A high-speed normality check scheme based on a lump link check for large-scale networks

Ippei Taga
Department of Computer Science and Engineering, Toyohashi University of Technology,
Aichi, Japan
taga.ippei.py@tut.jp

Akio Kawabata
Department of Computer Science and Engineering, Toyohashi University of Technology,
Aichi, Japan
kawabata.akio.oc@tut.jp

***Abstract*—This paper proposes a high-speed normality check scheme based on a lump link check for silent failure that does not notify alarms. In the proposed scheme, a lump link check is performed for each divided network, thereby reducing the number of times that all links are checked one by one (conventional scheme), such as high-accuracy network monitoring and control (HANMOC). The proposed scheme consists of two processes. In the first process, the number of divisions to minimize the number of checks from the number of links in the network is determined. In the second process, each check route for a lump link check is decided in all divided networks based on the number of divisions obtained in the first process. The check route decision problem is formulated as an integer linear programming (ILP) problem improved from the vehicle routing problem (VRP). Numerical results indicate that the number of checks of the proposed scheme to identify a failure link is on average about 50% that of the conventional scheme in the NSFNET and COST239 networks. From these results, the proposed scheme is effective for high-speed normality check in large-scale networks.**

***Index Terms*—Lump link check, vehicle routing problem, silent failure, high-speed normality check, HANMOC**

I. Introduction

With the launch of various cloud services and the Internet of Things (IoT), diverse applications are provided via networks. In this situation, prolonged network failures significantly impact society, and telecommunication carriers are building highly reliable networks that can switch to redundant routes during a network failure. Most network failures are detected by network systems and notified to a network operator by alarms. However, when failures that are not detected and notified occur or when only increased latency (named silent failure) occurs, it cannot switch to a redundant route and may take significant time to locate the failure point. To detect such silent failures, approaches that check the normality of each link using segment routing (SR) [1], such as high-accuracy network monitoring and control (HANMOC) [2], have been proposed the conventional scheme. However, because the number of checks required in the conventinal scheme is equal to the number of links, a significant number of checks are required for a large-scale network. To reduce the number of checks, we propose a high-speed normality check scheme based on a lump link check per divided network. In a lump link check, the check packet passes through all links in the target network in a single stroke. In the proposed scheme, the divided network where failure is detected is divided into smaller networks. Then, the lump link check is repeated in each divided network, and the faulty network is divided again to identify a failure point. In this way, the number of checks in the proposed scheme is reduced compared to that of the conventional scheme, which checks the normality of each one by one.

The proposed scheme consists of two processes. In the first process, the number of divisions to minimize the number of checks is determined from the number of links in the network. In the second process, the network is divided by the number of divisions obtained by the first process, and each lump link check route is decided. The route decision problem for the lump link check in the divided networks is formulated as an integer linear programming (ILP) problem improved from the vehicle routing problem (VRP) [3]. In the evaluation, we indicate that the proposed scheme is effective in reducing the number of normality checks compared to that of the conventional all-link check at HANMOC [2] in two network topologies (NSFNET [4] and COST239 [5]).

II. Conventional Scheme

We explain the conventional scheme of checking the normality of each link, using HANMOC [2] as an example. The conventional scheme uses SR [1] for packet routing, which differs from conventional routing schemes such as OSPF [6], to realize specified routing policies by pre-defining them. The normality check packet is routed for the target link in the network using SR and checks the state of the data plane (latency, jitter, and connectivity). In the case of HANMOC, symmetric round trip (SRT) packets are sent from a measurement system to the target link in the network. The SRT packet (error detection packet) identifies the failure link. As shown in Fig. 1, if the delay significantly increases from Router B to Router A, the SRT packets are sent to two routes that include and exclude the target link, respectively, to confirm the reachability and occurrence of a considerable delay and identify the failure link.

The number of error detection packet, such as SRT packet in HANMOC, measurements increases proportionately to the

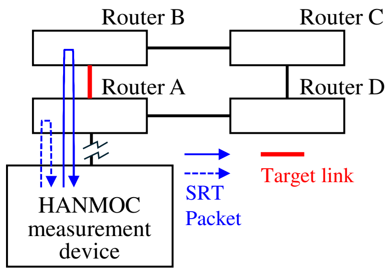


Fig. 1: SRT measurement procedures in HANMOC.

number of links because, in the conventional scheme, all links are checked one by one (all-link check). This is an issue for large-scale networks with extended periods of failure time because the normality check for all-link check increases the time it takes to detect the failure point. Silent failure is a rare network failure; most failures are immediately recovered by rerouting to the backup route or the network operator's handling. Therefore, we introduce a high-speed normality check scheme based on the lump link check to reduce the number of error detection packet measurements, assuming that multiple silent failures do not occur at once.

