Symptom-aware hybrid fault diagnosis algorithm in the network virtualization environment

Yuze SU[*], **Xiangru MENG, Xiaoyang HAN, Qiaoyan KANG**
College of Information and Navigation, Air Force Engineering University, Xi'an, P.R. China

Abstract: As an important technology in next-generation networks, network virtualization has received more and more attention. Fault diagnosis is the crucial element for fault management and it is the process of inferring the exact failure in the network virtualization environment (NVE) from the set of observed symptoms. Although various traditional fault diagnosis algorithms have been proposed, the virtual network has some new characteristics, which include inaccessible fault information of the substrate network, inaccurate network observations, and a dynamic embedding relationship. To solve these challenges, a symptom-aware hybrid fault diagnosis (SAHFD) algorithm in the NVE is proposed in this paper. First, a multifactor Bayesian hierarchical model is proposed to denote the relationships between multiple factors in different layers. Second, the contribution degree is improved to locate the faults in the virtual layer and the active detection algorithm is introduced to filter some spurious faults in virtual layer fault diagnosis. Then, in substrate layer fault diagnosis, the active detection algorithm is introduced to solve the problem of incomplete network observations. Finally, a heuristic greedy algorithm is proposed to select appropriate actions based on minimum weight set covering method with minimum cost. Simulation results show that, compared with other algorithms, the SAHFD algorithm has a higher accuracy rate, lower false positive rate, and better environmental adaptability in the NVE.

Key words: Virtual network, fault diagnosis, symptom-aware, Bayesian network, active detection

1. Introduction

As a key technology to solve the ossification of Internet architecture and promote innovation, network virtualization has received extensive attention in recent years [1–3]. In the network virtualization environment (NVE), the network infrastructure and network services are decoupled and multiple heterogeneous virtual networks (VNs) are allowed to share the substrate network (SN) resources through abstraction, distribution, and isolation [4]. Each virtual network (VN) consists of virtual nodes and virtual links, which can be dynamically constructed and reasonably configured according to the actual needs of users. An example of the NVE is shown in Figure 1.

Different VNs are isolated from each other, which can use different network technologies and protocols to provide customized traditional network architecture and more flexible services, and greatly improve the efficiency of network resource utilization. Network virtualization technology effectively reduces the cost of network operation and improves the quality of network service, which has become one of the key technologies for next-generation Internet.

Faults are network events that are the root cause of problems in the network. Faults in SNs are not

---
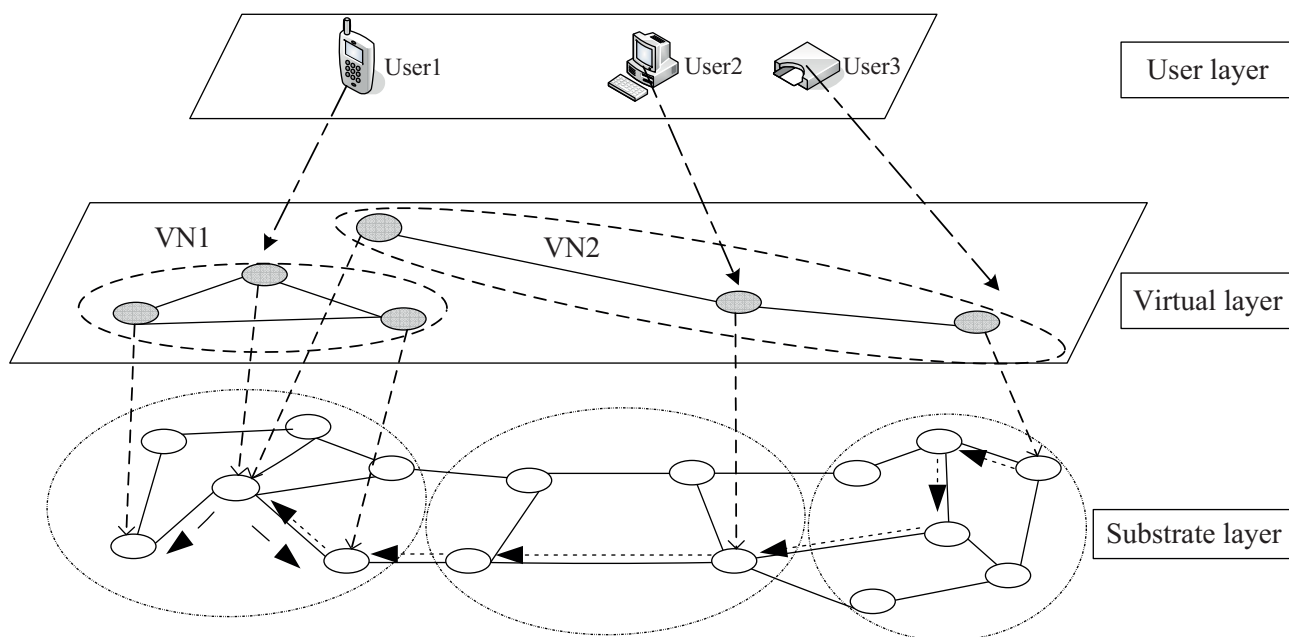
[*]Correspondence: glgiuip@163.com

Figure 1. Example of NVE.

uncommon due to a variety of reasons and the occurrence of faults becomes more frequent because of the introduction of network virtualization technology. A fault refers to an inability of a device or service to function correctly. Faults may occur in components that are hardware devices or software. In the NVE, resource consumption is faster and the network components easily fail. The VNs are embedded into the SN, and if the SN fails, all VNs embedded in it will break down. Hence, the detection, identification, and localization of faults in a timely manner are important for the reliable operation of VNs.

As an important part of fault management, fault diagnosis in the NVE is necessary to improve the survivability of the VN and it is the basis of faulty VN recovery. Fault diagnosis in the NVE is done to diagnose the fault types and positions accurately based on the symptoms, thus reducing the effect of faults on the services. Symptoms are external manifestations of the faults and they can be observed in various ways, such as by alarms in managed networks, by human observation, or by using active detection like probing. Alarms are considered to be symptoms of possible faults in the network. Hence, the relationship between symptoms and faults can be used to diagnose the faults. However, the characteristics of the NVE bring the following challenges for fault diagnosis in NVEs:

(1) The SN information is transparent to service providers (SPs). Different from traditional networks, infrastructure providers (InPs) and SPs are divided in NVEs. InPs are generally unwilling to share the detailed network information with SPs. In virtual layer fault diagnosis, the prior probability of the SN cannot be used and the faults that cannot be diagnosed in the virtual layer have to be transferred to the SN.

(2) The symptom set is large and the observed symptom information is inaccurate. The SN carries multiple heterogeneous VNs and each VN also carries multiple network services. Fault symptoms inevitably increase and it also takes a long time to diagnose the faults. In addition, some symptoms may be lost and some spurious symptoms may be added to the symptom set because of network noise.

(3) The fault types are various. The fault types in NVEs are more than in traditional networks. They

contain the independent faults and correlative virtual faults. The faults caused by software errors in the VN are called independent VN faults. If the VN faults are caused by some SN faults because of the embedding relationship between VN and SN, they are called correlative virtual faults. Some SN faults do not cause the failure of corresponding virtual components, which are called independent SN faults.

