The Implementation and Evaluation of High-Speed Link Monitoring Tool for Supercomputer

Jiaqing Xu[(✉)], Jie He, Xiaotao Hu, Jijun Cao, Lei Zhang, and Chongfeng Wang

School of Computer, National University of Defense Technology, Changsha, China
xujiaqing@nudt.edu.cn

Abstract. With the increase of system scale and link speed, the link failure has become the most important type of interconnect fault in supercomputers, which has brought great challenges to the maintenance of high-performance interconnect networks. In order to meet the needs of operation and maintenance personnel to monitor the status and performance of all high-speed links of supercomputer in real-time, this paper designs a high-speed link monitoring tool based on in-band network, which has good scalability and robustness for real-time monitoring of high-speed link status and performance information. The tool has been practically utilized in the operation and maintenance of domestic supercomputers to speed up the process of locating and troubleshooting link failures, effectively reducing the downtime of supercomputers.

Keywords: Supercomputer · Interconnection networks · High-speed link Monitoring tool

1 Introduction

Supercomputer refers to a type of computer that has extremely fast computing speed, great storage capacity, and extremely high communication bandwidth. It is mainly used in the fields of big sciences, large projects, and industrial upgrading, and plays an important role in national security, economic, and social development. It is an important symbol of national scientific and technological development level and comprehensive national strength. To meet the demand for higher computing power in scientific research and production activities, the performance of supercomputers increases 1000 times every ten years. At present, the maximum computing speed of these computers is close to 200 Petaflop/s [1] and is expected to reach Exaflop/s [2] around the year of 2020.

High performance interconnection networks [3, 4] is an important global infrastructure for supercomputers. It is the key to achieve high-speed collaborative parallel computing for all types of nodes in the system, directly affecting the performance and scalability of the system. The high-performance interconnection network is mainly composed of high performance adapters, high-radix switches, and high-speed links.

Although the probability of failure of a single interconnect component is very low, as both the system scale and the link rate are increasing, the overall failure rate of high-performance interconnect networks will continue to rise [5], giving great challenge to the reliability of supercomputers.

Systems A, B and C are three operating domestic supercomputers. Their online operating hours, interconnected network scale, and link rates are shown in Table 1. Among them, System A has the longest service life and has been online for more than 7 years. System B has the largest scale of interconnected networks. It has used more than 2,000 switches, more than 46,000 optical fibers, and more than 18,000 adapters. The deployment time of System C is the latest, but its link rate is the highest, reaching 25 Gb/s.

Table 1. The scale and link speed of three supercomputers

	System A	System B	System C
System online time	>7 years	>4 years	>2 years
Link speed	QDR (10 Gb/s)	FDR (14 Gb/s)	EDR (25 Gb/s)
Number of switches	220+	2000+	340+
Number of links	3100+	46000+	4600+
Number of NICs	3000+	18000+	11300+

According to different properties, interconnection faults can be classified into software faults and hardware faults, where hardware faults can be divided into switch, link, and adapter faults. Because the two ends of the link are connected to different switches, the failure of the link usually indicates that the port of the switch is in faulty. In real systems, link failures can be discovered by monitoring the port state of the switch.

Due to different deployment time of each system, the time span of operation and maintenance data statistics are also different. The first investigated HPC system, System A, in operation from December 2015 to May 2018. The system B was in operation from January 2017 to May 2018. The third investigated HPC system, System C, was online from January 2017 to June 2018. The proportions of various types of interconnection failures in the three systems are shown in Table 2. Hardware failures account for more than 90% of total interconnection failures in all three systems. Among them, the proportion of adapter failures is relatively small, mainly focusing on switch and link failures. In system A, the switch failures reached 81.05%, and the link failures were only 10.53%. The reason is that the link rate of system A is QDR, and as the service time of the system increases, the aging of electronic components leads to increased switch failures. System B and System C, on the other hand, use FDR and EDR fibers, and their link failure ratios reach 76.61% and 61.94%, respectively.