III. Proposed scheme

A. Overview

In the proposed scheme, the network is divided into multiple areas for the lump link check, and the divided network where failure is detected is divided into smaller networks. The error detection packet passes through all links in the divided network in a single stroke. The proposed scheme consists of two processes. In the first process, the number of divisions to minimize the number of checks is decided from the number of links in the network (division number decision process). In the second process, the routes of lump link checks in the divided networks are decided by the number of divisions obtained in the first process (check route decision process).

B. Division number decision process

In the first process, the number of divisions is decided to minimize the total number of checks to identify the location of a failure link based on the number of links in the network. The number of divisions is decided without considering the network topology. For example, when a network with 100 links is divided into five divisions (example 1), the network is divided into five sets of 20 links, requiring five checks if 20 links can be checked in a single stroke. After the previous check, only the network in which a failure is detected is further divided into 5 (5 sets of 4 links) and checked in a single stroke. These divisions and checks are repeated until the number of links in the divided network is less than the number of divisions.

In example 1, division procedures have two stages (stage 1 is 5 sets of 20 links, and stage 2 is 5 sets of 4 links). The total numbers of links and divisions are denoted as N and a, respectively. The number of division stages until a failure link is identified, k, is given as a value when $\frac{N}{a^k} < a$ is satisfied for the first time by continuously dividing the number of links N by a. In example 1, $k = 2$. The maximum number of remaining links when division is no longer possible is expressed by dividing the total number of links by the number of division raised to the power of the number of stages, and the result is rounded up to an integer. In example 1, the maximum number of remaining links is 4 (=$\lceil \frac{100}{5^2} \rceil$).

The total number of checks to identify the location of a failure link is expressed as (1a). In example 1, the total number of checks is 14 (=$5 * 2 + \lceil \frac{100}{5^2} \rceil$).

$$ak + \lceil \frac{N}{a^k} \rceil \tag{1a}$$

The total number of checks is compared by varying the number of divisions, a, from 2 to N in (1a). From the results, the division number with the fewest total number of checks is used in the second process.

C. Check route decision process

In the second process, the lump link check routes, including all the links in each divided network, are decided by the number of divisions obtained in the first process. Considering the network topology, each check route must be determined to pass through all links. We introduce a route decision improved from VRP [3] to realize this route decision problem. VRP is a well-known combinatorial optimization problem that determines the optimal route for each truck when delivering packages to many customers via multiple trucks. All trucks start from a depot, go to a package delivery point, deliver packages to customers, and then return to the depot. Each truck delivers only once to each customer. Under these conditions, VRP finds routes that pass through all customers at a lower cost (e.g., total distance traveled by all trucks).

We now describe how we improved VRP for the check route decision process. In the process, the number of trucks in VRP is treated as the number of divisions. Each truck's route is the route each error detection packet takes in one stroke. In VRP, the truck visits all the nodes (customers), but error detection packets pass through all the links in the divided network. The error detection packets of different divided networks passing through the same link (overlap-link) or the same error detection packet passing through a link multiple times are allowed. The objective function of VRP aims to minimize the total cost of all trucks. In the process, the first objective function is the number of overlap links, and the second is the maximum route for all routes. The second objective function is minimized under the condition that the first objective function is minimized. The first objective function aims to divide the network without overlaps, and the second objective function aims to minimize the check time. In parallel checking, the longest route determines the checking time, so the maximum route is minimized.

D. Formulation

The check route decision process is formulated as an ILP problem to determine each check route by the number of divisions as a given parameter. The network is modeled as a directed graph (V, E). Let N be the number of nodes in the network and a be the number of divisions obtained by the first process. Let $V = \{1, 2, \ldots, N\}$ indicate the set of nodes, where $i \in V$ denotes a node. Let E indicate the set of links between nodes and $(i, j) \in E$ denote the link between nodes $i \in V$ and $j \in V$. Let $K = \{1, 2, \ldots, a\}$ indicate the set of check routes of each divided network and $k \in K$ denote the check routes.

1) Parameter and variables: Table I lists the given parameters and decision variables used in the ILP problem. The given parameters are as follows. The cost of link $(i, j) \in E$ is denoted by c_{ij} as the distance of a link.