(4) The purpose of fault diagnosis in NVEs is different. The main purpose of fault diagnosis in NVEs is to find the source of faults to make preparations to reconstruct the VN or migrate nodes or links instead of fixing them immediately.

In view of the above background, the symptom-aware hybrid fault diagnosis (SAHFD) algorithm in the NVE is proposed. The main contributions are as follows:

(1) The SAHFD algorithm is proposed in this paper, which integrates active detection with passive monitoring to improve the performance of fault diagnosis in the NVE. In the SAHFD algorithm, the relationships between symptoms and faults are selected to diagnose the faults. The symptoms are collected from both passive monitoring and active detection to improve the accuracy of fault diagnosis.

(2) A multifactor Bayesian hierarchical model is proposed to take the factors of fault diagnosis into consideration, such as the embedding relationship and symptom-virtual layer fault causality. Different types of faults can be diagnosed and the contribution degree is improved to locate the virtual layer faults without SN information.

(3) Active detection is introduced to filter spurious virtual layer faults and locate substrate layer faults. Then the action selection problem is transformed into a minimum weight set covering problem (MWSCP) and a heuristic greedy algorithm is proposed to solve it.

This paper is structured as follows. Section 2 summarizes the related work on fault diagnosis algorithms in NVEs. In Section 3, we discuss the block diagram of the SAHFD algorithm. In Section 4, a multifactor Bayesian hierarchical model is proposed. In Section 5, we discuss the method to diagnose the virtual layer faults. Section 6 describes the method to diagnose the substrate layer faults. In Section 7, we propose the method to select the appropriate actions in active detection. In Section 8, we present the simulations to evaluate the performance of our algorithm. Finally, Section 9 includes the conclusions of this paper.

2. Related work

Fault management in the NVE includes two core aspects: fault diagnosis and fault recovery [5, 6]. Fault diagnosis uses the relevant technology and observed symptom set to locate the exact network faults. Fault diagnosis is the basis of fault recovery. In this paper, fault diagnosis methods in the NVE are summarized based on two aspects: passive monitoring and active detection.

2.1. Passive monitoring

In passive monitoring, the network is monitored by some monitoring devices [7]. Once a network component fails, alarms will be sent by monitoring units to the network management system (NMS) and these alarms will become symptoms in network monitoring system. The process is shown in Figure 2.

Passive monitoring technology usually has three categories: artificial intelligence techniques, model traversing techniques, and graph-theoretic techniques [5]. The Bayesian network in graph-theoretic techniques is most widely used. Huang in [8] proposed a max-covering algorithm (MCA) by transforming the fault diagnosis in a bipartite fault propagation model into a set-covering problem. This method could not diagnose the substrate layer faults in the NVE. Zhang et al. in [9] presented a service fault diagnosis algorithm based on inherent
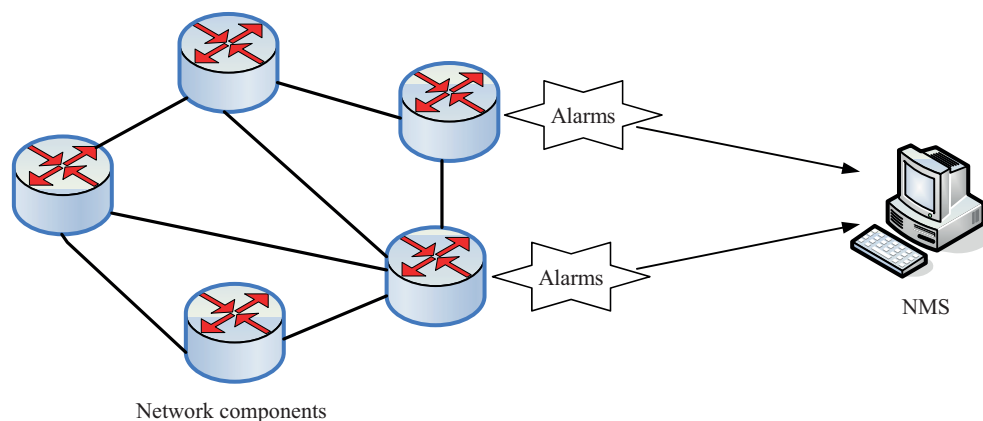
Figure 2. The process of passive monitoring.

correlation among symptoms in a dynamic NVE. They made a service fault propagation model based on embedding relationships to solve the problem that the information of the SN was transparent to SPs. However, the independent VN faults could not be diagnosed. Yan et al. in [10] proposed a multilayer fault diagnosis (MLFD) algorithm in the NVE. A layer-by-layer strategy and a filtering algorithm were proposed to distinguish the different faults in the NVE. Liu et al. in [11] proposed an improved VN fault diagnosis algorithm based on a trust evaluation algorithm to improve the fault diagnosis accuracy and reduce the false positive rate. This method was sensitive to network noise. Gillani et al. in [12] used the formal evidential reasoning for VN fault localization and the problem was formulated as a constraint-satisfaction problem. This model exploited the correlation in the end-user negative symptoms due to shared symptoms and path segments, and in conjunction it also identified network loss invariants, which were all then encoded as constraints and solved using satisfiability modulo theories. However, this approach inherited the problem of insufficient observations by end-users. Wang et al. in [13] presented a fault diagnosis system in VNs. It improved the existing end-user fault diagnosis method by screening evidence before analyzing to reduce time-consumption and improve the antinoise ability of the system. Bennacer et al. in [14] presented a hybrid approach that combined a Bayesian network and case-based reasoning in order to overcome the usual limits of fault diagnosis techniques and reduce human intervention in this process. The proposed mechanism allowed the identification of the root cause with fine precision and higher reliability. Zhang et al. in [15] proposed a fault diagnosis method by deep learning to predict the failure of a VN. It enabled earlier failure prediction by the long short-term memory network, which discovered the long-term features of the network history data.

However, all the above algorithms were sensitive to network noise, which would lead to the loss of symptoms and spurious symptoms.

2.2. Active detection
In active detection, the probing station is selected to transmit one or more packets called probes for the purpose of monitoring the state of the network [16]. It helps the NMS to respond more quickly and accurately to a large number of network events, as opposed to traditional passive monitoring technology.

Probe station selection was the basis of active detection in NVEs. Liu et al. in [17] proposed a dynamic probe station selection algorithm based on the greedy method in view of the fact that existing methods could not adapt to the dynamics of VNs. Pan et al. in [18] proposed a novel algorithm for probe station selection in

the nondeterministic network environment using a probabilistic dependency model. Tang et al. in [19] proposed a overlay fault diagnosis framework, which integrated the use of active detection with passive monitoring. It was the first combination of these two methods in overlay network fault diagnosis. However, the overlay network was different from the NVE and this method could not solve the challenges of fault diagnosis in the NVE as introduced in Section 1.

Both passive monitoring and active detection have their advantages and limitations. Passive monitoring takes advantage of the intrinsic correlation between the symptoms and network components. It does not consume probing resources. However, the symptoms collected are sensitive to the network noise. Active detection consumes many probing resources and the accuracy of fault diagnosis is better. Hence, it is of great significance to find a fault diagnosis algorithm in the NVE to balance resource consumption and diagnosis performance.

