions pertaining to the propagation of viruses in wireless sensor networks (WSNs). One is to incorporate computer network architecture to analyze the effects of various network environments and protocol parameters on the propagation of worm viruses, the other is to model based on biological infectious disease models. The relevant studies are shown in the Table 1.

**Table 1.** Research on network viruses in wireless sensor networks

| Author | Characteristics of the paper | Shortcomings of the paper |
|---|---|---|
| Shakya [6], Song [12] | Capturing the dynamic conditions of time and space. | Constrained to conventional biological models (SI, SIR), the proposed model has yet to substantiate its stability. |
| Luo [7] | Considering the communication radius of nodes. | Failing to take into account the node entry and removal rates d. |
| Wang [8] | Considering time delay factors in model design. | The model's feasibility remains unverified due to the absence of a comprehensive simulation. |
| Dong [18] | Proposing a fuzzy fractional SIQR model to describe dynamics of virus propagation with quarantine in the network. | Failure to consider the limitation and recovery mechanism of node energy. |
| Singh [9], Feng [19] | Considering the impact of network communication radius and node distribution density. | Constrained by a finite number of state intervals, resulting in a coarse representation of the virus propagation path that fails to capture the nuances inherent to WSNs. |
| Zhang [10] | Introducing an e-epidemic SITR mathematical model and analyzed node changes under varying noise intensities. | Confined to the influence of remote sensing network noise levels and does not reflect the applicability to WSNs. |
| Khayam [11] | Proposing a novel topology aware worm propagation model (TWPM). | Not discussing the impact of different parameters in the model on virus transmission. |
| Mishra [13], Ojha [14] | Proposing the two Exposure (E) and vaccine (V) cabins. | Not considering the finite energy of WSN networks nodes. |
| Song [15] | Considering the Multi-state antivirus measures and proposing an e-SEIR model in. | Insufficient simulation experiments to demonstrate the trend of transitions between different states. |
| Wang [16] | Established the corresponding sleep compartments for the S, I, and R, with node communication radius and node communication density as two parameters added to the model design. | Lack of model stability validation; data simulation is restricted to the discussion of node communication radius with insufficient investigation into the sleep node characteristics introduced by the model. |
| Jiang [17] | Introducing the MAC layer sleep/listening mechanism. | It does not directly utilize the sleep listening mechanism. |
| Hu [20] | Proposing the utilization of directional antenna technology to establish an SEIRS model incorporating rotating directional antennas. | Lack of proof of system stability in disease-free equilibrium and no discussion of sleep techniques. |

There are still many deficiencies in the virus propagation in wireless sensor networks studied in the above literature. For example, the energy management countermeasures of wireless sensor network nodes are not considered enough, and corresponding energy-saving measures cannot be taken according to the energy consumption of nodes in the network; Viruses, the corresponding virus prevention and control information cannot be generated in a targeted manner. In view of the above problems, this paper combines the characteristics of small energy storage and frequent communication of wireless sensor network nodes, uses sleep and monitoring technology, sets up two separate cabins, and proposes an improved $S_1S_2IMR$ model. It further describes the propagation dynamics of the virus spreading from a single node to the entire network, studies the virus propagation path between different node states, and introduces the impact of sleep-monitoring nodes on virus infection in the network.

In this paper, we propose a worm suppression strategy based on sleep-monitoring techniques to limit worm propagation in WSNs. We set the sensors in WSNs are divided into subsets, infected nodes in each subset enter the sleep state according to a set ratio. The rest of the infected nodes enter the monitoring state. For the nodes in the sleeping state, the virus cannot spread around, thus effectively restricting the spread of the virus; at the same time, the monitoring node collects the characteristics of the infected nodes and generates virus prevention and control information to be fed back to the susceptible nodes. In order to further improve the security of the model and verify the effectiveness of the model, we conducted a series of experiments. The experimental results show that the model can effectively detect malicious code in the process of information dissemination prevention and control. This will improve the network's prevention and control ability. By analyzing the experimental results, it can be seen that the model has superior performance under different network sizes and structures. It resists virus attacks and has strong scalability and practicality.

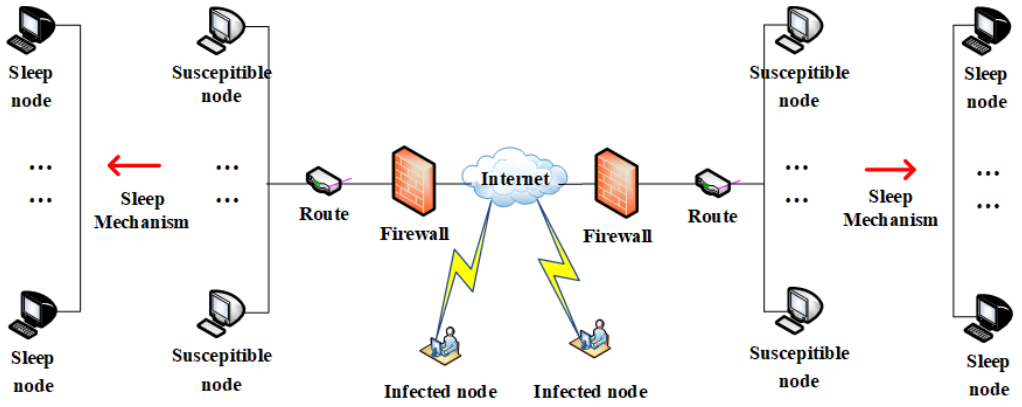


**Fig. 1.** Schematic diagram of wireless sensor networks structure

## 3 Modeling

Inspired by the classic SIR model, this paper proposes a new malicious code infection model: $S_1S_2IMR$ model according to event-driven sleep technology in wireless sensor networks. Inspired by the classic SIR model, this paper proposes a new malicious code infection model: $S_1S2IMR$ model according to event-driven sleep technology in wireless sensor networks. There are five states in all nodes of the model, and these five states are susceptible states (Susceptible, $S_1$), infected states (Infected, I), sleep states (Sleep, $S_2$), monitoring states (Monitoring, M), recovered states (Recovered, R).

The key mechanism of the $S_1S_2IMR$ model is to use sleep technology to perform forced dormancy on infected nodes, and to use monitoring technology to learn and share information about viruses in infected nodes, so as to cut off the virus transmission path and improve the system's ability to learn malicious code. To achieve the suppression of the spread of computer viruses. In order to achieve this effect, the key part can be realized by sleep-monitoring algorithm. For each infection node I in the set N(I), the specific process is as follows:

**Algorithm 2.** Sleep Monitoring mechanisms in WSNs

**Algorithm 0.2** Infected nodes Sleep-Monitoring Algorithm

**Input:** Infected Nodes $I = \{I_{i_1}, I_{i_2}, \ldots, I_{i_n}\}$, $I \in N(I)$
**Output:**Sleep Nodes $S_2 = \{S_{2,i_1}, S_{2,i_2}, \ldots, S_{2,i_n}\}$, $S_2 \in N(S_2)$;
Monitoring Nodes $M = \{M_{i_1}, M_{i_2}, \ldots, M_{i_n}\}$, $M \in N(M)$

```
for each I ∈ N(I) do
    if The number of infected nodes reaches the set dormancy threshold
      then Turn off main processorRF transceiver
           Suspend the execution of the task and close the communication operation
           Convert to Sleep nodes S2
           for each S2 ∈ N(S2) do
               if he number of sleep nodes reaches the set threshold
                 then Turn on the main processor
                      Trigger the wake-up mechanism
                      Resume communication operations
                      Convert to Recovered nodes R
               end if
           end for
    else
           Monitoring nodes detect the characteristics of network worms
           Add filename to malicious code list
           Generate virus prevention information and send it to all nodes in the system
           Convert to Monitoring nodes M
    end if
end for
return Infected nodes I // Go back to and loop the infected node again.
```

Combing the sleep-monitoring algorithm above helps to further study the state transition.

The $S_1S_2IMR$ model divides nodes into one of five different stages, and any node may be in any of these stages at any time. The state of nodes in our model is defined as follows:

(1) Susceptible ($S_1$): including nodes that are vulnerable to worm attack. After being attacked by the worm, nodes S have $\beta$ probability of being infected, becoming node I. Node S can reach node R both through autoimmunity and immunization through virus prevention and control information generated by node M.