The decision variables are as follows. x^k_{ij} is a binary variable for check route $k \in K$ and link $(i, j) \in E$, where $x^k_{ij} = 1$ when check route $k \in K$ passes through link $(i, j) \in E$, and $x^k_{ij} = 0$ otherwise. x'^k_{ij} is a binary variable for link $(i, j) \in E$, where $x'^k_{ij} = 1$ when check route $k \in K$ passes through either links $(i, j) \in E$ or $(j, i) \in E$, and $x'^k_{ij} = 0$ otherwise. d_i is a binary variable for node $i \in V$, where $d_i = 1$ when node $i \in V$ is selected as a depot, and $d_i = 0$ otherwise. y^k_{ijl} is a binary variable for check route $k \in K$ and links $(i, j) \in E$ and $(j, l) \in E$, where $y^k_{ijl} = 1$ when $k \in K$ passes through link $(i, j) \in E$ after link $(j, l) \in E$, and $y^k_{ijl} = 0$ otherwise. y'^k_{ijl} and y''^k_{ij} are binary variables for calculating y^k_{ijl}. $y'^k_{ijl} = 1$ when check route $k \in K$ passes through both links $(i, j) \in E$ and $(j, l) \in E$, and $y'^k_{ijl} = 0$ otherwise. $y''^k_{ij} = 1$ when the number of times that the check route $k \in K$ passes through link $(i, j) \in E$ is less than or equal to one, and $y''^k_{ij} = 0$ when the number of times that the check route $k \in K$ passes through link $(i, j) \in E$ is greater than one. u^k_{ij} is a variable that stores the order of passes through link $(i, j) \in E$ on check route $k \in K$. $L_{\max}$ is the maximum route of $k \in K$.

2) Formulation of problem: The check route in each divided network is determined based on the number of divisions by the following equation:

$$\text{Objective} \quad \min\left(\frac{(\sum_{(i,j)\in E, i\neq j}(\sum_{k\in K} x'^k_{ij} - 1)}{2} + \alpha L_{max}\right) \tag{2a}$$

$$\text{s.t.} \quad 1 \leq \sum_{k\in K} x'^k_{ij}, (i, j) \in E, i \neq j \tag{2b}$$

$$\sum_{i\in V, i\neq j} x^k_{ij} - \sum_{i\in V} x^k_{ji} = 0, \forall j \in V, k \in K \tag{2c}$$

$$x'^k_{ij} = |x^k_{ij} - x^k_{ji}|, \forall k \in K, (i, j) \in E \tag{2d}$$

$$\sum_{i\in V} d_i = 1 \tag{2e}$$

$$\sum_{(i,j)\in E, i\neq j} x^k_{ij} \geq d_i, \forall k \in K, i \in V \tag{2f}$$

TABLE I: Parameters and decision variables.

Given Parameters	Description
V	Set of nodes
E	Set of links
K	Set of check routes
c_{ij}	Cost of link $(i, j) \in E$
Description Variable	**Description**
x^k_{ij}	$x^k_{ij} = 1$ if check route $k \in K$ passes through link $(i, j) \in E$, $x^k_{ij} = 0$ otherwise
x'^k_{ij}	$x'^k_{ij} = 1$ if check route $k \in K$ passes through either links $(i, j) \in E$ or $(j, i) \in E$, $x'^k_{ij} = 0$ otherwise
d_i	$d_i = 1$ if node $i \in V$ is selected as a depot, $d_i = 0$ otherwise
y^k_{ijl}	$y^k_{ijl} = 1$ if check route $k \in K$ passes through link $(i, j) \in E$ after link $(j, l) \in E$, $y^k_{ijl} = 0$ otherwise
y'^k_{ijl}	$y'^k_{ijl} = 1$ if check route $k \in K$ passes through both links $(i, j) \in E$ and $(j, l) \in E$, $y'^k_{ijl} = 0$ otherwise
y''^k_{ij}	$y''^k_{ij} = 1$ if number of times that check route $k \in K$ passes through link $(i, j) \in E$ is less than or equal to one, $y''^k_{ij} = 0$ otherwise
u^k_{ij}	Order of passes through link $(i, j) \in E$ on check route $k \in K$
$L_{\max}$	Maximum route length of $k \in K$

$$u^k_{ij} - u^k_{jl} + N \cdot y^k_{ijl} \leq N - 1 + \beta d_j, \quad \forall i, j, l \in V, k \in K, i \neq j, j \neq l \tag{2g}$$

$$y'^k_{ijl} = x^k_{ij} \cdot x^k_{jl}, \forall (i, j), (j, l) \in E, k \in K \tag{2h}$$

$$y^k_{ijl} = y'^k_{ijl}, \forall (i, j), (j, l) \in E, i \neq l, k \in K \tag{2i}$$

