# A multi-fault recovery oriented deployment strategy of backup controllers in SDN

Wenwen Zhao[a,b], Xiangru Meng[a], Qiaoyan Kang[a*], Dong Zhai[b], Yong Yang[b]
[a] Information and Navigation College, Air Force Engineering University, Xi'an, Shaanxi, China;
[b] Unit 69235 of the PLA, Wusu, Xinjiang, China

## ABSTRACT

In Software Defined Network (SDN), the control network is a logically mapped network that is like the brain and control center of the entire network. The reliability of the control network is the ability to survive failures of switching nodes on a large scale. However, there are few studies that consider the control network of SDN as a whole. In this work, we took the SDN control network as a standalone network that is separate from the underlying switching network. We developed a model to evaluate the average reliability of the control network in the presence of multiple failures of the underlying switching networks based on its topology characteristics. Then we searched for the optimal solutions of the model through an opposite learning revised sparrow search algorithm, based on which we proposed a multiple fault recovery oriented deployment strategy of backup controllers in SDN. The simulated experiment shows that the revision of the algorithm is effective and our proposed strategy could help us increase the reliability of the control network to a significant extent.

**Keywords:** SDN, Multi-fault recovery, controllers' deployment, reliability

## 1. INTRODUCTION

The control network of Software Defined Network (SDN), which is logically mapped over the underlying switching networks by specific mapping rules, not only has its own attributes but also depends heavily on the underlying network[1]. The performance metrics of the control network such as time delay, bandwidth, load and reliability are directly affected by the underlying network. However, the SDN control network is logically independent of its underlying network and the allocation and adaptation of the control network is a very spontaneous and dynamic process.

As shown in Figure 1, there is a possibility that switching nodes in the underlying network may fail due to a poor manufacturing process, a particular environment, an emergency, or other reasons. When the underlying switching nodes fail, it causes a disruption in the control network and affects the reliability of the SDN. Currently, research on the reliability of the control network in SDN caused by the failure of switching nodes mainly focuses on the synchronization and consistency of control information. The question of how to build a reliable control network mainly focuses on the reliability of the controllers themselves. There are few studies that have considered the control network as a whole, and the study of the influence of the location of backup controllers in the underlying network on the reliability of the control network is still insufficient. In this paper, the SDN control network as a whole is considered and how backup controllers can be deployed in the event of a switching node failure is investigated. In addition, automatic dynamic adaptation of the control network is carried out according to the previously established principles and relevant strategies to adapt to the changes and development requirements of the network situation and tasks at any time.

To the best of our knowledge, multi-controller failure in SDN has not been addressed yet; therefore, in this section we present a summary of representative studies on SDN resilience. Kiadehi[2] proposed a recovery scheme that computed redundant paths as main and backup paths with no overlap between network devices by creating a mathematical model called Shared Risk Link Group (SRLG) and reduced failure recovery time and packet loss. Jalili[3] discussed the deployment mode of SDN control networks and studied the impact of different control modes on availability and cost, security, link failure, and boot time from two aspects: In-band and Out-of-band. Canini[4] proposed a self-organizing fault-tolerant algorithm for the in-band control mode, that is, according to the changes in the network (such as switch or link failure, high packet loss rate or excessive delay, controller increase or decrease, etc.), the self-organizing method is

[*] kgdkqy@163.com

Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)
edited by Yingfa Lu, Changbo Cheng, Proc. of SPIE Vol. 12506, 125061M
© 2022 SPIE · 0277-786X · doi: 10.1117/12.2662519