(2) Infected (I): including nodes actively scanning for and spreading viral information. Nodes I are transformed into two states M and $S_2$ based on the code transportation rate of $\alpha$. According to the sleep rate $\theta$, some of the nodes I are transformed into nodes $S_2$, and the remainder into nodes M.

(3) Sleep ($S_2$): including nodes that can receive information, but cannot process them. Nodes S in state $S_2$ have a probability $\gamma$ of obtaining autoimmunity to reach the recovery state.

(4) Monitoring (M): including nodes that monitor signals of virus infection and generate prevention and treatment information. Node M has $\delta$ probability to feed back to the node R.

(5) Recovered (R): including nodes that have patched and thus immune to a worm attack temporarily.

The transformation between the states of the S1S2IMR model is shown schematically in Fig. 2.

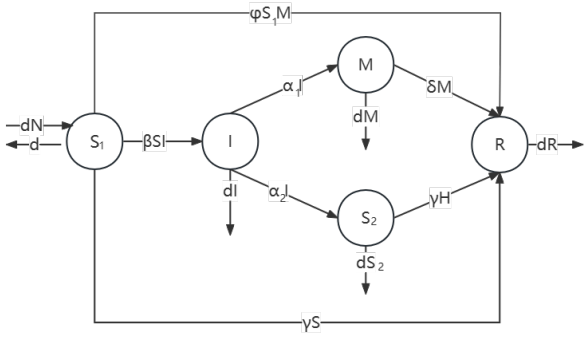


**Fig. 2.** The state-transition rules of the $S_1S_2IMR$ model

According to the state transition in Fig. 2, the parameter of the $S_1S_2IMR$ model in this paper can be written as Table 2:

**Table 2.** Notation and explanation for proposed models

| Notation | Explanation |
|---|---|
| $S_1(t)$ | Number of susceptible nodes at time t |
| I(t) | Number of infected nodes at time t |
| M(t) | Number of monitoring nodes at time t |
| $S_2(t)$ | Number of sleep nodes at time t |
| R(t) | Number of recovered nodes at time t |
| $\beta$ | Infection coefficient |
| $\alpha$ | Code delivery rate |
| $\theta$ | Sleep rate |
| $\delta$ | Feedback rate |
| $\varphi$ | Acquired immunity rate |
| $\gamma$ | Self-immunization rate, including $S_1 \rightarrow R$ and $S_2 \rightarrow R$ |
| $d$ | Natural decay rate |
| $N$ | Total number of nodes |

The set of differential equations for the $S_1S_2IMR$ model is the following system:

$$\begin{cases} \dfrac{dS_1(t)}{dt} = dN - \beta S_1 I - (\gamma + d)S_1 - \varphi S_1 M \\ \dfrac{dI(t)}{dt} = \beta S_1 I - (a + d)I \\ \dfrac{dM(t)}{dt} = a_1 I - (d + \delta)M \\ \dfrac{dS_2(t)}{dt} = a_2 I(\gamma + d)S_2 \\ \dfrac{dR(t)}{dt} = \gamma(S_1 + S_2) + \delta M - dR + \varphi S_1 M \end{cases} . \quad (1)$$

In particular, the first four equations in (1) do not depend on the fifth equation, so the system equations in (1) are the same as the following kinetic equations:

$$\begin{cases} \dfrac{dS_1(t)}{dt} = dN - \beta S_1 I - (\gamma + d)S_1 - \varphi S_1 M \\ \dfrac{dI(t)}{dt} = \beta S_1 I - (a + d)I \\ \dfrac{dM(t)}{dt} = a_1 I - (d + \delta)M \\ \dfrac{dS_2(t)}{dt} = a_2 I - (\gamma + d)S_2 \end{cases} . \quad (2)$$

In a dynamical system, the set of all possible points is a feasible region. In the dynamical system (2) of the $S_1S_2IMR$ model, the feasible region $\Omega$ is: $\Omega = \left\{ (S_1, I, M, S_2) \in R_+^4 : S_1 + I + M + S_2 \le N \right\}$.

This set is the positive invariant set of the system (2).

## 4 Stability Analysis

In the model construction of Chapter 3, the dynamic equation of the $S_1S_2IMR$ model can be used to determine the equilibrium point of the $S_1S_2IMR$ model.

The equilibrium point of the $S_1S_2IMR$ model is determined by the following equation system:

$$\begin{cases} \dfrac{dS_1(t)}{dt}=0 \\ \dfrac{dI(t)}{dt}=0 \\ \dfrac{dM(t)}{dt}=0 \\ \dfrac{dS_2(t)}{dt}=0 \\ \dfrac{dR(t)}{dt}=0 \end{cases}. \tag{3}$$

The only worm-free equilibrium $\mathbf{E_0}$ of model (2) can be easily obtained

$$E_0=\left(S_{1,0},I_0,S_{2,0},M_0\right)=\left(\frac{dN}{\gamma+d},0,0,0\right).$$

The only endemic equilibrium $E^*$ of model (2) can be easily obtained

$$E^*=\left(S_1^*,I^*,M^*,S_2^*\right)=\left(\frac{\alpha+d}{\beta},I^*,\frac{\alpha(1-\theta)I^*}{(d+\delta)},\frac{\alpha\theta I^*}{(d+\delta)(d+\gamma)}\right)$$

$$=\left(\frac{\alpha+d}{\beta},\frac{(d+\delta)[N\beta d-(\gamma+d)(\alpha+d)}{(\alpha+d)[\beta(d+\delta)+\alpha\varphi(1-\theta)},\frac{\alpha(1-\theta)[N\beta d-(\gamma+d)(\alpha+d)}{(\alpha+d)[\beta(d+\delta)+\alpha\varphi(1-\theta)},\frac{\alpha\theta[N\beta d-(\gamma+d)(\alpha+d)}{(\alpha+d)(d+\delta)[\beta(d+\delta)+\alpha\varphi(1-\theta)}\right).$$

According to Theorem 2 in [20], we compute the basic reproduction number through the spectral radius of a matrix [21-22].

Let $x=(I, S_1, M, S_2)^T$, the dynamic system (2) of model $S_1S_2IMR$ can be presented:

$$\frac{dx}{dt}=F(x)-V(x)$$

$$F(x)=\begin{pmatrix}\beta S_1\\0\\0\\0\end{pmatrix},\ V(x)=\begin{pmatrix}(\alpha+d)I\\ \beta S_1I+(\gamma+d)S_1+\varphi S_1M-dN\\(d+\delta)M-a_1I\\(\gamma+d)S_2-\alpha_2I\end{pmatrix}.$$

Then

$$\zeta=DF|_{E_0}\begin{pmatrix}\beta S_1&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{pmatrix},\ \xi=DV|_{E_0}\begin{pmatrix}\alpha+d&0&0&0\\ \beta S_1&\gamma+d&\varphi S_1&0\\-a_1&0&d+\delta&0\\-\alpha_2&0&0&\gamma+d\end{pmatrix}.$$

Then, we have the basic reproduction number [23] of model (3) is

$$R_0=\frac{\beta S_1}{(\alpha+d)}=\frac{\beta dN}{(\alpha+d)(\gamma+d)}. \tag{4}$$

### 4.1 Worm-free Equilibrium Stability

**Lemma1.** The worm-free equilibrium $E_0$ is locally asymptotically stable in Ω if $\Re_0 < 1$ and unstable if $\Re_0 > 1$.

**Proof.** According to $E_0 = (S_{1|0}, I_0, M_0, S_{2|0}) = (\frac{dN}{\gamma + d}, 0, 0, 0)$, the Jacobian matrix at the worm-free equilibrium $\mathbf{E_0}$ is

$$J(E_0) = \begin{pmatrix} -(\gamma + d) & -\beta S_1 & -\varphi S_1 & 0 \\ 0 & \beta S_1 - (\alpha + d) & 0 & 0 \\ 0 & a_1 & -(d + \delta) & 0 \\ 0 & \alpha_2 & 0 & -(\gamma + d) \end{pmatrix}. \quad (5)$$

The corresponding eigenvalues of $J(E_0)$ are

$$\begin{cases} \lambda_1 = -(d + \delta) \\ \lambda_2 = -(\gamma + d) \\ \lambda_3 = -(\gamma + d) \\ \lambda_4 = \beta S_1 - (\alpha + d) \end{cases}. \quad (6)$$

According to stability theory [24], for the four-dimensional model to be asymptotically stable, only when $\lambda_i < 0$, ($i = 1, 2, 3, 4$). Assuming all model parameters are positive, when $\Re_0 < 1$, $\beta S_1 - (\alpha + d) < 0$, it can be obtained that $\lambda_4 < 0$. So, when $\Re_0 < 1$, the worm-free equilibrium $E_0$ is locally asymptotically stable in Ω, on the contrary, when $\Re_0 > 1$, $\lambda_4 > 0$. Therefore, the worm-free equilibrium $E_0$ is an unstable saddle point.