3. Block diagram of the SAHFD algorithm

In the NVE, there are many complex relationships. Symptoms are external manifestations of the faults. Network services are provided by VNs and VNs are embedded into the SN. The relationship between the VN and SN is also obvious. In this paper, we make full use of the existing relationships and propose the SAHFD algorithm, which diagnoses the faults layer by layer in the NVE.

The SAHFD algorithm is divided into four functional modules: the multifactor Bayesian hierarchical model, virtual layer fault diagnosis, substrate layer fault diagnosis, and active detection technology, as shown in Figure 3.
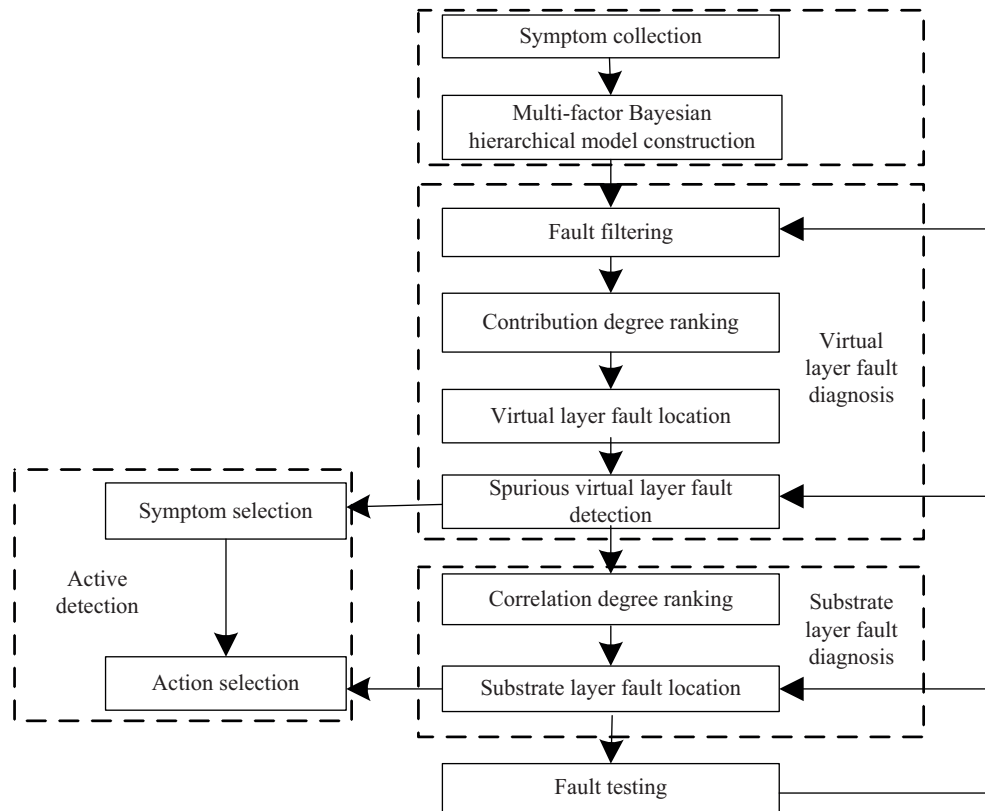

Figure 3. The block diagram of the SAHFD algorithm.

In the NVE, the multifactor Bayesian hierarchical model is constructed based on the relationships between different layers. The virtual layer fault diagnosis contains four modules. Fault filtering is used to filter some spurious faults. Then the suspected virtual layer faults are ranked based on the contribution degree and the virtual layer faults with high contribution degree are selected to explain the symptoms. At last, active detection is introduced to diagnose some suspected virtual layer faults based on the spurious fault detection threshold. The fault diagnosis results in the virtual layer are inputs of the fault diagnosis in the substrate layer. The embedding relationship between the VN and SN is used to establish the fault correlation model. Then the virtual and substrate faults can be distinguished effectively on the basis of correlation degree threshold. Active detection is used to filter spurious faults in the virtual layer and locate the faults in the substrate layer. Active detection includes symptom selection and action selection. Finally, the actual fault testing results are used as the prior information to adjust the thresholds used in the process of fault filtering, spurious fault filtering, and substrate fault diagnosis adaptively to improve the performance of the SAHFD algorithm.

4. Multifactor Bayesian hierarchical model

As shown in Figure 4, a multifactor Bayesian hierarchical model is presented to correlate the received end-to-end symptoms and different layers in the NVE. There are multiple factors in this Bayesian model, such as actions, symptoms, virtual layer faults, substrate layer faults, embedding relationships, and symptom-virtual fault causality. These symptoms contain the symptoms collected by network monitoring and the symptoms generated by the active detection. The relationship between symptoms and different layers in the NVE can be obtained as follows. If the substrate components fail, the corresponding virtual components embedded into the faulty substrate components will also fail. Hence, the causality between symptoms and virtual layer components can be used to locate the virtual layer faults with the help of observed symptoms. The embedding relationship between virtual layer components and substrate layer components can be used to distinguish the fault types. Owing to the network noise, some symptoms are lost and some spurious symptoms occur, which reduces the accuracy rate of fault diagnosis. Active detection is introduced to select some actions to detect the symptoms based on the causality between actions and symptoms.
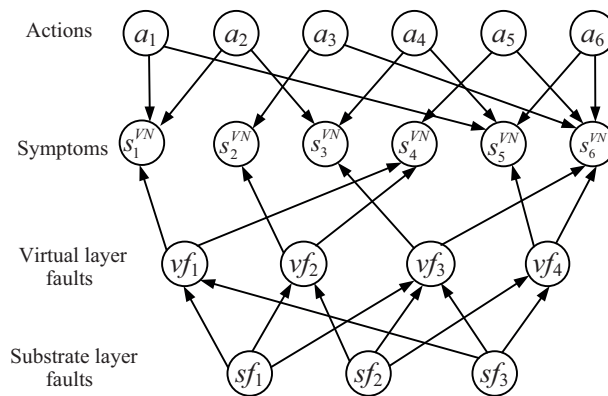

Figure 4. Multifactor Bayesian hierarchical model.

In this paper, $SF = \{sf_1, sf_2, \ldots, sf_k\}$ is used to denote the substrate layer faults and $VF = \{vf_1, vf_2, \ldots, vf_k\}$ is used to denote the virtual layer faults. $S^{VN} = \{s_1^{VN}, s_2^{VN}, \ldots, s_k^{VN}\}$ is used to denote the symptoms. When the network component is normal, the corresponding symptom is positive. Otherwise, negative symptoms occur. We use $A = \{a_1, a_2, \ldots, a_k\}$ to denote the actions in active detection. Action usually

means transmitting a probe to obtain the end-to-end symptoms, which can be used to infer the state of network components.

At the same time, some limitations of this paper are as follows:

(1) Different nodes in one VN cannot be embedded into the same substrate nodes.

(2) All faults are independent of each other.

(3) All potential faults can be directly observed with the help of actions.