used to maintain the relationship between the control plane and the data plane. Schiff[5] proposed boot method for the control network that automatically ensures that the switch is correctly routed to the appropriate controller when the controller is added or removed, and ensures the integrity of the control network and supports hot plug. Kiadehi[6] constructed a control network with high reliability and survivability and proposed appropriate solutions by employing reliability flooding, global snapshot, and fast establishment of the shared global view of the control network. Das[7] proposed an interactive network deployment model for the control plane and the data plane, and studied the impact of the model on the synchronous interactive information between the controller and the switch when the controller node fails. Xiang and Yu[8] proposed a method to quickly recover the control network and switch fault migration after an underlying failure by deploying the controllers in a distributed ring. Li[9] used a heuristic algorithm to solve the Byzantine fault-tolerant problem in the control network in SDN. Gu[10] proposed to solve the single-point vulnerability problem in SDN networks by setting up a dynamic architecture for deploying redundant controllers for possible large-scale faults in SDN. Hirayama[11] proposed a method to build a robust control plane based on robust topology coefficients and also proposed a method to quickly reconstruct the distributed SDN control plane in case of emergency. Savas[12] solved the elasticity problem of the control plane by using the virtual network mapping technology to reduce the communication interruption between the control plane and the data plane caused by the failure of the underlying physical network (switch network). For the node protection problem of routing forwarding number in SDN deployed by a single controller, Beheshti[13] proposed the weights to evaluate the reliability of nodes and routing forwarding tree, and proposed a heuristic algorithm and a traversal algorithm to solve the influence of different controller deployment modes on the weights of routing forwarding tree.

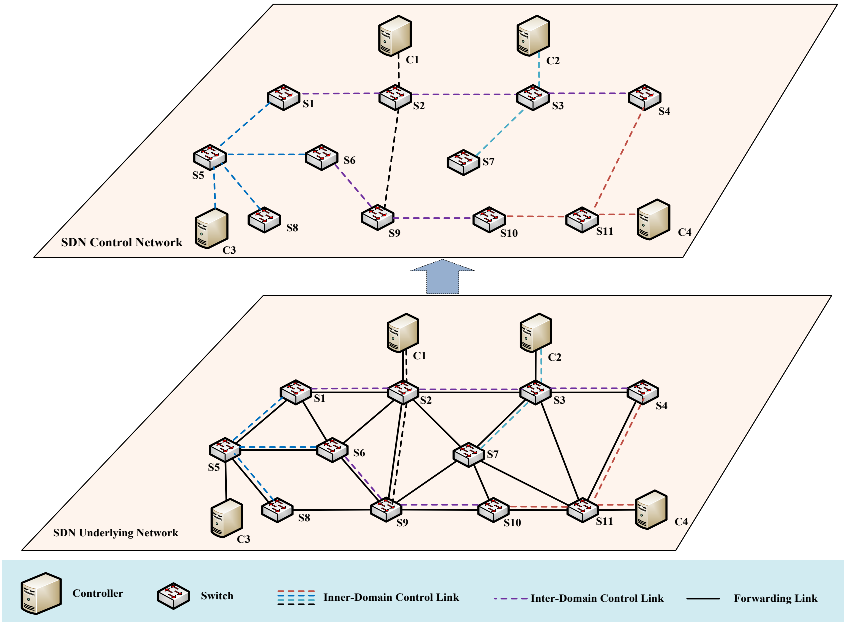

Figure 1. Sketch map of SDN control network.

## 2. PROBLEM DESCRIPTION AND AVERAGE NETWORK RELIABILITY EVALUATION MODEL FOR MULTIPLE FAULTS

### 2.1 Problem description

Currently, there are mainly two solutions for SDN controller network failure in the form of backup and recovery protection, namely active protection as shown in Figure 2, and passive protection in Figure 3[14]. Active protection means that all underlying switches maintain a control path with all controllers and each controller sends the same control instructions to all switches, each of which selects one of the controllers as its corresponding authorized controller through a certain strategy. In the event of a controller or switch failure that may cause the control network to fail, the protection strategy can 'automatically' ensure that at least one controller of each switch is connected to it.