**Lemma2.** The worm-free equilibrium $E_0$ is global asymptotically stable in Ω if $\Re_0 < 1$

**Proof.** From the first equation of Model 2, it can be obtained as follows

$S_1'(t) \le dN - (\gamma + d) S_1(t)$,

Thus $S_1(t) \le \frac{dN}{\gamma + d} - \left( S_1(0) - \frac{dN}{\gamma + d} \right) \exp\left[ -(\gamma + d) t \right]$.

When $t -> \infty$, We can obtain $S_1(t) \le \frac{dN}{\gamma + d}$ .

Consider a Lyapunov function:

$$V_1 = I. \quad (7)$$

The derivative of the model (3) with respect to the solutions along its trajectories is given by

$V_1' = I' = \beta S_1 I - (\alpha + d) I$

$= \frac{(\quad + d)}{(\alpha + d)} \beta S_1 I - (\alpha + d) I$

$\le \frac{(\alpha + d) dN}{(\alpha + d)(d + \gamma)} \beta I - (\alpha + d) I$

$= (\alpha + d)(R_0 - 1) I.$

When $\Re_0 < 1$, thus $\beta S_1 - (\alpha + d) < 0$, it can obtain that $V_1' < 0$. Therefore, according to the Lasalle invariance principle [26], when $\Re_0 < 1$, the disease-free equilibrium point $E_0$ of model (2) is globally asymptotically stable in the feasible region Ω.

This article verifies the stability of the disease-free equilibrium point $E_0$ in the $S_1S_2IMR$ model in Section 4.1. Through Theorem 1 and Theorem 2, it can be found that when $\Re_0 < 1$, the $S_1S_2IMR$ model will ultimately stabi-

lize at the disease-free equilibrium point $E_0$. This indicates that in Model (2), if $\Re_0 < 1$, the malicious code in the network will be gradually eliminated until it is completely eliminated.

### 4.2 Endemic Equilibrium Stability

**Lemma3.** The endemic equilibrium $E^*$ is locally asymptotically stable in Ω if $\Re_0 > 1$.

**Proof.** According to $E^* = (S_1^*, I^*, M^*, S_2^*)$, the Jacobian matrix at the endemic equilibrium $\boldsymbol{E}^*$ is

$$J\left(E^*\right) = \begin{pmatrix} -\beta I^* - (\gamma + d) - \varphi M^* & -\beta S_1^* & -\varphi S_1^* & 0 \\ \beta I^* & \beta S_1^* - (\alpha + d) & 0 & 0 \\ 0 & a_1 & -(d + \delta) & 0 \\ 0 & \alpha_2 & 0 & -(\gamma + d) \end{pmatrix}. \tag{8}$$

Thus, the corresponding characteristic equation can be denoted as

$$\lambda^4 + C_3\lambda^3 + C_2\lambda^2 + C_1\lambda + C_0 = 0. \tag{9}$$

Where,

$C_3 = \alpha + 4d + \delta + 2\gamma + \varphi M + (I - S_1)\beta > 0,$

$C_2 = (\alpha + \delta + \gamma)\beta I + (2d + \delta + \gamma)\varphi M + 2\delta(d + \gamma) + 4d\gamma + 3d^2 + \gamma^2 > 0,$

$C_1 = d^2(\delta + \gamma + d) + \gamma^2(\alpha + 2d + \delta) + I\alpha\beta(\delta + \gamma) + I\alpha\varphi(\alpha + d) + I\beta\delta\gamma$

$+2I\beta d(\alpha + \delta + \gamma) + 2d\delta\gamma + 3I\beta d^2 + I\alpha\varphi[d + \gamma - \theta(\alpha + \gamma)] > 0,$

$C_0 = I\beta(\alpha + d)(d + \gamma)(d + \delta) + I\alpha\varphi(\alpha + d)(d + \gamma)(1 - \theta) > 0.$

According to the calculation, it can be concluded that

$H_1 = C_3 > 0,$

$$H_2 = \begin{vmatrix} C_3 & 1 \\ C_1 & C_2 \end{vmatrix} = C_3C_2 - C_1 > 0,$$

$$H_3 = \begin{vmatrix} C_3 & 1 & 0 \\ C_1 & C_2 & C_3 \\ 0 & C_0 & C_1 \end{vmatrix} = C_1H_2 - C_3^2C_4 > 0,$$

$$H_4 = \begin{vmatrix} C_3 & 1 & 0 & 0 \\ C_1 & C_2 & C_3 & 0 \\ 0 & C_0 & C_1 & 0 \\ 0 & 0 & 0 & C_0 \end{vmatrix} = C_0H_3 > 0.$$

Therefore, according to the Lasalle invariance principle [26], when $\Re_0 < 1$, the endemic equilibrium point $E^*$ of model (2) is locally asymptotically stable in the feasible region Ω.

**Lemma4.** The endemic equilibrium $E^*$ is globally asymptotically stable in Ω if $\Re_0 > 1$.

**Proof.** Firstly,

$$g(x) = x - 1 - lnx\ . \tag{10}$$

It's easy to know that g(x) > 0 is constant established

Consider a Lyapunov function: [25]

$$V_2 = S_1^* g\left(\frac{S_1}{S_1^*}\right) + wg\left(\frac{I}{w}\right). \tag{11}$$

$$w = \frac{\beta dN - (\alpha + d)(\gamma + d)}{\beta(\alpha + d)}. \tag{12}$$

In Section 4 Stability Analysis, we have $\Re_0$ in Equation (4). When $\Re_0 > 0$, $\beta dN > (\alpha + d + \varepsilon)(\gamma + d)$, $w > 0$. By submitting (10) to (11), we can get:

$$V_2 = S_1 - S_1^* - S_1^* \ln\left(\frac{S_1}{S_1^*}\right) + I - w - w\ln\left(\frac{I}{w}\right).$$

The total derivative along model (2) is:

$$\frac{dV_2}{dt} = S_1^* \frac{1}{S_1^*} S_1'\left(1 - \frac{S_1^*}{S_1}\right) + w\frac{1}{w} I'\left(1 - \frac{w}{I}\right)$$

$$= \left(1 - \frac{S_1^*}{S_1}\right)\left[dN - \beta S_1 I - (\gamma + d) S_1 - \varphi S_1 M\right] + \left(1 - \frac{w}{I}\right)\left[\beta S_1 I - (\alpha + d) I\right]$$

$$\le \left(1 - \frac{S_1^*}{S_1}\right)\left[dN - \beta S_1 I - (\gamma + d) S_1\right] + \left(1 - \frac{w}{I}\right)\left[\beta S_1 I - (\alpha + d) I\right]$$

$$= -dN\frac{S_1}{S_1^*}\left(\frac{S_1^*}{S_1} - 1\right)^2 - dN + \frac{\beta dN}{\alpha + d} S_1 + (\alpha + d) I + \frac{(\gamma + d)(\alpha + d)}{\beta}$$

$$-(\gamma + d) S_1 - \frac{\beta dN - (\alpha + d)(\gamma + d)}{\alpha + d} S_1 - (\alpha + d) I + \frac{\beta dN - (\alpha + d)(\gamma + d)}{\beta}$$

$$= -dN\frac{S_1}{S_1^*}\left(\frac{S_1^*}{S_1} - 1\right)^2 < 0.$$

Thus, when $\Re_0 > 1$, the endemic equilibrium point $E^*$ of model (2) is globally asymptotically stable in the feasible region $\Omega$.