(4) The simultaneous occurrence probability of multiple faults is low. In this paper, we select the least faults to explain the symptoms.

5. Virtual layer fault diagnosis

As can be seen from Figure 4, active detection is incorporated into passive monitoring to diagnose the faults layer by layer. Many network protocols and applications run on the VNs. Hence, we use the Bayesian network and active detection to diagnose the faults in the virtual layer first.

5.1. Fault filtering

Compared with traditional networks, the relationships between different layers in the NVE are more complex. Some observed symptoms may be changed because of the network noise. These spurious symptoms cannot accurately reflect the states of network components. For instance, if the observed symptoms s_1^{VN}, s_2^{VN}, s_3^{VN}, and s_4^{VN} are negative, the virtual node vn_1 may break down. Another symptom s_5^{VN} being positive reflects that vn_1 may be normal. We cannot distinguish the state of vn_1 from its associated symptoms. Since the probability of spurious symptoms is very low, the number of spurious symptoms is small. In this paper, we use confidence degree to filter the spurious faults, which was explained in [13]. It is defined as follows:

$$\alpha_i = \frac{num(S_{O_i}^{VN})}{num(S_i^{VN})},$$ (1)

where $num(S_{O_i}^{VN})$ denotes the number of all observed symptoms related to fault vf_i. $num(S_i^{VN})$ denotes the number of all symptoms caused by the fault vf_i. α_i denotes the confidence degree of fault vf_i, $0 < \alpha_i \leq 1$.

In the fault filtering algorithm, the confidence degrees of all suspected virtual faults in VF_1 are calculated. If the confidence degree α_i of the suspected virtual layer fault is not larger than the confidence degree threshold th_1, this fault is considered to be spurious and it will be deleted. Usually, th_1 can be set and changed dynamically based on the network environment and noise. For instance, the corresponding observed symptoms s_1^{VN}, s_2^{VN}, s_3^{VN}, and s_4^{VN} are negative; another symptom s_5^{VN} is positive; and α_i=0.8. If $th_1 = 0.7$, $\alpha_i > th_1$ and vn_1 is diagnosed as breaking down. After filtering the spurious faults, the suspected virtual layer fault set FVF is obtained. For every symptom in symptom set S_1, we search the fault components in FVF to explain it. If the symptom cannot be explained, this symptom is considered to be spurious and it will be filtered from S_1.

5.2. Virtual layer fault location

In virtual layer fault location, contribution degree is selected as a criterion to find faults that have the maximal contribution to the observed symptoms. The Bayesian network and the prior information of suspected virtual

layer fault fvf_i in FVN are selected to calculate the contribution degree. It is defined as follows:

$$C(fvf_i) = \frac{\sum_{s_i^{FVN} \in S_{O_i}^{FVN}} u(fvf_i \mid S_i^{FVN})}{\sum_{s_i^{FVN} \in S_i^{FVN}} u(fvf_i \mid S_i^{FVN})}, \qquad (2)$$

where contribution degree $C(fvf_i)$ indicates the fault probability of fvf_i. $S_{O_i}^{FVN}$ denotes the observed symptom set related to fvf_i. S_i^{FVN} denotes the symptom set related to fvf_i. $u(fvf_i \mid S_i^{FVN})$ is the relative probability of fvf_i happening by observing s_i^{FVN}. It is defined as follows:

$$u(fvf_i \mid S_i^{FVN}) = \frac{p(s_i^{FVN} \mid fvf_i) \cdot p(fvf_i)}{\sum_{fvf_i \in F_{s_i^{FVN}}} p(s_i^{FVN} \mid fvf_i) \cdot p(fvf_i)}, \qquad (3)$$

where $p(fvf_i)$ denotes the prior probability of fvf_i. $p(s_i^{FVN} \mid fvf_i)$ denotes the causal certainty between fvf_i and s_i^{FVN}. $F_{s_i^{FVN}}$ denotes the fault set that can cause the symptom s_i^{FVN}. In the NVE, virtual components are the logic abstraction of substrate components. The prior information of SN components is transparent to the VN. Hence, the prior information of the virtual software is selected to denote $p(fvf_i)$. After fault filtering, the symptom set caused by fvf_i is denoted by S_i^{FVN}.

In the virtual layer fault location algorithm, the faults in FVF are ranked based on the contribution degree from large to small. If there is only one component in FVF, we save it in the suspected virtual layer fault set VF_2. Otherwise, we calculate the intersection of S_i^{FVN} and S_1, which is denoted by Φ_i. Rank Φ_i is based on the number of symptoms from large to small. There are three situations, as follows:

(1) If $\Phi_i \subset \Phi_j$ $(i < j)$, delete fvf_i from FVF.

(2) If the lengths of Φ_i and Φ_j are equal, save fvf_i and fvf_j into VF_2 and delete them from FVF. Delete the intersection of S_i^{FVN} and S_1 from S_1. Delete the intersection of S_j^{FVN} and S_1 from S_1.

(3) If the lengths of Φ_i and Φ_j are different, calculate the intersection of S_i^{FVN} and S_1. If the intersection is empty, delete fvf_i from FVF. Otherwise, save fvf_i into VF_2 and delete it from FVF. Delete the intersection of S_i^{FVN} and S_1.

Go on with these steps until S_1 is empty.

5.3. Spurious fault detection

In the virtual layer fault location algorithm, all symptoms in S_1 are explained by the virtual layer fault location algorithm. However, some spurious faults are also introduced into VF_2. There are inherent correlations between actual symptoms and faults. The fault with high contribution degree is preferred to explain the symptoms in S_1. After repeated fault diagnosis, the residual symptoms are often spurious. In order to detect these spurious faults, the spurious fault detection algorithm is proposed.

In the spurious fault detection algorithm, the faults in VF_2 are ranked based on the contribution degree from small to large. Take out the first β elements in VF_2 and save them in VF_3, in which $\beta = \rho \cdot length(VF_2)$. ρ is the spurious fault detection threshold. Take out the first β elements in VF_2 and save them in VF_3. Run the active detection algorithm to diagnose the suspected faults in VF_3 and get the result set VF_{31}. Calculate the virtual layer fault set VF_4, in which $VF_4 = VF_{31} \cup (VF_2 - VF_3)$.

6. Substrate layer fault diagnosis

After virtual layer fault diagnosis, it is not clear whether the faults in the virtual layer are independent virtual faults or correlative virtual faults. They can be distinguished with the help of the embedding relationships between the virtual layer and substrate layer. At the same time, the results of virtual layer fault diagnosis are the inputs of substrate layer fault diagnosis and there may be some spurious virtual layer faults. In the substrate layer fault diagnosis algorithm, the correlation degree threshold and active detection are introduced to improve the performance of fault diagnosis. The definition of correlation degree is as follows:

$$\gamma_i = \frac{num(DVF_{sf_i})}{num(VF_{sf_i})}, \tag{4}$$

where γ_i denotes the correlation degree and $0 < \gamma_i \leq 1$. $num(DVF_{sf_i})$ denotes the number of faulty virtual components embedded onto substrate fault sf_i. $num(VF_{sf_i})$ denotes the total number of virtual components embedded into the substrate fault sf_i. The correlation degree of the substrate fault is $\gamma_i = 1$ if sf_i occurs. Owing to the symptom loss, some faulty virtual components can be diagnosed and sf_i may occur when $\gamma_i < 1$. Hence, active detection is introduced to detect the suspected lost virtual component set LVF_{sf_i}. It is defined as follows:

$$LVF_{sf_i} = VF_{sf_i} - DVF_{sf_i}, \tag{5}$$

where VF_{sf_i} denotes the total virtual components embedded into substrate fault sf_i. DVF_{sf_i} denotes the faulty virtual components embedded into substrate fault sf_i.