$$y^k_{ijl} = y'^k_{ijl} \cdot y''^k_{ij}, \forall (i, j), (j, l) \in E, i = l, k \in K \tag{2j}$$

$$\sum_{(j,l)\in E} x^k_{jl} \geq 2 - N * y''^k_{ij}, \quad \forall (i, j) \in E, i \neq j, k \in K \tag{2k}$$

$$u^k_{ij} \leq N + \beta d_i, \forall i, j \in V, k \in K \tag{2l}$$

$$L_{\max} \geq \sum_{(i,j)\in E} c_{ij} x^k_{ij}, \forall k \in K \tag{2m}$$

$$x^k_{ij} \in \{0, 1\}, \forall k \in K, (i, j) \in E \tag{2n}$$

$$x'^k_{ij} \in \{0, 1\}, \forall k \in K, (i, j) \in E \tag{2o}$$

$$y^k_{ijl} \in \{0, 1\}, \forall k \in K, (i, j) \in E, (j, l) \in E \tag{2p}$$

$$y'^k_{ijl} \in \{0, 1\}, \forall k \in K, (i, j) \in E, (j, l) \in E \tag{2q}$$

$$y''^k_{ij} \in \{0, 1\}, \forall k \in K, (i, j) \in E \tag{2r}$$

$$u^k_{ij} \in \{0, 1, 2, \cdots\}, \forall k \in K, (i, j) \in E. \tag{2s}$$

Equation (2a) minimizes the total number of links with multiple overlapping check routes as the first objective function and the largest value of the total cost of each check route as the second objective function. Because the first objective function is integer-valued, α is determined such that the maximum possible value of the second objective function is a fraction, making the first objective function much greater than the second objective function. Equation (2b) indicates that if a link exists, at least one check route must pass through the link. Equation (2c) indicates that the number of routes that visit each node is equal to the number of routes that leave the node, in all nodes. Equation (2d) indicates that x'^k_{ij} is 1 if either x^k_{ij} or x^k_{ji} is 1. Equation (2e) indicates that there is only one node that is a depot. Equation (2f) indicates that there is always at least one check route leaving the depot node. Equation (2g) is an MTZ constraint [7] that eliminates partial cycles. The links are numbered, starting from 1, in the order in which the check route passes through them, and the check route has been restricted so that it cannot visit a link with a lower number than the current number. βd_j is a term set to exclude node j from the constraints only when node j is a depot, in accordance with the MTZ constraint. β is a sufficiently large number. Equation (2h) indicates that if $y'^k_{ijl} = 1$, check route $k \in K$ paths through both links $(i, j) \in E$ and $(j, l) \in E$ because $x^k_{ij} = 1$ and $x^k_{jl} = 1$. Equation (2i) indicates that for $i \neq l$, the value of y^k_{ijl} is the same as that of y'^k_{ijl}. Equation (2j) indicates that if $y^k_{ijl} = 1$, $y''^k_{ij} = 1$, and $i = l$, check route $k \in K$ passes through both links $(i, j) \in E$ and $(j, l) \in E$ and node j is a turnaround point. Equations (2k) indicates that, for a probing check route $k \in K$, if $y''^k_{ij} = 1$, then the number of links originating from node j is at most one. This means that node j is a turnaround point on check route k. Equation (2l) indicates that u^k_{ij} is less than or equal to N, except when node i is a depot. Equation (2m) indicates that the maximum route does not exceed L_{max}. Equations (2n)–(2r) indicate that each variable is a binary variable. Equation (2s) indicates that u^k_{ij} is a natural number.

3) Linearization of the problem as ILP: We linearize (2a)-(2s) as an ILP problem. Because x^k_{ij}, y'^k_{ijl}, and y''^k_{ij} are binary variables. (2d), (2h), and (2j) are linearized, and (2a)–(2s) are formulated as the following ILP problem:

$$\text{Objective} \quad \min\left(\frac{(\sum_{(i,j)\in E, i\neq j}(\sum_{k\in K} x'^k_{ij} - 1)}{2} + \alpha L_{max}\right) \tag{3a}$$

$$\text{s.t.} \quad x'^k_{ij} \geq x^k_{ij}, \forall k \in K, (i, j) \in E \tag{3b}$$

$$x'^k_{ij} \geq x^k_{ji}, \forall k \in K, (i, j), (j, i) \in E \tag{3c}$$

$$x'^k_{ij} \leq 1, \forall k \in K, (i, j) \in E \tag{3d}$$

$$y'^k_{ijl} \leq x^k_{ij}, \forall (i, j), (j, l) \in E, k \in K \tag{3e}$$

$$y'^k_{ijl} \leq x^k_{jl}, \forall (i, j), (j, l) \in E, k \in K \tag{3f}$$

$$y'^k_{ijl} \geq x^k_{ij} + x^k_{jl} - 1, \quad \forall (i, j), (j, l) \in E, k \in K \tag{3g}$$