This article verifies the stability of the endemic equilibrium point $E^*$ in the $S_1S_2IMR$ model in Section 4.2. Through Theorem 3 and Theorem 4, it can be found that when $\Re_0 > 1$, the $S_1S_2IMR$ model will ultimately stabilize at the endemic equilibrium point $E^*$. This indicates that in Model (2), if $\Re_0 > 1$, the malicious code in the network will not disappear, but will continue to exist.

When $\Re_0 > 1$, the endemic equilibrium point $E^*$ of model (2) is globally asymptotically stable in the feasible region $\Omega$.

### 4.3 Sensitive Evaluation

In order to maintain the worm within a certain number, it is necessary to analyze the causes of the basic reproduction number $\Re_0$. As a next step, we perform a sensitivity evaluation of $\Re_0$.

From equation (4), the basic reproduction number of system (1) depends on the following parameters: $\alpha$, $\beta$, $\gamma$, $d$, Using a normalized sensitivity index (NSI), one may estimate the rate of change of $\Re_0$ given a change in the parameter value. NSI [parm] is defined as:

$$NSI[parm] = \frac{parm}{\Re_0} \cdot \frac{\partial \Re_0}{\partial [parm]}. \tag{13}$$

Thus, The NSI of $\Re_0$ with respect to $\alpha$, $\beta$, $\gamma$ and $d$ is:

$$NSI[\beta]=1.$$

$$NSI[\alpha]=-\frac{\alpha}{\alpha+d}.$$

$$NSI[d]=\frac{\alpha\gamma-d^2}{(\alpha+d)(d+\gamma)}.$$

$$NSI[\gamma]=-\frac{\gamma}{\gamma+d}.$$

From the above, we can see that $\Re_0$ decreases as β decreases or α, d, γ increase. To illustrate, we calculated the parameter values listed in Table 3 for the NSI for special cases. The NSI and corresponding % values in Table 4 indicate the change in parameter values required for a 1% decrease in $\Re_0$.

**Table 3.** Parameter values

| Parameter | Value | Unit |
|---|---|---|
| β | 0.0000001 | $Second^{-1}$ |
| α | 0.06 | $Second^{-1}$ |
| γ | 0.004 | $Second^{-1}$ |
| d | 0.00004 | $Second^{-1}$ |

From Table 3, to get a 1% decrease in the value of $\Re_0$, it is necessary to decrease the values of β and d by 1% and 1.0007%. Besides, a 1.01% increase in the values of γ and a 1.0007% increase in the values of α are required to achieve a 1% reduction in the value of $\Re_0$. Consequently, from the NSI, the optimum approaches of reducing the value of $\Re_0$ are to decrease the connection rate between susceptible and connected nodes (β), decrease the Natural decay rate from all nodes (d), and increase the delivery rate from infected nodes (α), increase the Self-immunization rate from susceptible and monitoring nodes to recovered nodes (γ), respectively.

**Table 4.** NSI of $\Re_0$ and change in parameter for 1% change in $\Re_0$

| Parameter | NSI [parm] | Corresponding % changes |
|---|---|---|
| β | 1 | 1 |
| α | -0.99933 | -1.0007 |
| γ | -0.9901 | -1.01 |
| d | 0.98943 | 1.0007 |

## 5 Numerical Simulations

Evaluate the effectiveness of the $S_1S_2IMR$ model through numerical simulation and simulation related experiments. This experiment was conducted using the MATLAB R2023a platform under the Intel Core i5-8265U CPU with a main frequency of 1.60 GHz, 8GB of memory, and Windows 11 operating system environment. In order to study the propagation pattern of malicious code in the Internet of Things, the initial number of nodes was set to 100000 in all numerical simulations and related experiments.

### 5.1 Changes of Each Node in the System Over Time

The $S_1S_2IMR$ model proposed in this paper is based on the classic SIR model. If the sleep and monitoring state after the infection node is not considered, the $S_1S_2IMR$ model will degenerate into the SIR model. Adding the sleep mechanism, the infected nodes are forcibly transformed into sleep nodes, cutting off the virus transmission path; at the same time, the monitoring technology is used to collect the information after the virus infection of the infected nodes, and feedback the generated virus prevention information. The sleep -monitoring mechanism can be divided into two parts: sleep and monitoring feedback, so it needs to be analyzed and demonstrated separately.

The first and second numerical simulation experiments were conducted to validate the SIRC proposed in the paper model's disease-free equilibrium point $E_0$ and viral equilibrium point $E^*$ of the SIRC model, that is, to verify the theorem in the stability analysis in Section 4.

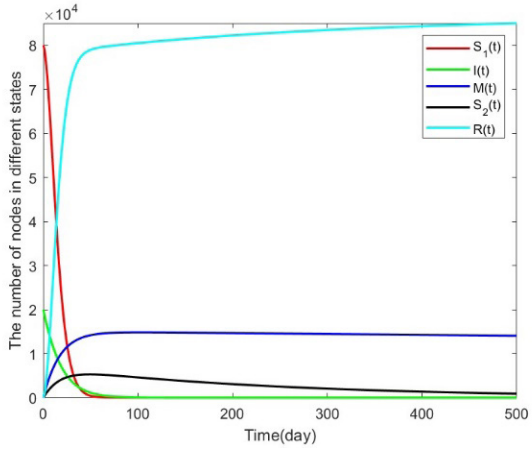


**Fig. 3.** Number of nodes in each state changes over time when $\Re_0 < 1$

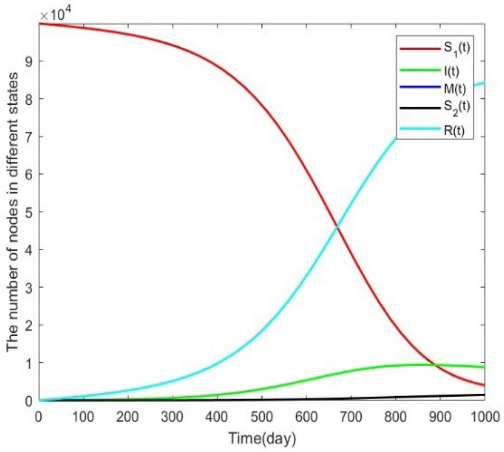


**Fig. 4.** Number of nodes in each state changes over time when $\Re_0 > 1$

In the first experiment, the number of initial nodes of susceptible nodes, infected nodes, monitoring nodes, sleep nodes and immune nodes were set as $S_1(0) = 80000$, $I(0) = 20000$, $M(0) = 0$, $S_2(0) = 0$, $R(0) = 0$. The selection of each parameter is as follows $\beta = 0.0000001$, $\alpha = 0.06$, $\theta = 0.3$, $\delta = 0.0001$, $\gamma = 0.004$, $\varphi = 0.00001$, $d = 0.00004$. According to the formula (4) obtained above, the basic reproduction number $\Re_0 = 0.0016491 < 1$ can be calculated. At this time, according to the relevant theories of Theorem 1 and Theorem 2, it can be known that the malicious code in the network will eventually disappear. Fig. 3 shows the change process of the number of nodes in each state with time, Fig. 3 shows that since the infected nodes infected the susceptible nodes through direct contact at the initial stage of the system, the number of infected nodes increased slightly in a short period, but then gradually decreased and approached at 0. Therefore, in the long run, the entire network is in an immune state, which is consistent with Theorem 1 and Theorem 2.

In the second experiment, while keeping the total number of 100000 nodes unchanged, unlike the first experiment, the initial number of nodes in each state was set to $S_1(0) = 99950$, $I(0) = 50$, $M(0) = 0$, $S_2(0) = 0$, and $R(0) = 0$. The selection of each parameter is as follows $\beta = 0.0000001$, $\alpha = 0.0006$, $\theta = 0.3$, $\delta = 0.0001$, $\gamma = 0.0001$, $\varphi = 0.00001$, $d = 0.0004$. According to formula (4) above, it can be obtained that $\Re_0 = 8 > 1$ at this time. By Theorem 3 and Theorem 4, it can be known that the infected nodes do not disappear with time; instead, the infected nodes grow and reach asymptotic stabilization at a later stage. From Fig. 4. In the early stage, the number of susceptible nodes decreases drastically, while the number of recovered nodes increases at a similar rate, and the number of infected nodes increases gradually, reaches a peak and then steadily stabilizes. It is due to the propagation of malicious information from the surrounding infected nodes that makes a sharp increase in the number of susceptible nodes, while the infected code enters the sleep state after reaching the set value, acquires immunity and arrives at the recovery state, the increase levels off and eventually stabilizes. Compared to Fig. 3, this scenario in Fig. 4 is closer to reality. In the WSN networks, adding sleep-monitoring technique can significantly reduce the size of the infected nodes in the short term, but also can learn and monitor the worm attack information in the WSN networks, feedback to the nodes, keep the infected nodes within the controllable range, and enhance the robustness of the WSN networks.