In the substrate layer fault diagnosis algorithm, the correlation degrees of suspected substrate faults are calculated and compared with the correlation degree threshold. If $\gamma_i > th_2$, the suspected lost virtual components are detected by the active detection algorithm. Then the results are added into VF_4 and the correlation degrees are calculated again to distinguish the fault types. If the new correlation degree is $\gamma'_i = 1$, the corresponding $sf(i)$ is the substrate fault. The residual suspected virtual faults in VF_4 are independent virtual faults.

7. Active detection algorithm based on minimum weight set covering

In this paper, we integrate active detection with passive monitoring in the NVE. As can be seen from Figure 4, each action can detect the network component and generate several symptoms. These symptoms are added to the symptom set and used to diagnose the faults. Several symptoms can be generated by one action and one symptom can be generated by several actions. Different actions have different costs. The goal of active detection is to select suitable actions to detect the faults with minimum cost. This problem can be defined as finding several actions to cover the symptoms that need to be generated and used in fault diagnosis with the minimum total cost. The MWSCP is used to find the subset of one set to cover another given set. Hence, the action selection problem can be transformed into the MWSCP.

Theorem 1 *MWSCP*

For symptom sets $\{s_1, s_1, \ldots, s_m\}$ and $\{S_{a_1}, S_{a_2}, \ldots, S_{a_m}\}$, S_{a_i} denotes all symptoms generated by action a_i. The cost of every action a_i is defined as $cost(a_i)$, $cost(a_i) > 0$. The MWSCP can be defined as finding a subset C of $\{S_{a_1}, S_{a_2}, \ldots, S_{a_m}\}$ to cover the symptoms in S with the minimum total cost $cost(a_i)$.

The MWSCP is an NP-complete problem and it is difficult to find the optimal solution. Thus, a heuristic greedy algorithm is designed in this paper to get the approximation. The covering cost rate can be defined as follows:

$$\eta(i) = \frac{|S_{a_i}|}{cost(a_i)}. \tag{6}$$

Among them, $|S_{a_i}|$ denotes the total number of symptoms generated by action a_i.

The steps of a heuristic greedy algorithm of MWSCP are described as follows:

(1) Save the symptoms that need to be generated in symptom set S_2.

(2) For one symptom $s_2(i)$ in S_2, select one action a_i with the highest covering cost rate, which can generate $s_2(i)$.

(3) Delete the symptoms that can be generated by a_i from S_2.

(4) Go on with these steps until S_2 is empty.

8. Simulation and analysis

8.1. Simulation environment and metrics

In this paper, the SN is composed of 300 nodes and about 1000 links. The initial available CPU and the bandwidth resources are real numbers following a uniform distribution between [50, 100]. The arrivals of VN requests follow a Poisson process with an average arrival rate of 5 per 100 time units. The lifetime of each VN request is modeled by an exponential distribution with an average of 1000 time units. The required CPU and bandwidth resources are real numbers uniformly distributed between [0, 5]. In addition, all virtual nodes have a constant position constraint value $D = 500$.

The number of negative symptoms follows a uniform distribution between [3, 6]. The virtual node faults and substrate node faults occur randomly. $p(s_i^{FVN} \mid fvf_i)$ follows a uniform distribution between [0.8, 1]. The prior probability of a faulty component is extremely low in the real network environment [16]. The prior probability of a virtual fault $p(fvf_i)$ follows a uniform distribution between [0.006, 0.008].

The computer used for the simulation experiments is a Lenovo Tianyi 510Pro with the Windows 10 operating system. The hardware platform is composed of an Intel Core i7-7700 3.6 GHz processor with 8 GB of RAM. The software is MATLAB R2007a. Each simulation is performed 50 times and we take the average values as the final results to eliminate the influence of random factors.

In this paper, the accuracy rate r_a and false positive rate r_{fp} are selected to estimate the performance of the algorithm. They are defined as follows:

$$r_a = \frac{|H \cap F|}{|F|}, \tag{7}$$

$$r_{fp} = \frac{|H \cap \overline{F}|}{|F|}. \tag{8}$$

H denotes a set of faults diagnosed by the algorithms. F represents a set of real faults that occurred in the network. $\overline{F}$ denotes the spurious faults diagnosed by the algorithms.

In this paper, our simulations focus on the five fault diagnosis algorithms listed in Table 8.1.

Table. Algorithm comparison.

Notation	Description
MCA	Fault diagnosis method proposed in [8]. The MCA is proposed by transforming the fault diagnosis in a bipartite fault propagation model into a set-covering problem. The substrate layer fault diagnosis method in this paper is added to the MCA.
ACMCA	Improved MCA. In ACMCA, the active detection technology is added to the MCA.
MLFD	Fault diagnosis method proposed in [10]. A layer-by-layer strategy and filtering algorithm are proposed to distinguish the different faults in the NVE.
ACMLFD	Improved MLFD algorithm. In the ACMLFD algorithm, the active detection technology is added to the MLFD algorithm.
SAHFD	The fault diagnosis method proposed in this paper. It integrates active detection with passive monitoring and can be divided into four functional modules: multifactor Bayesian hierarchical model, virtual layer fault diagnosis, substrate layer fault diagnosis, and active detection technology.

8.2. Simulation results

In Figure 5 and Figure 6, the performance of the SAHFD algorithm is optimal with different virtual fault numbers. The average accuracy rates of the SAHFD and ACMLFD algorithms are close to 0.95. The SAHFD algorithm achieves approximately 3.3%, 11.8%, and 18.8% higher accuracy rates than the MLFD, ACMCA, and MCA algorithms. The average false positive rate of the SAHFD algorithm is close to 0.19. The SAHFD algorithm achieves approximately 74.1%, 74.3%, 84.2%, and 85.7% lower false positive rates than the ACMLFD, MLFD, ACMCA, and MCA algorithms.