The first investigated foreign HPC system, Deimos, in operation from March 2007 to April 2012 at TU-Dresden, is a 728-node cluster with 108 IB switches and 1,653 links. Their hardware failure ratios reach 87%. The second foreign HPC system, TSUBAME2.0, online from April 1997 to August 2005, uses a dual-rail QDR IB network with 501 switches and 7,005 links to connect the 1,408 compute nodes. The dominated hardware failure is link failure, which reached 93% [5].

It is not difficult to find that with the increase of the system scale and the link rate, the link failure has become the most important type of failure in the interconnection network, bringing great challenges to the maintenance of the interconnection network.

Table 2. Percentage of different kinds of interconnection failures

	System A	System B	System C	Deimos	TSUBAME2.0
Percentages of network-related failures					
Software	6.86%	0.53%	3.6%	13%	1%
Hardware	93.14%	99.47%	96.4%	87%	99%
Percentages for hardware only					
NIC/NIS	8.42%	0.46%	6.72%	59%	1%
Switch	81.05%	22.93%	31.34%	14%	6%
Link	10.53%	76.61%	61.94%	27%	93%

At present, the fault localization and recovery of the interconnection network have become an important part of daily operation and maintenance of the supercomputers. When interconnection faults occur, how to assist the system operation and maintenance personnel to quickly locate and eliminate interconnection faults, so as to reduce the scope of the interconnection faults as much as possible, is an important issue that needs to be solved in the process of operation and maintenance of interconnection networks. In order to meet the needs of operation and maintenance personnel to monitor the status and performance of all high-speed links in the system in real-time, this paper designs a high-speed link monitoring tool based on in-band access [6] to monitor link connectivity, stability, bandwidth, etc. Information, with good real-time, scalability and robustness, has now been practically used in the operation and maintenance of domestic supercomputers to speed up the process of locating and troubleshooting link failures, which can effectively reduce the downtime of supercomputers.

The contributions of this paper can be summarized as follows:

The probe, aka Network Management Agent, based on hardware implementation reduces the latency of acquiring the status information of high-speed link, and improves the real-time performance of monitoring tools.

The process of information collection and processing is optimized from dimensions of time and space, which effectively reduces the time overhead of information collection and processing, and increases the scalability of monitoring tools.

A dynamic in-band path construction method is proposed, which can effectively solve the problem of unreachable switch caused by link failure, and improve the robustness of monitoring tools.

The structure of this paper is as follows. Section 1 introduces the background. Section 2 presents the related work. Section 3 details the structure and implementation of the high-speed link monitoring tool, and Sect. 4 provides the performance evaluation of the high-speed link monitoring tool, as well as robustness analysis. Finally, Sect. 5 concludes this paper.

2   Related Work

At present, there are two major categories in supercomputers: general networks represented by Ethernet and high-performance networks represented by InfiniBand [7].

Ethernet uses the SNMP network management protocol [8] to implement network management. This protocol checks the port status by periodically sending BFD packets between network ports. In normal circumstances, the operation and maintenance personnel need to log in to the network device to view the status information of the port. This method is directly monitored and has low efficiency, which cannot meet the requirements for real-time monitoring of system-wide links of large-scale systems. Pingmesh [9] adopted the idea of indirect detection and implemented link fault detection by sending an end-to-end probe. However, this method requires maintaining one probe between each pair of servers in the network. As the network scales growing, the number of probes that need to be maintained will increase exponentially, which results in a prolonged period of time for each probe and at the same time excessive bandwidth load, yielding it difficult to implement real-time detection. Both [10, 11] have improved the Pingmesh method based on topology-aware thinking. By simplifying the detection path, the number of end-to-end probes is effectively reduced. Microsoft's NetBouncer [12] also belongs to indirect monitoring. It sends a large number of IP-in-IP probe messages firstly, and then infers the location of the failed link based on the success or failure of probe packets. Compared with direct monitoring, indirect detection methods still have the possibility of false positives.

Mellanox's InfiniBand and Intel's OPA [13] are the main representatives of high-performance networks. Both have added a layer of subnet management [14], and the subnet manager perceives the status of the entire interconnect network through the subnet management agent. The Unified Fabric Manager (UFM) [15] developed by Mellanox and the Fabric Suite Fabric Manager (FM) [16] developed by Intel can efficiently monitor and manage the entire high-performance network. However, UFM and FM are proprietary software developed by vendors for their own high-performance networks and are not open source. This paper draws on the design concept of UFM and uses a combination of hardware and software to design and implement a high-speed link monitoring tool for domestic high-performance networks, which can monitor the status of the entire system's link in real time and fill the gaps in the country.