### 5.2 Comparison of the $S_1S_2IMR$ Model with Other Models

The third and fourth experiments were conducted to demonstrate that in WSN networks, the newly proposed $S_1S_2IMR$ model with sleep-monitoring technique is more effective than Kermack-Mckendrick's classical SIR

model (Model II), and the SIFR model (Model III) which considers the virus information prevention and control feedback, but does not take into account the infected node dormancy, the WSNs' sleep-monitoring technique model has better results in malicious code containment.

In both experiments, Model II does not consider the monitoring node feedback role and thus no acquired immunity, and also does not use the infected node dormancy technique, so it is not affected by the acquired immunity rate φ, feedback rate δ, and code delivery rate α, i.e., the acquired immunity rate φ = 0, the feedback rate δ = 0, the code delivery rate α = 0 and the sleep rate θ = 0. Constructing the classical SIR model, which is referred to in the text as Model II, its differential dynamics equation can be written as:

$$\begin{cases} \dfrac{dS(t)}{dt} = dN - \beta SI - (\gamma + d)S \\ \dfrac{dI(t)}{dt} = \beta SI - (d + \gamma)I \\ \dfrac{dR(t)}{dt} = \gamma I + \gamma S - dR \end{cases} . \tag{14}$$

Model III is not affected by the code delivery rate α, sleep rate θ, that is, the malicious code delivery rate α = 0, and the sleep rate θ = 0. A worm propagation model with feedback is proposed in the literature [27]. In order to better compare with the $S_1S_2IMR$ model construct a model SIFR in line with the above literature, which is called Model III in that paper, and its differential dynamics equation can be written as:

$$\begin{cases} \dfrac{dS(t)}{dt} = dN - \beta SI - (\gamma + d)S - \varphi SF \\ \dfrac{dI(t)}{dt} = \beta SI - (d + \gamma)I \\ \dfrac{dF(t)}{dt} = \gamma I - (d + \delta)F \\ \dfrac{dR(t)}{dt} = \gamma S + \delta F + \varphi SF - dR \end{cases} \tag{15}$$

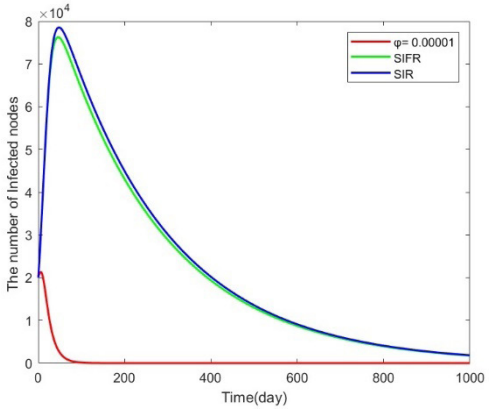


**Fig. 5.** Changes in the number of infected nodes in different systems over time when $\Re_0 < 1$

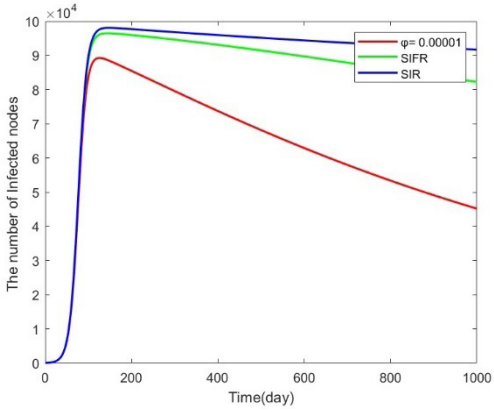


**Fig. 6.** Changes in the number of infected nodes in different systems over time when $\Re_0 > 1$

In the third experiment, the number of initial nodes is identical to that in experiment 1. The parameters of Model II are the same as Model I except φ = 0, δ = 0, α = 0, θ = 0. The parameters of Model III are the same as

Model I except that $\delta = 0.004$, $\alpha = 0$, $\theta = 0$. For the parameters are adjusted so that the three models satisfy the condition that the basic reproduction number $\Re_0 < 1$ As shown in Fig. 5, the sleep-monitoring technique brings the number of infected nodes down to converge to zero earlier and faster compared to the classical SIR model and SIFR model. On the other hand, it can make the peak value of the number of infected nodes much smaller than that of the two models. Moreover, the $S_1S_2IMR$ model with sleep-monitoring reduces the peak reached by the number of infected nodes in the point WSN networks compared to model III. The number of infected nodes at the peak is decreased by 72.02% in the case of the acquired immunity rate $\varphi$ of the $S_1S_2IMR$ model taken as 0.00001. It shows that the $S_1S_2IMR$ model with sleep technology can reduce the scale of malicious code propagation through dormancy technology, and at the same time disseminate anti-malicious code related prevention and control information, thus eliminating the problem of antivirus software not being able to respond in time to the new type of malicious code at the early stage of the system. This demonstrates that the $S_1S_2IMR$ model has the potential to effectively minimize malicious code propagation while promoting proactive measures to protect against it.

In the fourth experiment, the number of initial nodes is identical to that in experiment 2. The parameters of model II are the same as Model I except that $\varphi = 0$, $\delta = 0$, $\alpha = 0$, $\theta = 0$. The parameters of Model III correspond to those of Model I. At this time, the basic reproduction number $\Re_0 > 1$ holds for all three models. As shown in Fig. 6, When $\Re_0 > 1$, the infected nodes of the three systems will not disappear in the end, but always exist and converge to a stable peak value respectively, but in the WSN networks, the final stable peak value of the infected nodes of the SIFR model with the added feedback mechanism is also smaller than that of the traditional SIR model, and the final stable value of the infected nodes is smaller than that of model two. The $S_1S_2IMR$ model proposed in this paper is able to further reduce the peak value reached by infected nodes compared to model III. Taking the edge node feedback rate of 0.0000005 in the $S_1S_2IMR$ model, the number of infected nodes at the peak value is reduced by 43.45%.

### 5.3 The Effect of Other Parameters on the Model

*Infection rate β impact on changes in the number of infected nodes.* The next fifth and sixth experiments are designed to investigate the effect of different Infection coefficient on the number of infected nodes in the WSN networks.

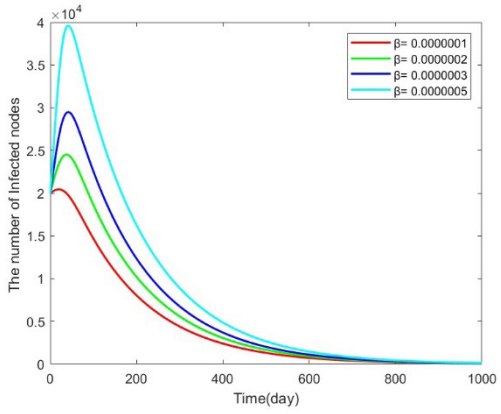


**Fig. 7.** Evolution of infected nodes with time under different infection coefficients when $\Re_0 < 1$

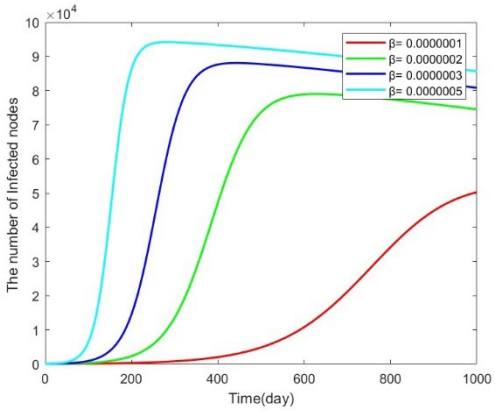


**Fig. 8.** Evolution of infected nodes with time under different infection coefficients when $\Re_0 > 1$

The fifth experiment is to compare the influence of different infection rates on the number of infected nodes when $\Re_0 < 1$. The infection coefficients are $\beta = 0.0000001$, $\beta = 0.0000002$, $\beta = 0.0000003$, $\beta = 0.0000005$, respectively, and to make the results more observable, we take the code delivery rate to be $a = 0.006$. Except for $\beta$ and a, the initial number of nodes in each state and other parameters are consistent with Experiment 1. From the changing trend of the curve in Fig. 7, it can be seen that at $\Re_0 < 1$, the infected nodes eventually decrease to 0 and

reach the steady state. With an increasing infection rate, the number of infected nodes increases faster, reaching a larger peak at a faster rate, and then gradually decreasing to 0 at a faster rate until the infection stabilizes. This indicates that in the early stage of malicious code propagation in WSN networks, the effect of controllable size of infected nodes can be achieved by controlling the infection coefficient between susceptible and infected nodes.