Accordingly, passive protection means that each switch maintains a control path with only one of the controllers, which can be referred to as an authorized controller. The authorized controller interacts with the switch and synchronously copies the interactive information to the backup controller. At this time, if the main authorized controller fails and causes the control network failure, the switch 'passively' switches its authorized controller to the backup controller. Passive protection can also be enhanced by adding an additional shared data server outside the control network. In this case, the information between the main authorized controller and the corresponding switch does not need to be synchronized with the backup controller, but is stored on the shared database server. If the main authorized controller fails, the backup controller receives this information from the server and restores control of the 'lost' switch. This method eliminates the need to exchange information between controllers and individual backup controllers, significantly reduces traffic between the controller and its backup, and effectively improves backup efficiency.

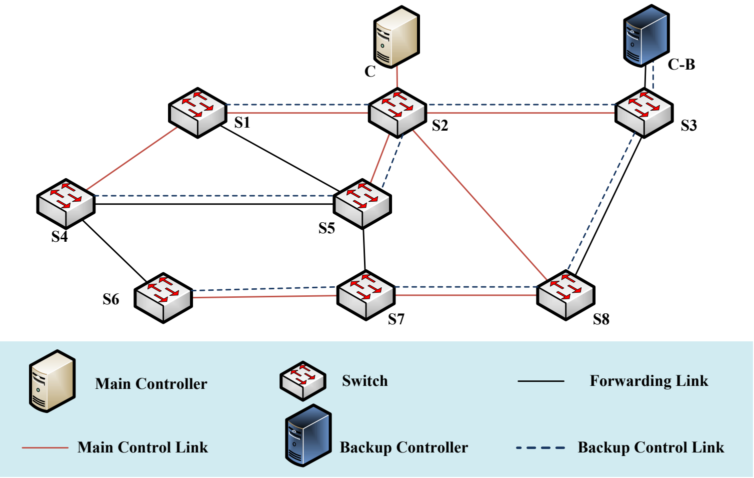

Figure 2. Active protection.

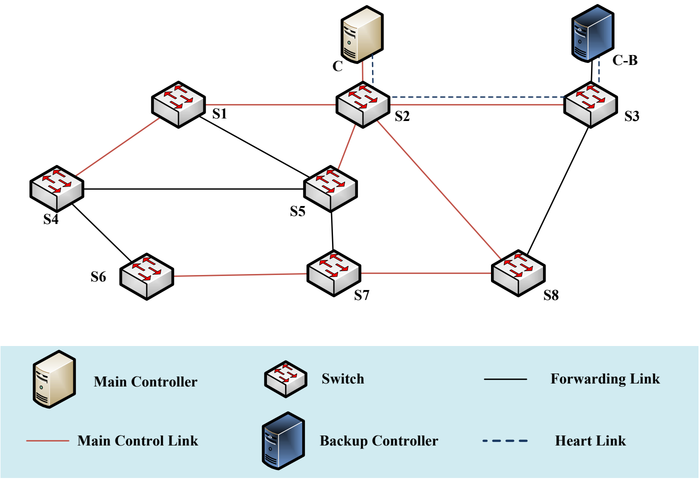

Figure 3. Passive protection.

In this paper, we consider a passive protection method in which the deployment location of the backup controller is selected according to the number of backup controllers when the failure of the underlying switching network causes a chain reaction of disruption in the control network to ensure that the reliability of the control network is guaranteed and realized to a certain degree. The selection of the number of backup controllers is mainly based on the reliability index and the cost of building links and nodes required for the backup controller. In this paper, we assume that the number of controllers is already fixed.

For the underlying SDN switching network, we first deploy different controllers according to a certain deployment strategy in different application scenarios without considering the controller backup, and divide the whole network into a certain number of control domains. Considering possible multiple failures of the underlying switching network, we then build the reliability model of the backup controller deployment network to evaluate the recovery strategy. Currently, the common recovery model indicators for multiple failures in the network are node degree, betweenness centrality, and adjacency centrality. These indicators suggest the importance of controlling network nodes from the network topology perspective. With this in mind, we examine the protection strategies of each network node for potential failures.