3   High-Speed Link Monitoring Tool Design

The aim of this paper is to design a tool that can monitor all high-speed links in the system in real-time. This tool will be deployed in home-grown supercomputers to achieve the effect of real-time acquisition of all link status and performance information in the system.

3.1   Overall Structure of the High Speed Link Monitoring Tool

The overall structure of the high speed link monitoring tool is shown in Fig. 1. It consists of link status register, link performance register, Network Management Agent

(NMA) [6], in-band network, in-band path construction module, link information collection and processing module, and link information display module. The link state register, link performance register, network management agent and in-band path are implemented by hardware. The in-band path construction module, link information collection and processing module and link information display module are implemented by software.

Link Status Register: The function of the link status register is to record the basic status information of the current link, including linkup, handup, retry, lane, and credit.

Link Performance Registers: The Link Performance Registers feature records link performance information, namely real-time transceiver traffic and bandwidth.

Network Management Agent: Each Switch contains a hardware-based network management agent module. Its role is to receive management request messages, read and write the corresponding link state or performance registers based on the contents of the messages, and then construct a management response report.

In-band network: The in-band channel is responsible for the transmission of in-band management packets. It forwards the management request packet of the management server to the destination NMA or forwards the management response packet constructed by the NMA to the management server.

In-band path construction module: The path construction module is implemented by software, and its function is to build an in-band path to each switch of the access system.

Link information collection and processing module: The function of the link information collection and processing module is to collect link status and link performance information through an in-band path and process the information. After processing is completed, it is passed to the link information display module for display.

Link information display module: The function of the link information display module is to receive the data of the link information collection and processing module, and then visually display it.
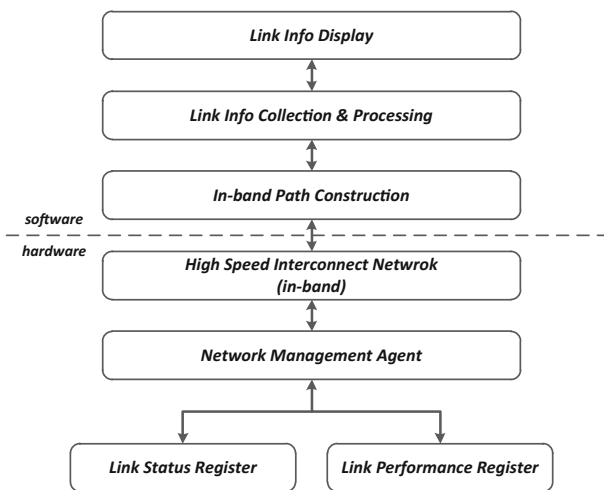

Fig. 1. The architecture of high-speed link monitoring tool

3.2   The Operating Mechanism of the High-Speed Link Monitoring Tool

The operating mechanism of the high-speed link monitoring tool is shown in Fig. 2. It mainly includes three steps: in-band path construction, information collection and processing, and information display.

Step 1: The path construction module will construct an in-band access path according to the position of each Switch in the system.

Step 2: The link collection module sequentially obtains the port state information of all the switches in the system through the in-band path.

Step 3: The link information display module completes the display of the port status information.

Step 2 and Step 3 successively display the real-time status of the system. If the link information cannot be acquired in a certain cycle, Step 1 is triggered to rebuild the in-band path. The detailed operation of each step will be detailed later.
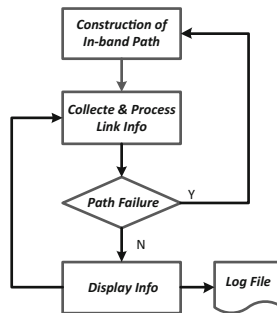

Fig. 2. The operating mechanism of the high-speed link monitoring tool

Path Construction

In-band path construction is the basis of in-band access. The path construction algorithm adopts a breadth-first search strategy. The first switch is searched from the adapter of the management server, and an access path to this switch is constructed, and then all the ports of this switch are accessed. Depending on the status of the port, the following cases are handled separately:

(1) If the port is connected to an adapter, no processing is required.
(2) If the port is connected to a switch and the switch does not construct an in-band path, an in-band path of the switch is built and added to the seed queue.
(3) If the port is connected to the switch and the in-band path of the switch already exists, no processing is required.
(4) No processing is required if the port is disconnected.