The sixth experiment is to compare the influence of different infection rates on the number of infected nodes when $\Re_0 > 1$. The infection coefficients are $\beta = 0.0000001$, $\beta = 0.0000002$, $\beta = 0.0000003$, $\beta = 0.0000005$, respectively, and to make the results more observable, we take the code delivery rate to be $a = 0.0006$. Except for $\beta$ and a, the initial number of nodes in each state and other parameters are the same as those of Experiment 2. From the curve trend in Fig. 8, it can be seen that the larger the $\beta$ value is, the faster the number of infected nodes increases and the earlier the peak value is reached. The relationship between infection rate, growth rate, and peak value of infected nodes is similar to that shown in the fifth experiment. When $\beta = 0.0000001$, the peak value of infected nodes is only half of $\beta = 0.0000005$, while the time to reach the peak value is four times as long. Therefore, by using firewalls, intrusion detection and defense techniques and other measures to reduce the worm's infection factor in the WSN networks, to achieve the purpose of suppressing the rapid spread of malicious code.

*Sleep rate θ impact on changes in the number of sleep nodes.* The next seventh and eighth experiments are designed to investigate the effect of different sleep rates on the number of sleep nodes in the WSN networks. In this part of the experiment, we also consider the extreme sleep case to study the trend of the number of sleeping nodes under the extreme sleep rate.

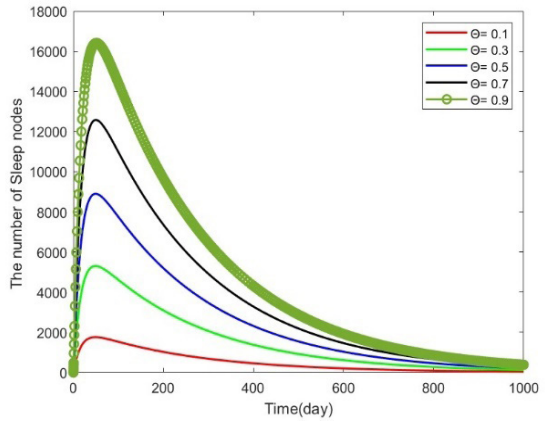


**Fig. 9.** Evolution of sleep nodes with time under different sleep rates when $\Re_0 < 1$

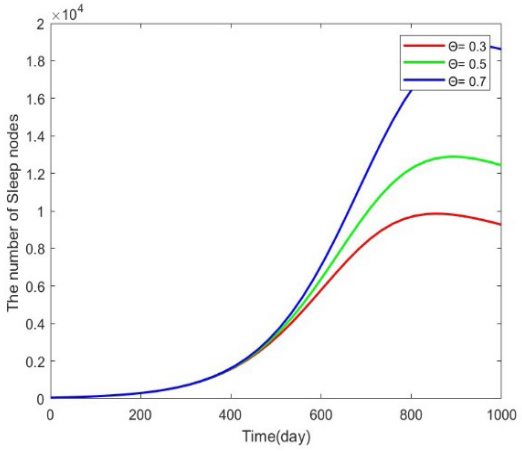


**Fig. 10.** Evolution of sleep nodes with time under different sleep rates when $\Re_0 > 1$

The seventh experiment is to compare the influence of different sleep rates on the number of sleep nodes when $\Re_0 < 1$. The sleep rates are $\theta = 0.1$, $\theta = 0.3$, $\theta = 0.5$, $\theta = 0.7$, $\theta = 0.9$, respectively, the initial number of nodes in each state and other parameters are consistent with Experiment 1. From the changing trend of the curve in Fig. 9, it can be seen that, when the sleep rate $\theta$ is 0.1, the sleep nodes account for the infected nodes only 8.24% of the number of infected nodes; when the sleep rate is 0.3, the peak number of sleeping nodes accounts for 24.85%, when the sleep rate is 0.5, it is 41.70%, when the sleep rate is 0.7, it is associated with 58.94%, and when the sleep rate $\theta$ is 0.9, it reaches 59.82%. From the values it can be seen that when the sleep rates accounts for less than 50% of the code delivery rates, only a small portion of the infected nodes are transferred to the sleep nodes to reduce the node energy consumption caused by a large number of malicious code attacks in a short period of time, the majority of the infected nodes are transferred to the monitoring state, and the WSN networks will not be paralyzed by the large number of dormant nodes.

The eighth experiment is to compare the influence of different sleep rates on the number of sleep nodes when $\Re_0 > 1$. We removed the sleep rates take of 0.9 too large and 0.1 too small and considered three other sleep rates takes, so the sleep rates are $\theta = 0.3$, $\theta = 0.5$, $\theta = 0.7$, respectively, the initial number of nodes in each state and other parameters are consistent with Experiment 2. From the changing trend of the curve in Fig. 10, it can be seen that, when the basic reproduction number $\Re_0 > 1$, the WSNs is in the steady state of proliferating malicious

code. The higher the sleep rate, the larger the number of sleeping nodes will be when the WSNs reaches this state. Therefore, when applying sleep technology in WSN networks, the sleep rates of the network can be controlled within a reasonable range by dynamically adjusting the sleep cycle and implementing remote monitoring and management of it. As a result, malicious code is less likely to spread to the steady state and the network's energy consumption is reduced.

*Code delivery rate α impact on changes in the number of infected nodes.* The next ninth and tenth experiments are designed to investigate the effect of different feedback rates δ on the number of infected nodes in the WSN networks.

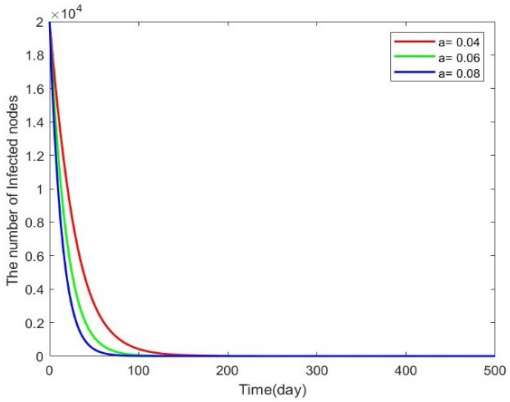


**Fig. 11.** Evolution of infected nodes with time under code delivery rates when $\Re_0 < 1$

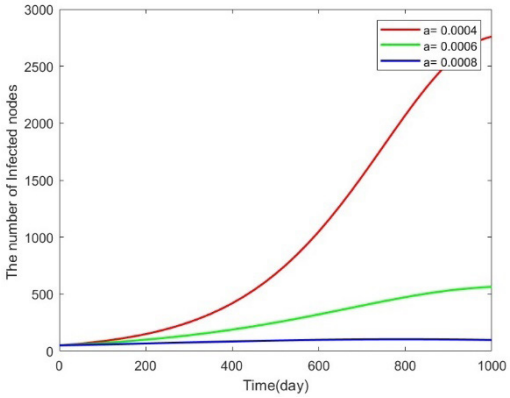


**Fig. 12.** Evolution of infected nodes with time under different code delivery rates when $\Re_0 > 1$

The ninth experiment is to compare the influence of different code delivery rates on the number of infected nodes when $\Re_0 < 1$. The code delivery rates are $\alpha = 0.04$, $\alpha = 0.06$, $\alpha = 0.08$, respectively, the initial number of nodes in each state and other parameters are consistent with Experiment 1. From the changing trend of the curve in Fig. 11, it can be seen that, when $\Re_0 < 1$, There is a negative correlation between the number of infected nodes and the code delivery rate α. In the early stages of malicious code infection, the number of infected nodes goes straight to zero over time. This means that malicious code can be effectively removed if the code transfer rate is high enough. If the code transfer rate is kept at a high level, it can also prevent malicious code from spreading.