### 2.2 Average network reliability evaluation model for multiple faults

Based on the definition of network topology proposed[15], we present a strategy for the case when one or more switching nodes fail.

In this regard, the symbol *F* is used to represent the node failure, and $F \subseteq S$. In case of failure *F*, all nodes *s* in *F* will be deleted from the original network *G*, and the edges and control nodes connected to *s* will also be deleted. Therefore, the network after failure *F* can be expressed as $g(F) = (S(F), E(F))$, which is the largest subgraph of *G* after being divided by fault *F*, where *S*(*F*)=*S*\*F*, and $E(F) = E \cap \mathrm{S}(\mathrm{F})$. Each fault *F* divides G into several "islands" *I*. the set formed by all islands is expressed as $\ell(F)$, for each specific island *I*, if node *s* is directly affected by fault *F*, then $I(F,s)=\varnothing$, where *s* is the node in island *I* and *I*(*F,s*) is the island formed by fault *F*. Based on the above network and the constructed model, we proposed a corresponding model to evaluate the network topology (Figure 4).

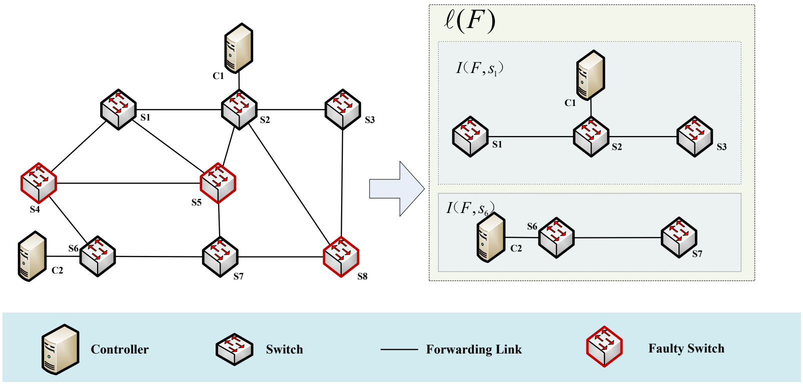

Figure 4. Island and Island collection.

2.2.1 Network Topology Evaluation Model for Multiple Faults. For fault *F*, the following model has been defined in this document to evaluate the reliability of the network topology.

$$INF(F) = \sum_{I\in\ell(F)} |I|(|I|-1) \tag{1}$$

$$INF(F) =| \ell(F)| \tag{2}$$

$$INF(F) = \max\{|I|,\ I \in \ell(F)\} \tag{3}$$

Equation (1) represents the number of nodes surviving fault *F*, equation (2) represents the number of islands caused by fault *F*, and equation (3) represents the size of the largest island caused by *F*. These models are used to evaluate the reliability of the network topology exposed to *F*. Further analysis shows that the above equation (1) can also be described by the average reliability of two-terminal reliability (as equation (4)):

$$\mathrm{AR}(F) = \frac{1}{|S|(|S|-1)} \sum_{I \in \ell(F)} |I|(|I|-1) \tag{4}$$

2.2.2 Network Reliability Model for Multiple Faults. Based on the previous network topology evaluation model, this paper proposes a model to assess the average reliability of a network:

$$\mathrm{NR}(F) = \frac{\sum_{I \in \ell(F)} |I| \cdot \mathrm{Coef}(F,I)}{|S|} \times 100\% \tag{5}$$

where $\mathrm{Coef}(F,I)$ is a binomial coefficient. If there is at least one controller in island *I*, $\mathrm{Coef}(F,I)=1$, otherwise $\mathrm{Coef}(F,I)=0$. Therefore, NR (*F*) represents the fraction of switching nodes where at least one controller is connected in the network after error *F*.