Figure 3 shows an in-band path schematic diagram of a 10 switches tree network. In this figure, svr0 is taken as the starting point, and the thick line is the built-in in-band path. The switch with the number X is represented by switchX. The specific construction process is: first find that svr0 is connected to switch0, then build the path to switch0, and then scan ports 1–7 of switch0. Ports 1–3 connected to the adapter are not processed.

Ports 4–7 are connected to switch4 and switch5. Since switch4 and switch5 have not constructed the in-band path, the in-band path to switch4 and switch5 are built, and two switches are added to the seed queue at the same time. After scanning all the ports of switch0 is completed, there are switch4 and switch5 in the seed queue. Then switch4 and switch5 perform port scanning in turn. When scanning switch4, new seeds switch8 and switch9 are generated, then these new seeds will be scanned in sequence. After the scanning of switch8 and switch9 is completed, the new seeds switch6 and switch7 are generated, and then switch6 and switch7 will be scanned in sequence. After that, new seeds switch2 and switch3 are generated. Finally, when the scanning of switch2 and switch3 is completed, no new seeds will be generated and the seed queue will be empty. At this point, the construction of in-band path to each switch is completed.
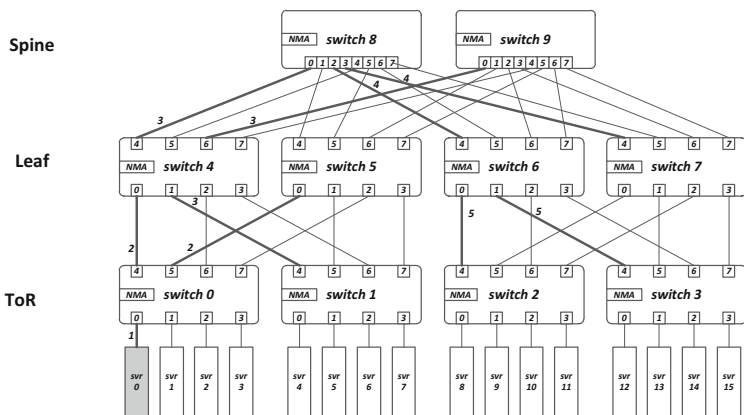

Fig. 3. The schematic diagram of in-band path

The port number Y of the switch with the number X is represented by switchX.pY. Each hop of in-band path is represented by the remote port of link from the starting point (svr0). Table 3 shows the in-band path of each switch that is constructed. In this example tree network, the maximum number of hops for the in-band path is 5.

Table 3. The in-band path of each switch

Switch name	Hops	In-band path
switch0	1	svr0→switch0.p0
switch1	3	svr0→switch0.p0→switch4.p0→switch1.p4
switch2	5	svr0→switch0.p0→switch4.p0→switch8.p0→switch6.p4→switch2.p4
switch3	5	svr0→switch0.p0→switch4.p0→switch8.p0→switch6.p4→switch3.p4
switch4	2	svr0→switch0.p0→switch4.p0
switch5	2	svr0→switch0.p0→switch5.p0
switch6	4	svr0→switch0.p0→switch4.p0→switch8.p0→switch6.p4
switch7	4	svr0→switch0.p0→switch4.p0→switch8.p0→switch7.p4
switch8	3	svr0→switch0.p0→switch4.p0→switch8.p0
switch9	3	svr0→switch0.p0→switch4.p0→switch9.p0

Link Information Collection Processing

The link information collection processing is divided into two subtasks: information collection and information processing. The information collection is responsible for collecting the status and performance information of all the links in the network through in-band access. The information processing is responsible for disconnection of link, the change of handup, the oversize of retry, the credit abnormality and other key information extraction. The time required for information collection and processing is an important factor affecting the real-time and scalability of the monitoring tools. In order to speed up the process of information collection and processing and shorten the information collection and processing time, the information collection and processing module is optimized from two dimensions: space and time. Regarding to spatial dimension, when tasks are too large, tasks are grouped and processed in parallel by multiple threads. With respect to the time dimension, streamlining operations are used to concurrently send management request packets to multiple switches and then receive switch management response packets. The message processing process reduces waiting time.