As can be seen from Figure 7 and Figure 8, with the increase of the substrate fault numbers, the performance of the SAHFD algorithm remains optimal. The average accuracy rate of the SAHFD algorithm
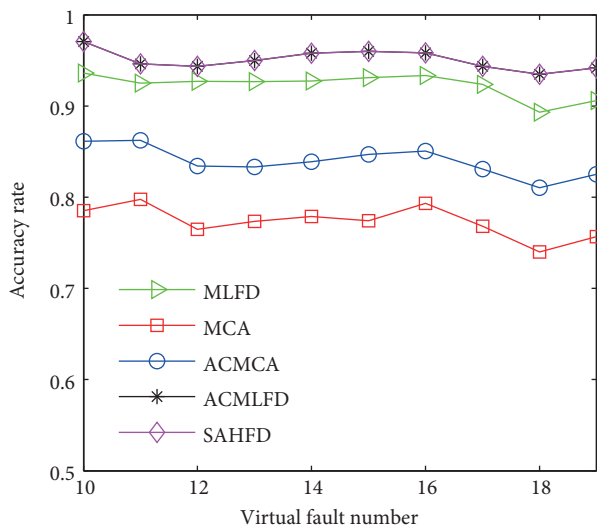

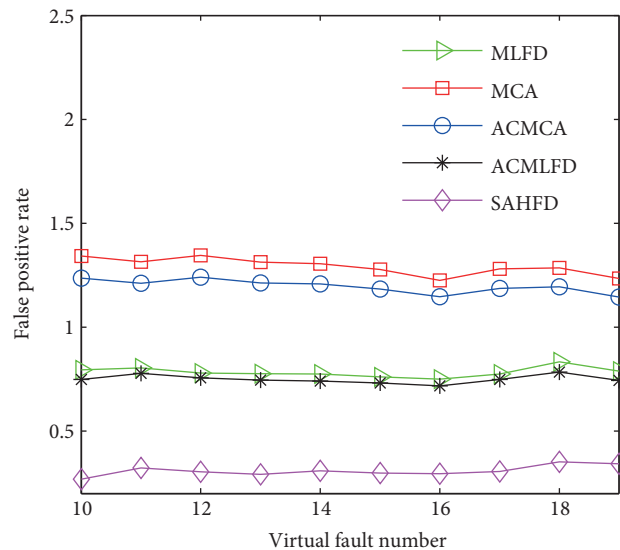**Figure 5**. Accuracy rate comparison with different virtual fault numbers.

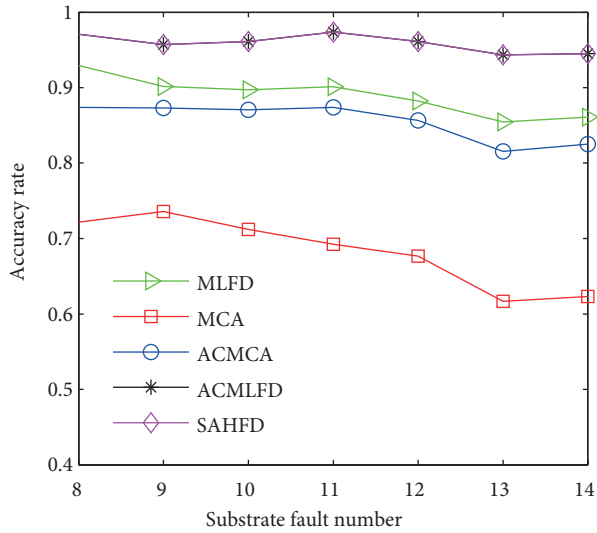**Figure 6**. False positive rate comparison with different virtual fault numbers.

Figure 7. Accuracy rate comparison with different substrate fault numbers.
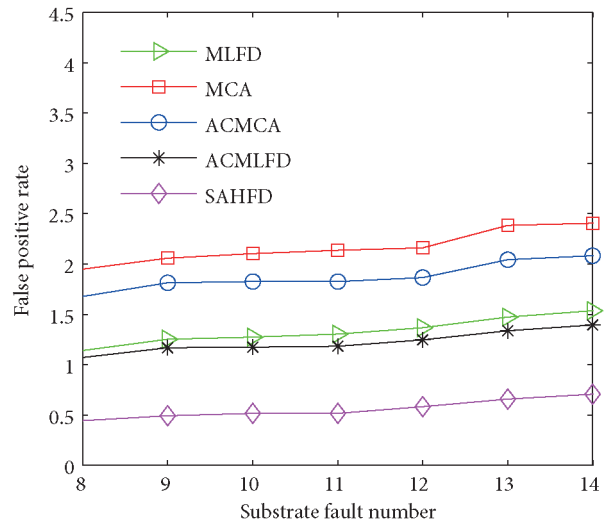

Figure 8. False positive rate comparison with different substrate fault numbers.

is close to 0.96. The accuracy rate of the ACMLFD algorithm is equal to that of the SAHFD algorithm. The SAHFD algorithm achieves approximately 9.1%, 14.3%, and 45.5% higher accuracy rates than the MLFD, ACMCA, and MCA algorithms. The average false positive rate of the SAHFD algorithm is close to 0.58. The SAHFD algorithm achieves approximately 54.7%, 55.4%, 65.9%, and 74.2% lower false positive rates than the ACMLFD, MLFD, ACMCA, and MCA algorithms.

As can be seen from Figure 9 and Figure 10, with the increase of the hybrid fault numbers, the performance of the SAHFD algorithm remains optimal. The average accuracy rate of the SAHFD algorithm is close to 0.95. The accuracy rate of the ACMLFD algorithm is equal to that of the SAHFD algorithm. The SAHFD algorithm
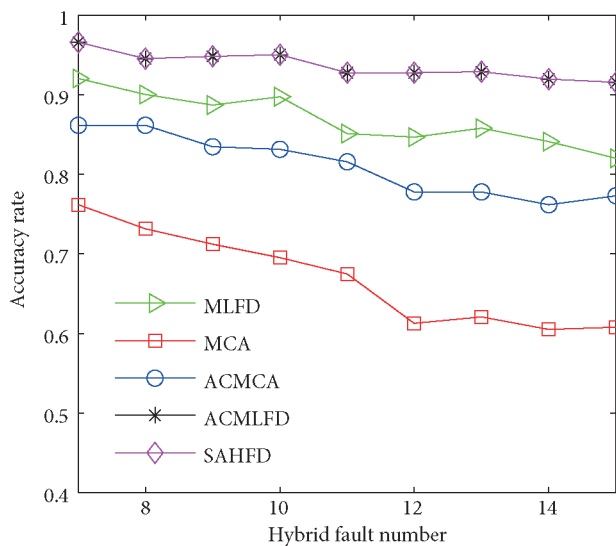

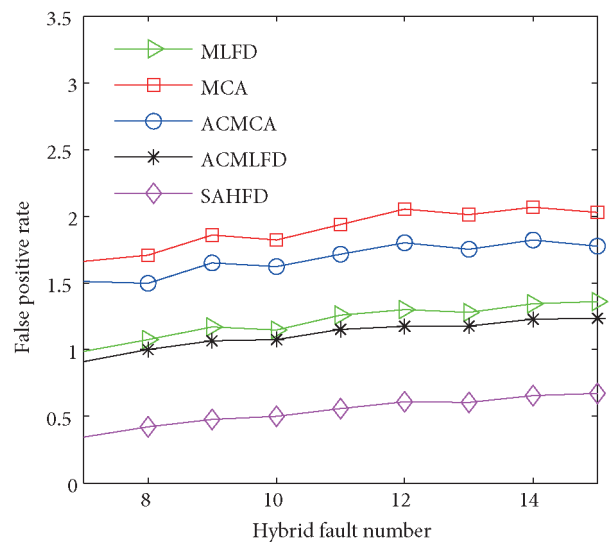**Figure 9**. Accuracy rate comparison with different hybrid fault numbers.