Based on NR (F), the average network reliability model $\mathrm{ANR}(\Gamma)$ for the set $\Gamma$ consisting of all faults *F* as elements can be defined as follow:

$$\mathrm{ANR}(\Gamma) = \frac{1}{|\Gamma|} \sum_{F \in \Gamma} NR(F) \tag{6}$$

2.2.3 Controller Reliable Deployment Model for Multiple Faults. Based on the above model for evaluating average network reliability, another model for controller deployment reliability has been proposed in this paper, which consists of finding an appropriate deployment scheme for the total number of *M* controllers to maximize the value of equation (6), as shown in equation (7):

$$\begin{gathered} \max\{\mathrm{ANR}(\Gamma)\} = \max\{\frac{1}{|\Gamma|} \sum_{F \in \Gamma} NR(F)\} \ , \\ \sum_{c \in S'} x_c \le M \ , \ \mathrm{Coef}(F,I) \le 1 \ , \ x_c \in \{0,1\} \ , \ I \in \ell(F) \ , \ F \in \Gamma \end{gathered} \tag{7}$$

## 3. DEPLOYMENT STRATEGY OF BACKUP CONTROLLERS BASED ON PG-OLSSA

### 3.1 Population guided inversed optimization sparrow search algorithm (PG-OLSSA)

Based on the models presented in the previous sections, we propose a strategy for the reliable deployment of controllers to ensure the continuity of services in case of node failures. For the numerous failures that may occur in the underlying switches, assuming that the SDN controllers have been well deployed and the number of backup controllers is given, we search for the deployment location of the backup controllers to optimize the reliability model of the network. To find the best location for deploying these backup controllers, the Population-Guided Opposite Learning Sparrow Search Algorithm (PG-OLSSA) is used to find a better solution set for the model.

As mentioned earlier, the Sparrow Search Algorithm (SSA) has the advantages of high optimization speed, low complexity, and high comprehensibility when dealing with NP-hard problems[16]. Nevertheless, it can easily fall into a local optimum when applied separately. In view of these shortcomings, the mechanism of Opposite Learning in SSA(OLSSA) was introduced[17] to extend the optimization range and prevent the method from falling into the trap of local optimum. In this work, the optimization capability of OLSSA is further improved by using the idea of population guidance[18] after considering the idea of opposite learning in the initialization of the algorithm.

The OLSSA is a revised SSA with the idea of reverse learning in the initialization phase and the population-guided procedure: The traditional sparrow search algorithm usually needs the “leading sparrows” (optimal solution) in the current best position to guide the other sparrows to move to the optimal destination to solve the unreasonable optimal solution problem caused by randomness. Therefore, the leader selection method in the current step is crucial to obtain the global optimal solution. To select the leading solutions for population management, we need to divide the whole population into different levels of Pareto fronts by non-dominant sorting. Then, we can determine the Pareto level by calculating the degree of congestion of each solution according to equation (8), and then sort all these solutions to select the best N solutions for population guidance. The congestion distance refers to the sum of the distances between a Pareto solution and its nearest Pareto solution in each objective function dimension. The larger the congestion distance, the farther it is from the current nearest Pareto solution, i.e., the Pareto solution is sparser; otherwise, it means that the position of the solution is closer to the current nearest Pareto solution and the Pareto solution is denser. By introducing the congestion distance, we can select the population formed by the optimal Pareto solution from the distribution of the Pareto solution and then guide the population find a more uniform solution set at the Pareto front.

$$S[i]_{dis} = \sum_{m=1}^{M} \frac{S[i+1]_{(m)} - S[i-1]_{(m)}}{f_m^{\max} - f_m^{\min}} \qquad (8)$$

$m$ stands for the dimension of the objective function, $S[i-1]_{(m)}$ means that after being sorted, the value of the current solution is worse than the value of the $i$th solution of the objective function, and accordingly $S[i+1]_{(m)}$ stands for the value of the objective function that is only better than the $i$th solution. $f_m^{\max}$ and $f_m^{\min}$ stand for the maximum and minimum value of the $m$th dimensional objective function, respectively.