Information Display

The main function of the information display is to display the collected and processed data visually. Figures 4(a) and (b) show the effect of link retry and lane number respectively. In addition, the information display operation is also responsible for saving key information such as linkup disconnection, handup change, retry oversize (above the threshold), and credit abnormality to the log file for future analysis.
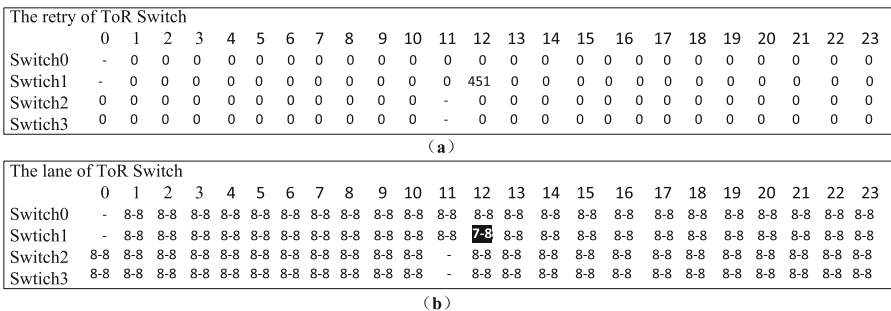
The retry of ToR Switch

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Switch0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Swtich1	-	0	0	0	0	0	0	0	0	0	0	0	451	0	0	0	0	0	0	0	0	0	0	0
Switch2	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0
Swtich3	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0

（a）

The lane of ToR Switch

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Switch0	-	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8
Swtich1	-	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	**7-8**	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8
Switch2	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	-	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8
Swtich3	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	-	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8	8-8

（b）

Fig. 4. The monitoring result of switches' retry and lane

3.3   Basic Functions of the High-Speed Link Monitoring Tool

The basic functions of the high-speed link monitoring tool are shown in Table 4, which include two parts: the status monitoring and performance monitoring. The state monitoring is responsible for real-time monitoring of the basic state of high-speed links in the interconnection system, including linkup, handup, retry, lane, and credit. The performance monitoring is responsible for real-time detection of the performance of high-speed links in the interconnected system, including traffic and bandwidth.

Linkup: This status indicator reflects the connectivity of the link. When the link is down, the link status register will be linked up.

Handup: This status indicator reflects the link stability, which will cause the value of the handup register to change.

Retry: This status indicator reflects the link quality. The smaller the retry, the better the link quality.

Lane: This status indicator reflects the link's communication capabilities. Decreasing the number of port lanes will cause the current link's communication capabilities to decrease.

Credit: This status indicator reflects the size of the buffer at the receiving end of the link.

Traffic: This performance indicator reflects the number of packets sent and received by the link over a period of time.

Bandwidth: This performance indicator reflects the utilization of the link.

According to the long-term accumulation of operation and maintenance experience, when operation and maintenance personnel can obtain the above-listed link information in real time, they can fully grasp the current operating conditions of the interconnection network, and quickly discover and locate various link failures, even the gray failures [17].

Table 4. The basic functions of the high-speed link monitoring tool

Function		The description of function
Status monitoring	Linkup	Get the linkup register of each link, report the disconnect link
	Handup	Get the handup register of each link, report the change of handup
	Retry	Get the retry register of each link, report the oversize of retry
	Lane	Get the lane register of each link, report the decreasing of port lanes number
	Credit	Get the credit register of each link, report the credit abnormality
Performance monitoring	Traffic	Monitor the real-time traffic of all links
	Bandwidth	Monitor the real-time bandwidth of all links

4 Performance Evaluation and Analysis

4.1 Real-Time Property

The in-band path construction and information collection processing time are two key factors that affect the real-time property of the monitoring tools. System D, System E, and System F are three online domestic supercomputers. Link monitoring tools are deployed on three systems. Table 5 shows the average in-band path construction time and information collection processing time, which can meet the need for real-time monitoring of all high-speed links in each system. Table 5 also shows the single switch

information collection and processing time of System D, E and F, and compares it with Tianhe-2. It can be seen that the single-switch information collection and processing time of System D, E and F are about 0.5 ms, which is obviously better than the 1.01 ms of Tianhe-2 [6].