Figure 10. False positive rate comparison with different hybrid fault numbers.

achieves approximately 8.1%, 16.1%, and 42.4% higher accuracy rates than the MLFD, ACMCA, and MCA algorithms. The average false positive rate of the SAHFD algorithm is close to 0.51. The SAHFD algorithm achieves approximately 50.1%, 58.3%, 69.7%, and 72.9% lower false positive rates than the ACMLFD, MLFD, ACMCA, and MCA algorithms.

In Figure 11 and Figure 12, the performance of the SAHFD algorithm is optimal with different symptom loss rates. The average accuracy rates of the SAHFD and ACMLFD algorithms are close to 0.86. The SAHFD algorithm achieves approximately 2.4%, 6.1%, and 17.6% higher accuracy rates than the MLFD, ACMCA, and MCA algorithms. The average false positive rate of the SAHFD algorithm is close to 0.13. The SAHFD algorithm achieves approximately 43.5%, 56.7%, 69.8%, and 77.6% lower false positive rates than the ACMLFD, MLFD, ACMCA, and MCA algorithms.
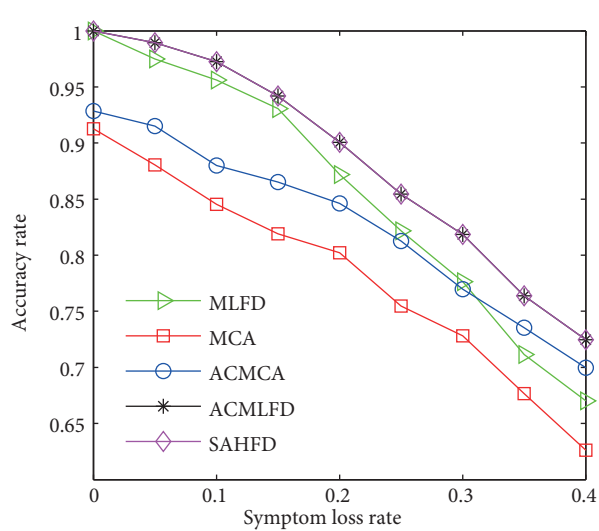

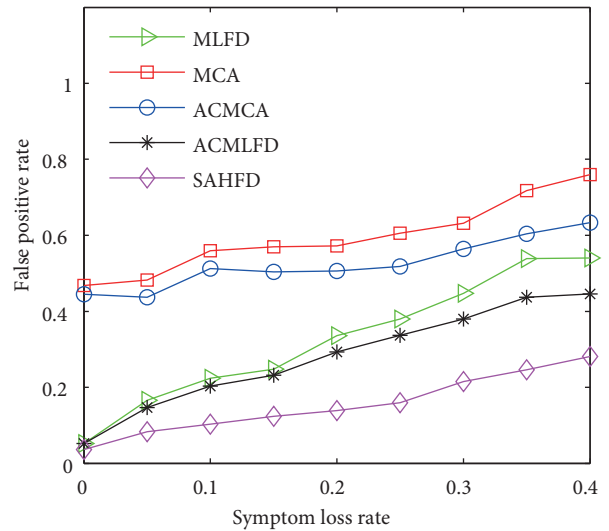**Figure 11**. Accuracy rate comparison with different symptom loss rates.

Figure 12. False positive rate comparison with different symptom loss rates.

As can be seen from Figure 13 and Figure 14, with the increase of the spurious symptom rates, the performance of the SAHFD algorithm remains optimal. The average accuracy rate of the SAHFD algorithm is close to 0.99. The accuracy rate of the ACMLFD algorithm is equal to that of the SAHFD algorithm. The SAHFD algorithm achieves approximately 2.1%, 8.8%, and 12.5% higher accuracy rates than the MLFD, ACMCA, and MCA algorithms. The average false positive rate of the SAHFD algorithm is close to 0.25. The SAHFD algorithm achieves approximately 67.1%, 67.1%, 80.8%, and 80.9% lower false positive rates than the ACMLFD, MLFD, ACMCA, and MCA algorithms.

8.3. Performance analysis

As can be seen from Figure 5 to Figure 14, the SAHFD algorithm proposed in this paper has the best performance. In the SAHFD algorithm, fault filtering is introduced to filter some spurious virtual faults, which improves the accuracy rate and narrows the scope of fault diagnosis. In virtual layer fault location, the contribution degree is used instead of the observed symptom number. In some situations, the fault does occur. However, the number of corresponding symptoms is small and it cannot be detected. Hence, the introduction of contribution degree can improve the performance of the SAHFD algorithm. The SAHFD algorithm filters
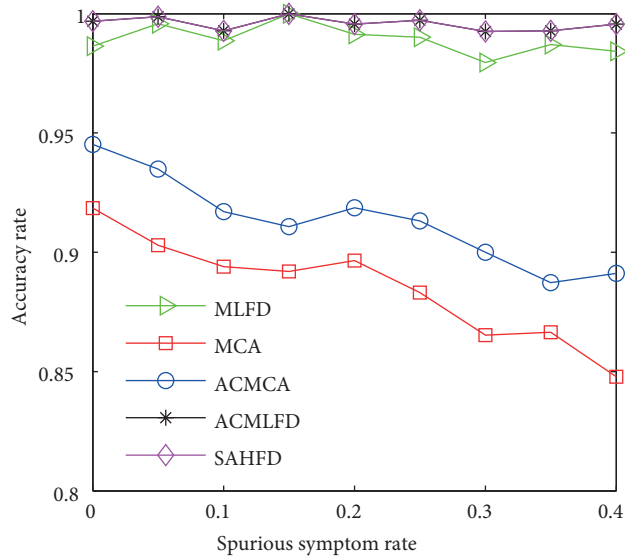
Figure 13. Accuracy rate comparison with different spurious symptom rates.
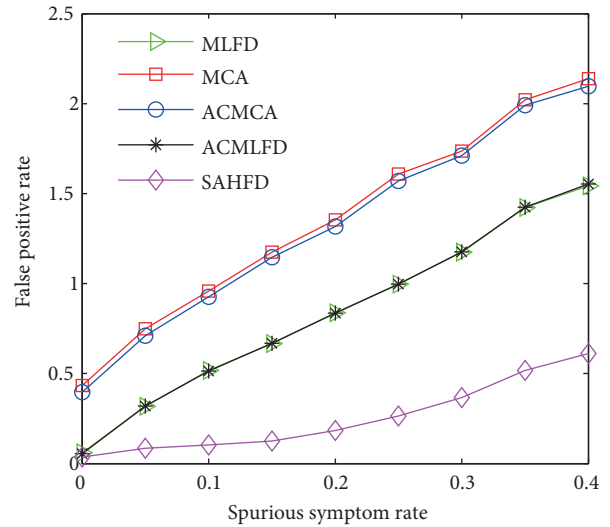

Figure 14. False positive rate comparison with different spurious symptom rates.

some spurious faults with the help of the confidence degree threshold in the early detection phase and the false positive rates are reduced. The SAHFD algorithm also introduces active detection to diagnose the suspected virtual layer faults with low contribution degree and its performance is the best.