The tenth experiment is to compare the influence of different code delivery rates on the number of infected nodes when $\Re_0 > 1$. The code delivery rates are $\alpha = 0.04$, $\alpha = 0.06$, $\alpha = 0.08$, respectively, the initial number of nodes in each state and other parameters are consistent with Experiment 2. From the changing trend of the curve in Fig. 12, it can be seen that, when the basic reproduction number $\Re_0 > 1$, as the code delivery rate increases, the increase in infected nodes slows down, and when the code delivery rate is 0.0008, the infected nodes can only be doubled at most. This strategy can be used to prevent the worm from spreading out of control, but it cannot completely prevent the worm from spreading. Therefore, other strategies, such as patching and system hardening, must be implemented to prevent worm spreading.

*Feedback rate δ impact on changes in the number of infected nodes.* The next eleventh and twelfth experiments are designed to investigate the effect of different feedback rates δ on the number of infected nodes in the WSN networks.

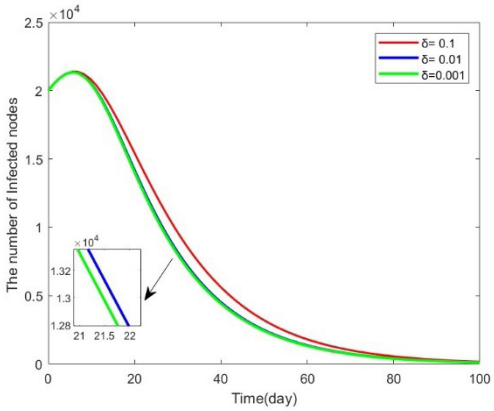


**Fig. 13.** Evolution of infected nodes with time under different feedback rates when $\Re_0 < 1$

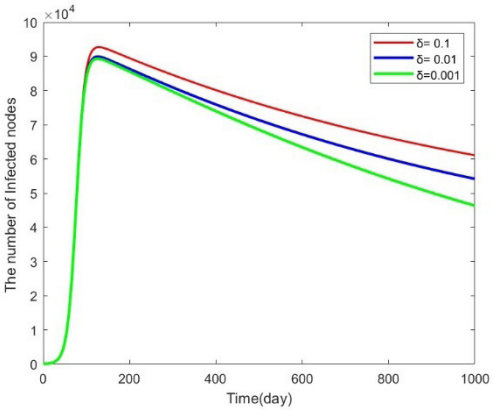


**Fig. 14.** Evolution of infected nodes with time under different feedback rates when $\Re_0 > 1$

Fig. 13 and Fig. 14 show the effect of feedback rates δ on the change of infected nodes when the basic reproduction number $\Re_0 < 1$ and $\Re_0 > 1$, respectively. Except for the feedback rate, the initial number of nodes and other parameters of each state for both models were the same as in Experiments 1 and 2. The feedback rate δ is set to 0.1, 0.01, and 0.001. As shown in Fig. 13, when $\Re_0 < 1$, the number of infected nodes gradually decreases to zero and the peak of the number of infected nodes gradually decreases with the increase of the feedback rate; and as shown in Fig. 14, the number of infected nodes does not decrease to zero over time when $\Re_0 > 1$. Instead, it gradually increases and reaches equilibrium. Improving the feedback rate can inhibit malicious code propagation at an early stage. This can be achieved by improving monitoring feedback measures such as enhancing node dense deployment to increase network fault tolerance. In addition, it can enhance the rate at which nodes collaborate to detect information, and other measures.

*Self-immunization rate γ impact on changes in the number of infected nodes.* The next thirteenth and fourteenth experiments are designed to investigate the effect of different Self-immunization rates on the number of infected nodes in the WSN networks.

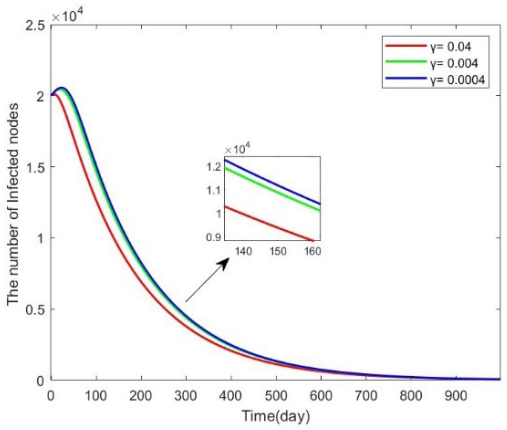


**Fig. 15.** Evolution of infected nodes with time under different Self-immunization rates when $\Re_0 < 1$

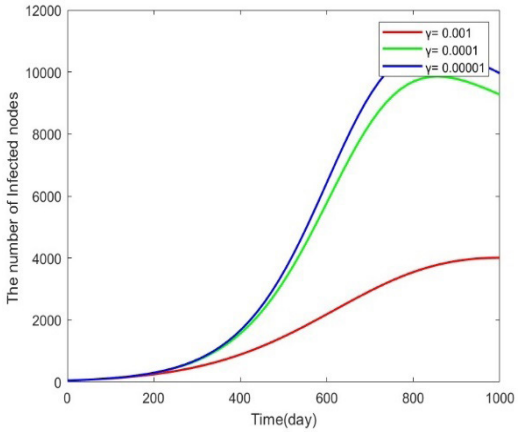


**Fig. 16.** Evolution of infected nodes with time under different Self-immunization rates when $\Re_0 > 1$

Figu. 15 and Fig. 16 show the effect of Self-immunization rate γ on the change of infected nodes when the basic reproduction number $\Re 0 < 1$ and $\Re 0 > 1$, respectively. Except for Self-immunization rate, the initial number of nodes and other parameters of each state for both models are the same as in Experiments 1 and 2. When $\Re_0 < 1$, self-immunization rate γ is set to 0.04, 0.004, and 0.0004, respectively; when $\Re_0 > 1$, self-immunization rate γ is set to 0.001, 0.0001, and 0.00001, respectively. As shown in Fig. 15, As self-immunization rates increase, the number of infected nodes gradually decreases to zero, and the peak value of the number of infected nodes gradually decreases; as shown in Fig. 16, when R0 > 1, the number of infected nodes does not decrease to zero over time, but gradually increases until equilibrium is reached. Self-immunization rates can be enhanced through multiple means such as enhanced virus library learning and updating, use of behavioral analysis techniques.

### 5.4 Discussion and Suggestion

Theoretical proofs and numerical simulations are given above to confirm the role of different parameters in the virus propagation model S1S2IMR for WSN networks. From the basic reproduction number $\Re_0$, it can be seen that in order to gradually reduce the number of worms in the network and reach equilibrium, the virus propagation model should be controlled to have $\Re_0 < 1$. $\Re 0$ in the model of this paper is as follows:

$$R_0 = \frac{\beta S_1}{(\alpha + d)} = \frac{\beta dN}{(\alpha + d)(\gamma + d)}.$$

The parameters β, α, d, and γ in the model are all important parameters affecting $\Re_0$, among which β and $\Re_0$ are positively correlated, and α, d, γ are negatively correlated with them. In order to achieve the value of the basic reproduction number $\Re_0$ as small as possible, it is necessary to control these parameters. The infection rate β can be controlled by installing a firewall and improving the security level of equipment. The improvement of the immunity rate γ can be obtained from the inside and outside of the system. Outside the system, measures such as timely vulnerability patching and user safety awareness education can improve the immunity rate of the system. In this model, the virus detected by the monitoring node M Information is returned to the network through feedback to generate malicious code prevention files, thereby enhancing the immunity rate. The determination of the sleep rate θ value needs to be based on the number of nodes in the network and the frequency of communication. When a large number of malicious code attacks are encountered in a short period of time, a rapid increase in the sleep rate can effectively control the number of worms in the system; at the same time, the higher the sleep rate is, the lower the network's working efficiency is.