### 3.2 Deployment strategy process of backup controller for multiple fault recovery

Based on the elements presented in the preceding sections, the backup controller's multiple failure recovery deployment strategy process consists of a six-step mechanism to solve the target problems in Section 2 as shown in Figure 5, namely:

- Initialize the population by the reverse learning mechanism and calculate the fitness value of the initial population according to equation (6).

- Based on the fitness value, the first 10% of the leader's optimal solution is used as the guide to update the position of the sparrows, obtaining the updated guide population by the reverse learning mechanism.

- Calculate the new fitness value of the updated population, as well as the Pareto level and the congestion distance.

- If the Pareto level is the same, choose the solution with the larger congestion distance.

- Following the rule above, the top 10% is selected as leaders for the next round, and the followers follow and complete the location updating.

- Continue the above cycle until the algorithm reaches the exit condition.

During this time, the vigilante performs the monitoring task according to the rules[19].

## 4. EXPERIMENTAL SIMULATION AND ANALYSIS

### 4.1 Experimental environment

Due to the known limitations in simulated environments regarding SDN properties. This study uses a simulated network on the MATLAB platform[15] to verify the functionality and effectiveness of our proposed strategy. The parameters of the network and network flows, such as switching nodes, physical connection, number of unknown flows, rated load, and controller load, have already been described[15]. Based on this, the underlying failure of the switching node is assumed to obey the Poisson distribution as a parameter $\lambda=3$ . For the variable initialization of the algorithm PG-OLSSA, we set the maximum number of iterations MAXINTERATION = 15 and the population size POP_SIZE = 50, the proportion of leaders PL = 0.4, the proportion of followers PF =1- PL =0.6, and the proportion of vigilantes PA =0.1. Among the population as leaders, set the proportion of selected guides as GPL = 20%.

### 4.2 Simulation results and analysis

Based on the results of the simulated network and controller distribution shown in Figure 6, this paper uses the PG-OLSSA algorithm to deploy the backup controller. The results are shown in Figure 6 below. It can be seen that based on Figure 6b, the black node in Figure 6c is the deployment site of the backup controllers, which is to further improve the reliability and security properties of the control network.

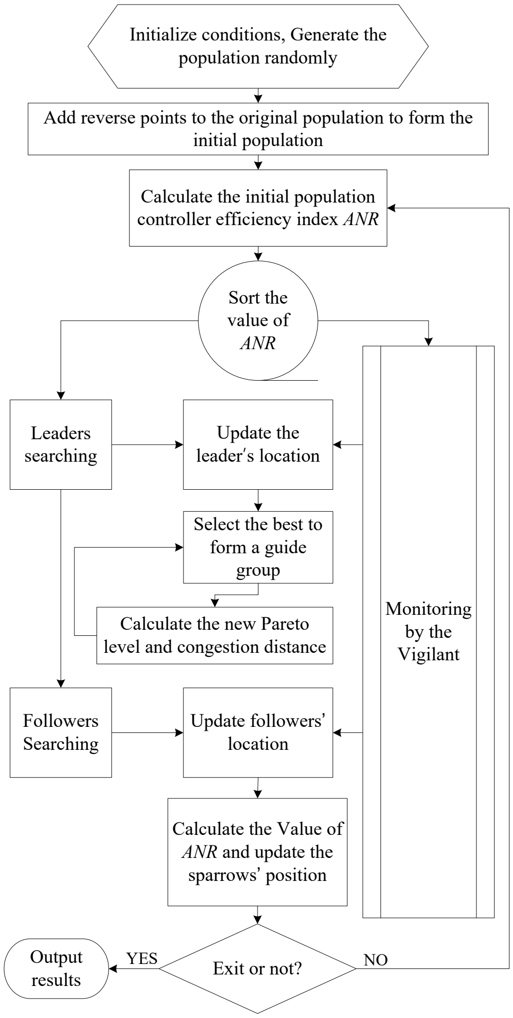
Initialize conditions, Generate the population randomly
Add reverse points to the original population to form the initial population
Calculate the initial population controller efficiency index *ANR*
Sort the value of *ANR*
Leaders searching
Update the leader's location
Select the best to form a guide group
Calculate the new Pareto level and congestion distance
Monitoring by the Vigilant
Followers Searching
Update followers' location
Calculate the Value of *ANR* and update the sparrows' position
Exit or not?
YES
NO
Output results

Figure 5. PG-OLSSA strategy flow chart.