Table 5. The results of real-time property test

System name	Number of switches	In-band path construction time (ms)	System-wide information collection and processing time (ms)	Single switch information collection and processing time (ms)
System D	86	3.862	1.818	0.4963
System E	86	4.439	1.837	0.5051
System F	96	5.183	2.219	0.4981
Tianhe-2	5856		2000	1.01

4.2 Scalability

This section discusses the scalability of the proposed tool from two aspects: topology change and system scale.

Topology Change
During usage, the monitoring tool is deployed on the management server and uses the server as the root node to build an in-band access path, as shown in Fig. 3. Different topologies will affect the hops of the in-band path from the management server to the switch.

Figure 5(a) shows the relationship between the number of hops and the time taken to collect and process single switch information on system D. When the number of hop steps increases by 5, the corresponding information collection processing time will increase by 74.2 μs (515.3 μs–441.1 μs), an increase of 16.8%. From the trend of the curve, it can be predicted that when the difference in the number of hop steps increases to 10, the increase in access time will not exceed 40%. According to the actual construction of the supercomputer, the difference in the number of hops caused by the topology will not exceed 10. It can be seen that the topology change has little impact on the performance of the monitoring tool and can be deployed on domestic supercomputers with different topologies.

System Scale
As the scale of the system increases, the processing time for system-wide information collection will inevitably increase. In order to reduce information collection processing time, this tool optimizes the information collection process. On system B, the time required for the collection and processing of different numbers of switch links was tested. Table 6 shows the comparison of the time required for collection and processing of different numbers of switches before and after optimization. When the number of switches increases to 1024, the optimized information is obtained. Approximately 34 times the speedup ratio can be obtained before the collection processing time is optimized.
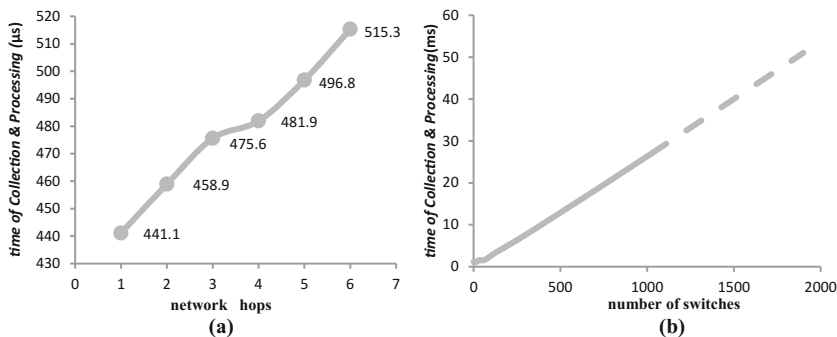
Fig. 5. The test of collection and processing time with system scale

Table 6. The required time for collection and processing before and after optimization

Number of switches	Required time before optimization (ms)	Required time after optimization (ms)	Speedup
1	1.33	1.24	1.077
2	2.27	1.12	2.022
4	4.09	1.06	3.858
8	7.72	1.12	6.876
16	15.12	1.12	13.443
32	28.37	1.57	18.071
64	58.14	1.64	35.327
128	116.84	3.39	34.411
256	229.73	6.50	35.339
512	458.37	13.21	34.693
1024	917.44	26.98	33.999

Figure 5(b) shows the curve of the change of the information collection processing time with the number of switches after optimization. From the figure, we can see that when the number of switches is less than 64, the time-consuming curve changes smoothly and the acceleration ratio increases linearly. When the number of switches is greater than 64, the time-consuming curve rises linearly, and the acceleration ratio remains basically unchanged. China is expected to complete the deployment of the exascale supercomputer around 2020. If 36-port switches and a 4-level fat-tree topology are adopted, the number of switches in the system will reach more than 40,000. According to the trend of the time-consuming curve, the time for completing the system-wide link information collection and processing with this monitoring tool is about 1.08 s, which can meet the needs of the operation and maintenance personnel to monitor the link status in real time.