In addition, the possible performance losses of the SAHFD algorithm are closely related to network noise. If the network noise is large, more spurious symptoms will be added to the symptom set. Although active detection is introduced to improve the performance, it cannot eliminate the influence of network noise completely.

In ACMLFD and ACMCA, active detection is introduced to detect the suspected network components by setting a suitable correlation degree threshold, which increases the accuracy rates and decreases the false positive rates. The layer-by-layer strategy and filtering algorithm are introduced into the MLFD algorithm and its performance is better than that of MCA.

8.4. Complexity analysis

The SAHFD algorithm includes virtual layer fault diagnosis and substrate layer fault diagnosis. In virtual layer fault diagnosis, the complexity of fault filtering is $O(|VF_1|)$, where $|VF_1|$ is the number of all suspected virtual faults. The complexity of virtual layer fault location is $O(|FVF|)$, where $|FVF|$ is the number of suspected virtual layer faults after fault filtering. The complexity of spurious fault detection is $O(\rho \cdot |VF_2|)$, where ρ is the spurious fault detection threshold and $|VF_2|$ is the number of suspected virtual layer faults after virtual layer fault location. In substrate layer fault diagnosis, the complexity is $O(|SF_1| + |LVF|)$, where $|SF_1|$ is the number of suspected substrate faults and $|LVF|$ is the number of suspected lost virtual components that need to be detected by active detection method. Hence, the total complexity of the SAHFD algorithm is $O(|VF_1| + |FVF| + \rho \cdot |VF_2| + |SF_1| + |LVF|)$.

The complexity of MCA is $O(|S_1|^2 \cdot |VF_1| + |SF_2|)$, where $|S_1|$ is the number of the symptoms and $|SF_2|$ is the number of suspected substrate faults. The complexity of ACMCA is $O(|S_1|^2 \cdot |VF_1| + |SF_2| + |AC_1|)$,

where $|AC_1|$ is the number of virtual components that need to be detected by active detection method in ACMCA. The complexity of the MLFD algorithm is $O(|VF_1| + |FVF_1| + |SF_3|)$, where $|FVF_1|$ is the number of suspected virtual layer faults and $|SF_3|$ is the number of suspected substrate faults. The complexity of the ACMLFD algorithm is $O(|VF_1| + |FVF_1| + |SF_3| + |AC_2|)$, where $|AC_2|$ is the number of virtual components that need to be detected by active detection method in the ACMLFD algorithm.

9. Conclusions

In this paper, we present a novel technique called the symptom-aware hybrid fault diagnosis algorithm or SAHFD algorithm in the NVE. The SAHFD algorithm is designed to overcome network noise by integrating passive monitoring with active detection. A multifactor Bayesian hierarchical model is proposed for the Bayesian network. Then fault filtering, confidence degree ranking, and active detection are introduced to filter the spurious faults in the virtual layer. In substrate layer fault diagnosis, the active detection algorithm is introduced to improve the accuracy of fault diagnosis. Finally, we propose a heuristic greedy algorithm to select the suitable actions with minimum total cost. The results of our simulation show that, compared with other algorithms, the SAHFD algorithm has the best performance in different fault types, different symptom loss rates, and spurious symptom rates.

Acknowledgment

References

[1] Morales F, Ruiz M, Velasco L. Virtual network topology reconfiguration based on big data analytics for traffic prediction. J Opt Commun Netw 2017; 9: 35-45.

[2] Blenk A, Basta A, Reisslein M, Kellerer W. Survey on network virtualization hypervisors for software defined networking. IEEE Commun Surv Tut 2017; 18: 655-685.

[3] Kou TW, Liou BH, Lin KC, Tsai MJ. Deploying chains of virtual network functions: on the relation between link and server usage. IEEE ACM T Network 2018; 26: 1-15.

[4] Su Y, Meng X, Kang Q, Han X. Survivable virtual network link protection method based on network coding and protection circuit. IEEE Access 2018; 6: 67477-67493.

[5] Dusia A, Sethi A. Recent advances in fault localization in computer networks. IEEE Commun Surv Tut 2016; 18: 3030-3051.

[6] Li RTH, Chung SH. Digital boundary controller for single-phase grid-connected CSI. In: IEEE 2008 Power Electronics Specialists Conference; Rhodes, Greece. New York, NY, USA: IEEE; 2018. pp. 4562-4568.

[7] Boem F, Ferrari RMG, Keliris C, Parisini T, Polycarpou MM. A distributed networked approach for fault detection of large-scale systems. IEEE T Automat Contr 2016; 62: 18-33.

[8] Huang XH, Zou SH, Wang WD, Cheng SD. Fault management for Internet service: Modeling and algorithms. In: IEEE International Conference on Communications; İstanbul, Turkey. New York, NY, USA: IEEE; 2006. pp. 854-859.

[9] Zhang S, Qiu X, Meng L. Service fault diagnosis algorithm in network virtualization environment. J Software 2018; 23: 2772-2781.

[10] Yan CX, Wang Y, Qiu XS, Li WJ, Guan L. Multi-layer fault diagnosis method in the network virtualization environment. In: Sixteenth Asia-Pacific Network Operations and Management Symposium; Taipei, China. New York, NY, USA: IEEE; 2014. pp. 1-6.

[11] Liu N, Zhang S, Wang X. Virtual network fault diagnosis using trust evaluation. Video Eng 2016; 40: 80-84.

[12] Gillani S, Demirci M, Al-Shaer E, Ammar M. Problem localization and quantification using formal evidential reasoning for virtual networks. IEEE T Netw Serv Man 2014; 11: 307-320.

[13] Wang H, Wang Y, Qiu XS, Li WJ, Xiao AL. Fault diagnosis based on evidences screening in virtual network. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management; Ottawa, Canada. New York, NY, USA: IEEE; 2015. pp. 802-805.

[14] Bennacer L, Amirat Y, Chibani A, Mellouk A, Ciavaglia L. Self-diagnosis technique for virtual private networks combining Bayesian networks and case-based reasoning. IEEE T Autom Sci Eng 2015; 12: 354-366.

[15] Zhang L, Zhu XR, Zhao S, Xu D. A novel virtual network fault diagnosis method based on long short-term memory neural networks. In: 2017 IEEE 86th Vehicular Technology Conference; Toronto, Canada. New York, NY, USA: IEEE; 2017. pp. 1-5.

[16] Mehmood A, Alrajeh N, Mukherjee M, Abdullah S, Song H. A survey on proactive, active and passive fault diagnosis protocols for WSNs: network operation perspective. Sensors 2018; 18: 1787.

[17] Liu S, Wang Y. A dynamic probe station selection algorithm in the network virtualization environment. J Beijing U Posts Telecommun 2016; 39: 1-5.

[18] Pan YL, Qiu XS, Zhang SL. Fault diagnosis in network virtualization environment. In: Eighteenth International Conference on Telecommunication; Ayia Napa, Cyprus. New York, NY, USA: IEEE; 2011. pp. 517-522.

[19] Tang Y, Shaer E, Boutaba R. Efficient fault diagnosis using incremental alarm correlation and active investigation for Internet and overlay networks. IEEE T Netw Serv Man 2008; 5: 36-49.