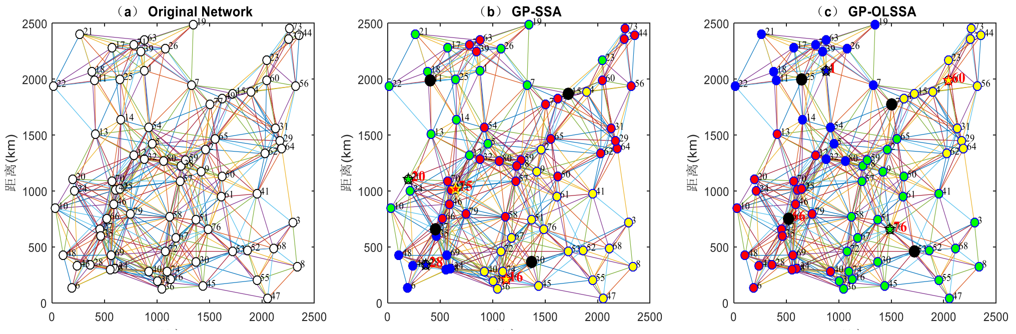

Figure 6. Deployment result of simulated network and backup controller.

Figure 7 shows the comparison between three algorithms: Standard SSA[19], OLSSA[17] and PG-OLSSA proposed in this paper. It can be seen that the algorithm PG-OLSSA has been optimized to some extent according to the reverse learning and population guided optimization in terms of optimization speed and optimization ability.

In terms of the reliability index described in equation (6), this paper compares the reliability index obtained from PG-OLSSA and OLSSA, respectively. The results are shown in Figure 8. As you can see, the reliability index obtained by PG-OLSSA is much higher than that of OLSSA, which means that the reliability index proposed in this paper is effective.

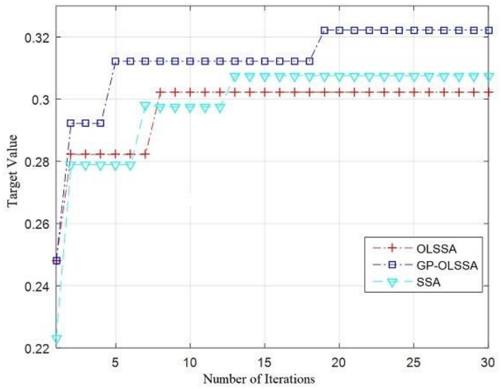

Figure 7. Comparison of optimization curves.

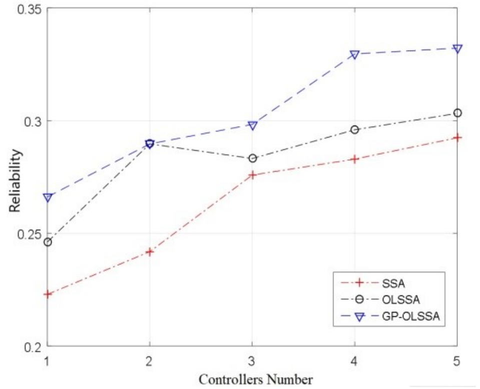

Figure 8. Reliability index comparison.

## 5. CONCLUSION

Based on the reliability of the SDN control network and the formation of the control network, this paper proposes a reliable deployment strategy for the multi-backup controller under multiple failures. Simulation experiments show that the proposed strategy is an effective solution to ensure high reliability of the control network in the presence of unknown faults, and has some practical significance in improving the overall reliability of the SDN network.