Therefore, in order to control the spread of malicious code in wireless sensor networks, the following two points must be determined:

The first is to determine the dormancy threshold of the system, which is related to the network scale, the density of network node distribution and the stage of malicious code propagation. The optimization of the dormancy coefficient needs to simulate the network worm propagation algorithm, set the threshold number of infected nodes in the system, the duration of dormancy, and the probability of death after the dormancy state transition fails, and obtain more accurate parameter values to ensure that more nodes in the network Infected nodes can save energy, prolong the lifetime of network, gain immunity after the dormant state ends, and suppress the spread of worms in time.

The second is to improve the immunity of the system, including acquired immunity and self-immunization. Enhancing the system's ability to learn from malicious code begins with the establishment of a multilevel monitoring mechanism, real-time updating of malware databases, and the sharing and collection of threat intelligence. Subsequently, behavioral monitoring and heuristic analysis are implemented to monitor the abnormal behavior of applications and systems. This helps to detect unknown viruses and new types of attack. Finally, robust threat intelligence is continuously collected and analyzed, with threat analysis and research based on information from the security community, vulnerability reports, and threat intelligence agencies. Helps update virus monitoring rules and warns of new virus threats.

## 6 Conclusion and Future Work

In the wireless sensor network, the security defense level between the sensors is low, the average node energy storage capacity is small, and the information exchange is frequent. When attacked by large-scale network malicious codes, the system is easy to be paralyzed due to energy exhaustion. Considering the characteristics of

wireless sensor network information transmission, using node sleep technology and monitoring technology, the nodes in sleep state and monitoring state are introduced on the basis of the classic SIR model, and the $S_1S_2IMR$ model is constructed. Through dynamic analysis, the transmission threshold $\Re_0$ of the $S_1S_2IMR$ model is first obtained, and then the stability of the model is analyzed to prove the stability of the disease-free equilibrium point $E_0$ and the virus equilibrium point $E^*$ respectively. Finally, the simulation proof is carried out. The results of the numerical simulation It shows that when the feedback mechanism of the $S_1S_2IMR$ model is adopted, the number of infected nodes is significantly reduced, and as the feedback rate increases, the suppression effect on malicious code is better.

Further discussion findings and countermeasures suggest that by reducing the infection rate β, increasing the code delivery rate α and self-immunity rate γ, choosing a reasonable sleep rate θ. The spread of network worms can be suppressed, and the stability and energy efficiency of the network can be improved.

There are still limitations to the research in this paper. In the next step of research, the effect of sleep rate on worm propagation in wireless sensor networks under different node densities can be investigated by introducing WSNs topology and taking node densities and communication radius as environmental variables. At the same time, considering the propagation laws of different types of viruses, a computer virus propagation dynamics system model with virus specificity is established and studied.

## 7 Acknowledgement

This work is supported by the Natural Science Foundation of Jiangsu Province under Grant No. BK20201462, and the Natural Science Foundation of Xuzhou City under Grant No. KC21018, and the Scientific Research Support Project of Jiangsu Normal University under Grant No 21XSRX006, and the Xuzhou Science and Technology Planning Project (KC21194).

## References

[1] Q. Pei, Y. Shen, J. Ma, Survey of wireless sensor network security techniques, Journal on Communications 28(8)(2007) 113-122.

[2] S. Shen, S. Yang, L. Huang, J. Liu, G. Wu, H. Zhang, Q. Cao, A Dynamics Model of Disclosing Malware Propagation in Heterogeneous WSNs, Chinese Journal of Sensors and Actuators 35(5)(2022) 683-691.

[3] H. Zhang, W. Han, X. Lai, D. Lin, J. Ma, J. Li, Survey on cyberspace security, Science China Information Sciences, 58(11)(2015) 1-43.

[4] H. Li, L. Li, H. Shi, C. Zhou, R. Wang, Survivability evaluation in wireless sensor network, Application Research of Computers 35(8)(2018) 2450-2453.

[5] Z. Zhang, J. Zou, R.K. Upadhyay, G.R. Rahman, An epidemic model with multiple delays for the propagation of worms in wireless sensor networks, Results in Physics 19(2020) 103424.

[6] R.K. Shakya, Modified si epidemic model for combating virus spread in spatially correlated wireless sensor networks, <https://arxiv.org/abs/1801.04744>, 2018 (accessed 07.05.23).

[7] X. Luo, Y. Zhu, R. Huang, Analysis of spreading dynamics of virus in wireless sensor networks, Chinese Journal on Internet of Things 1(2)(2017) 63-67.

[8] Y. Wang, H. Yang, J. Zou, Z. Zhang, Hopf Bifurcation of a Delayed Predator-prey Model for Virus Propagation in Wireless Sensor Network, Journal of Binzhou University 38(4)(2022) 37-41.

[9] A. Singh, A.K. Awasthi, K. Singh, P.K. Srivastava, Modeling and analysis of worm propagation in wireless sensor networks, Wireless Personal Communications 98-99(2018) 2535-2551.

[10] H. Zhang, V. Madhusudanan, R. Geetha, M.N. Srinivas, C.H. Nwokoye, Dynamic analysis of the e-SITR model for remote wireless sensor networks with noise and sokol-howell functional response, Results in Physics 38(2022) 105643.

[11] S.A. Khayam, H. Radha, Using Signal Processing Techniques to Model Worm Propagation over Wireless Sensor Networks, IEEE Signal Processing Magazine 23(2)(2006) 164-169.

[12] Y. Song, G. Jiang, Investigating Malware Propagation over Wireless Sensor Networks, Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition) 29(4)(2009) 1-7.

[13] B.K. Mishra, N. Keshri, Mathematical model on the transmission of worms in wireless sensor network, Applied Mathematical Modelling 37(6)(2013) 4103-4111.

[14] R.P. Ojha, P.K. Srivastava, G. Sanyal, Security model against worms attack in wireless sensor network, International Journal of Advanced Intelligence Paradigms 13(1-2)(2019) 178-192.

[15] L. Song, R. Zhang, Dynamical Analysis for a Malware Propagation Model in Wireless Sensor Network, Journal of Measurement Science and Instrumentation 7(2)(2016) 136-144.

[16] X. Wang, Q. Li, Y. Li, EiSIRS: a formal model to analyze the dynamics of worm propagation in wireless sensor networks, Journal of Combinatorial Optimization 20(1)(2010) 47-62.
[17] L. Jiang, Y. Dai, H. Pan, Q. Xu, F. Dong, Research on SIR virus spreading in wireless sensor networks with sleep/listening mechanism, Experimental Technology and Management 38(3)(2021) 83-87.
[18] N. Dong, H. Long, N. Giang, The fuzzy fractional SIQR model of computer virus propagation in wireless sensor network using Caputo Atangana–Baleanu derivatives, Fuzzy Sets and Systems 429(2022) 28-59.
[19] L. Feng, L. Song, Q. Zhao, H. Wang, Modeling and Stability Analysis of Worm Propagation in Wireless Sensor Network, Mathematical Problems in Engineering 2015(2015) 1-9.
[20] J. Hu, Y. Song, Malware Propagation Model for Wireless Sensor Network Based on Rotating Directional Antenna, Computer Engineering 42(4)(2016) 119-125.
[21] M.G. Roberts, J.A.P. Heesterbeek, Characterizing the next-generation matrix and basic reproduction number in ecological epidemiology, Journal of Mathematical Biology 66(4-5)(2013) 1045-1064.
[22] C. Castillo-Chavez, Z. Feng, W. Huang, On the computation of $\Re 0$ and its role on global stability, Mathematical Approaches for Emerging and Re-emerging Infection Diseases: An Introduction, Springer, Berlin, 2002 (pp. 229-250).
[23] P.V.D. Driessche, J. Watmough, Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission, Mathematical Biosciences 180(1-2)(2002) 29-48.
[24] R.C. Robinson, An introduction to dynamical systems: continuous and discrete, American Mathematical Society, 2012.
[25] F. Wang, Y. Zhang, C. Wang, J. Ma, Stability analysis of an e-SEIAR model with point-to-group worm propagation, Communications in Nonlinear Science and Numerical Simulation 20(3)(2015) 897-904.
[26] J.P. La Salle, The stability of dynamical systems, Regional conference series in applied mathematics, SIAM, Philadelphia, Pa, USA, 1976.
[27] C. Li, J. Ren, Malware Propagation Model Based on Feedback Mechanism in Point-to-Group Networks, Computer Engineering 49(1)(2023) 163-172.