4.3   Robustness

This section discusses the robustness of the tool from both in-band path failures and server failures.

In-band Access Path Failure

The obtaining of the switch port information depends on the in-band access path. If a link fails on the path, some switches will be unreachable and the port status information of these switches cannot be obtained. To deal with in-band access path failure, the monitoring tool dynamically constructs in-band access paths. When a tool finds that a switch is unreachable, it will restart the in-band path construction to automatically
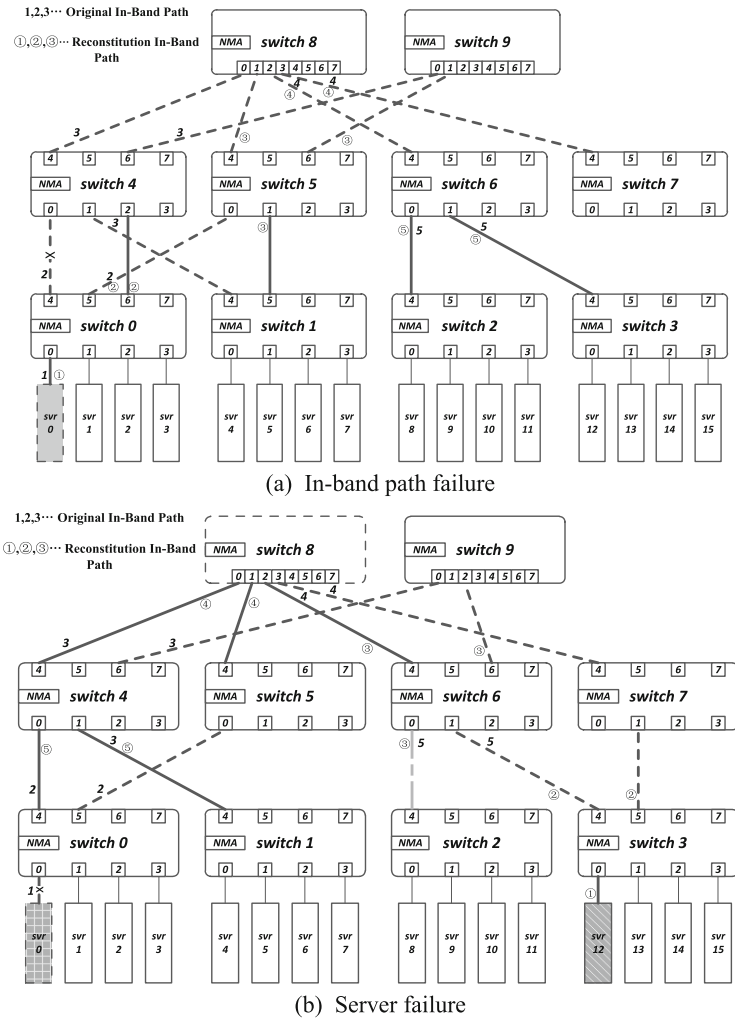

(a) In-band path failure

(b) Server failure

Fig. 6. The schematic diagram of in-band path reconstitution after failure

avoid failure links. The specific process is shown in Fig. 6(a). When the port switch4.port0 fails, the original in-band path from svr0 to switch4 svr0→switch0. p0→switch4.p0 is unreachable. After the tool detects that switch4 is unreachable, it will rebuild the in-band path from svr0 to switch4, which is svr0→switch0. p0→switch4.p2, and at the same time svr0 to switch1 ∼ switch3 and switch6 ∼ switch9 in-band paths have also been rebuilt. It can be seen that for a specific switch, only one reachable port is needed to build its in-band access path, ensuring that the status of all the ports on the switch can be obtained in real time.