## REFERENCES

[1] Schiff, L., Schmid, S. and Canini, M., “Medieval: Towards a self-stabilizing, plug & play, in-band sdn control network,” ACM Sigcomm Symposium on SDN Research, 132-141(2015).

[2] Kiadehi, K. B., Rahmani, A. M. and Molahosseini, A. S., “A fault-tolerant architecture for internet-of-things based on software-defined networks,” Telecommunication Systems, 6, 155-169(2021).

[3] Jalili, A., “A comprehensive analysis on control plane deployment in SDN: In-band versus out-of-band solutions,” 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI) IEEE, 125-133(2017).

[4] Canini, M., “A Self-Organizing Distributed and In-Band SDN Control Plane,” 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), IEEE, (2017).

[5] Schiff, L., Schmid, S. and Canini, M., “Ground control to major faults: Towards a fault tolerant and adaptive SDN control network,” IEEE/IFIP International Conference on Dependable Systems & Networks Workshop, IEEE, (90-96)2016.

[6] Khondoker, R., Zaalouk, A., Marx, R. and Bayarou, K., “Feature-based comparison and selection of Software Defined Networking (SDN) controllers,” 2014 World Congress on Computer Applications and Information Systems (WCCAIS), 1-7(2014)

[7] Das, R. K., “FT-SDN: A fault-tolerant distributed architecture for software defined network,” Wireless Personal Communications, 114(4), (2020).

[8] Xiang, B., Yu, L., “Research and design on SDN multi-controller fault tolerance,” Computer Engineering and Applications, 54(23), 81-88(2018).

[9] Li, J., Hu, Y. and Wu, J., “Research on improving the control plane’s reliability in SDN based on byzantine fault-tolerance,” Journal of Computer Research and Development, 54(5), 952-960(2017).

[10] Gu, Z., Zhang, X. and Lin, S., “Research on security mechanism for SDN control layer based on mimic defense theory,” Application Research of Computers, 035(007), 2148-2152(2018).

[11] Hirayama, T., Jibiki, M. and Harai, H. “Designing distributed SDN C-plane considering large-scale disruption and restoration,” The Institute of Electronics, Information and Communication Engineers, 3, 1-13(2019).

[12] Savas, S. S., “Disaster-resilient control plane design and mapping in software-defined networks,” IEEE 16th International Conference on High Performance Switching and Routing, 1-6(2015).

[13] Beheshti, N., and Ying, Z., “Fast failover for control traffic in software-defined networks,” Global Communications Conference, IEEE, 2665-2670(2013).

[14] Zhu, Z., Lin, Q., Xu, M., et al., "A customized and cost-efficient backup scheme in software-defined networks," 2017 IEEE 25th International Conference on Network Protocols (ICNP), 1-6(2017).

[15] Zhao, W. W., Meng, X. R., Kang, Q. Y., et al., "A delay and reliability aware multi-controller balancing deployment strategy," Journal of Air Force Engineering University (Natural Science Edition), 22(4), 85-91(2021).

[16] Heller, B., Sherwood, R. and Mckeown, N., "The controller placement problem," ACM Sigcomm Computer Communication Review, 42(4), 473-478(2012).

[17] Tizhoosh, H. R. "Opposition-based learning: A new scheme for machine intelligence," International Conference on International Conference on Computational Intelligence for Modelling, Control & Automation, IEEE, 695-701(2005).

[18] Wang, C., et al., "Multi-objective grasshopper optimization algorithm based on multi-group and co-evolution," Mathematical Biosciences and Engineering (MBE) 18(3), 2527-2561(2021).

[19] Xue, J. K. and Shen, B., "A novel swarm intelligence optimization approach: Sparrow search algorithm," Systems Science & Control Engineering, 8(1), 22-34(2020).