Server Failure

The server failures are divided into server itself failure and server link failure. The consequences of the two failures are that the server cannot monitor the link information. In actual operation, the monitoring tool is deployed on at least two servers. One is the main server and the other is the standby server. When the main server fails, the standby server will take over the monitoring of system link information. In order to avoid the simultaneous failure of the main server and the standby server caused by the switch failure, the main server and the standby server may be connected to different switches in the system. As shown in Fig. 6(b), svr0 is the main server, and svr12 is the standby server, which is connected to switch0 and switch3. When the main server svr0 fails, the system can also be monitored by the standby server svr12. The link information in the system effectively tolerates server failures.

5　Conclusion

This paper draws on the design idea of Unified Fabric Manager and uses a combination of hardware and software to design and implement a high-speed link monitoring tool for domestic high-performance networks. According to the actual performance evaluation and analysis, the tool has good real-time performance, robustness and scalability, which can meet current and even future exascale supercomputer system-wide link monitoring requirements, and can speed up the process of locating and troubleshooting link failures as well as shorten the supercomputer's downtime.

References

1. https://www.top500.org/. Last Accessed 30 May 2018
2. The Opportunities and Challenges of Exascale Computing. Summary Report of the Advanced Scientific Computing Advisory Committee (ASCAC) Subcommittee, Office of Science, DOE (2010)
3. Duato, J., Yalamanchili, S., Ni, L.: Interconnection Networks: An Engineering Approach. Morgan Kaufmann Publishers, California (2003)

4. Dally, W.J., Towles, B.: Principles and Practices of Interconnection Networks. Morgan Kaufmann Publishers, California (2004)
5. Domke, J., Hoefler, T., Matsuoka, S.: Fail-in-place network design: interaction between topology, routing algorithm and failures. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC 2014), pp. 597–608. IEEE Press, New Orleans (2014)
6. Cao, J.J., Xiao, L.Q., Wang, K.F.: The implementation and evaluation of in-band network management in supercomputing system. Chin. J. Comput. **39**(9), 1717–1732 (2016)
7. Introduction to InfiniBand-White Paper. https://www.mellanox.com/pdf/whitepapers/IB_Intro_WP_190.pdf. Last Accessed 3 June 2018
8. Wang, H.R., Xu, M.W.: Survey on SNMP network management. Mini Micro Syst. **25**(3), 358–366 (2004)
9. Guo, C.X., Yuan, L.H., Xiang, D., et al.: Pingmesh: a large-scale system for data center network latency measurement and analysis. In: Proceeding of ACM SIGCOMM 2015, pp. 139–152. ACM Press, London (2015)
10. Peng, Y., Yang, J., Wu, C., et al.: deTector: a topology-aware monitoring system for data center networks. In: 2017 USENIX Annual Technical Conference (USENIX ATC 2017), USENIX Association, pp. 55–68. USENIX Association, California (2017)
11. Wang, J.X., Qi, H.: Real-time link fault detection as a service for datacenter netwrok. J. Comput. Res. Dev. **55**(4), 704–716 (2018)
12. CloudBrain for Automatic Troubleshooting for the Cloud. https://www.microsoft.com/en-us/research/project/cloudbrain/. Last Accessed 5 June 2018
13. Birrittella, M.S., Debbage, M., et al.: Intel omni-path architecture: enabling scalable, high performance fabrics. In: Proceeding of 23rd IEEE Annual Symposium on High-Performance Interconnects, pp. 1–9. IEEE Press, California (2015)
14. Wen, J.W.: Infiniband subnet management technology. Master thesis, National University of Defense Technology (2009)
15. Unified Fabric Manager for InfiniBand User Manual. http://pleiades.ucsc.edu/doc/mellanox/UFM_5.2_User_Manual_IB_DOC-00600.pdf. Last Accessed 28 May 2018
16. Intel Omni-Path Fabric Suite Fabric Manager – User Guide. https://www.intel.com/content/dma/support/us/en/documents/network-and-i-o/fabric-products/Intel_OP_FabricSuite_Fabric_Manager_UG_H76468_v8_0.pdf. Last Accessed 12 June 2018
17. Huang, P., Guo, C.X., Zhou, L.D, et al.: Gray failure: the Achilles' heel of cloud-scale systems. In: Proceedings of the 16th Workshop on Hot Topics in Operating Systems (HotOS 2017), pp. 150–155. ACM Press, Whistler (2017)