$$y^k_{ijl} \leq y'^k_{ijl}, \forall (i, j), (j, l) \in E, i = l, k \in K \tag{3h}$$

$$y^k_{ijl} \leq y''^k_{ij}, \forall (i, j), (j, l) \in E, i = l, k \in K \tag{3i}$$

$$y^k_{ijl} \geq y'^k_{ijl} + y''^k_{ij} - 1, \quad \forall (i, j), (j, l) \in E, i = l, k \in K \tag{3j}$$

$$(2b) - (2c), (2e) - (2g), (2k) - (2i), (2l) - (2s). \tag{3k}$$

E. Full procedure of proposed scheme

In this section, we describe the proposed scheme's full procedure using the division number and the check route decision processes. Fig. 2 shows a flowchart of the total procedure of the proposed scheme. In Step 1, based on the number of links, the optimal number of divisions is determined using (1a) to minimize the total number of checks to identify the location of a failure link. In Step 2, each divided network check route is determined based on the number of divisions obtained in Step 1 using (3a)–(3k). In Step 3, each divided network is checked for normality in one stroke using the error detection packet. Depending on the outcome of Step 3, the next step branches into three steps. When all divided networks are normal, the next step returns to Step 3 again. When a failure is detected only in a specific divided network, the next step is Step 4-1. Because the other divided networks are confirmed to be normal, this divided network is the only one to be checked further. In Step 4-1, the number of links in the failure network is compared to the number of divisions. If the number of links in the divided network is less than the number of divisions, the network cannot be divided any further, and check (Step 5) is performed. Otherwise, the network can be divided further, and the process returns to Step 1. In Step 3, if failures are detected in multiple divided networks, the next step is Step 4-2. Because only a single fault location exists but failures have been detected in multiple divided networks, it can be identified that the fault has occurred at overlapping links of the multiple check routes. Therefore, these overlapping links are extracted in Step 4-2. After Steps 4-1 or 4-2, to identify the location of the failure link, all links in the failure network or the extracted links are checked in Step 5.

IV. Evaluation

A. Prerequisites

To evaluate the reduction of the normality check process, we compare the number of checks of the proposed scheme with those of the all-link check in NSFNET (14 nodes, 22 links) [4] and COST239 (11 nodes, 26 links) [5] as shown in Fig.3. In the evaluation, the number of checks indicates the sum of the error detection packet one-stroke checks to identify the failure link. We assume there is only a single-failure link and no simultaneous failures. In addition, each link failure is considered to be bidirectional, and one-way link failures are not considered. The costs between nodes i and j, c_{ij}, are evaluated as the linear distance from the latitude and longitude of the city in each network. The length of the error detection packet check route is the sum of the length of links in the

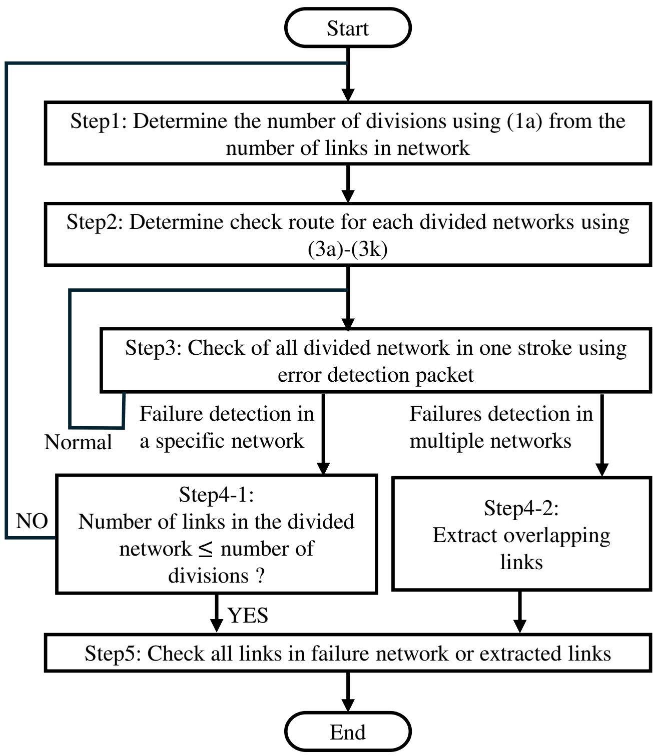
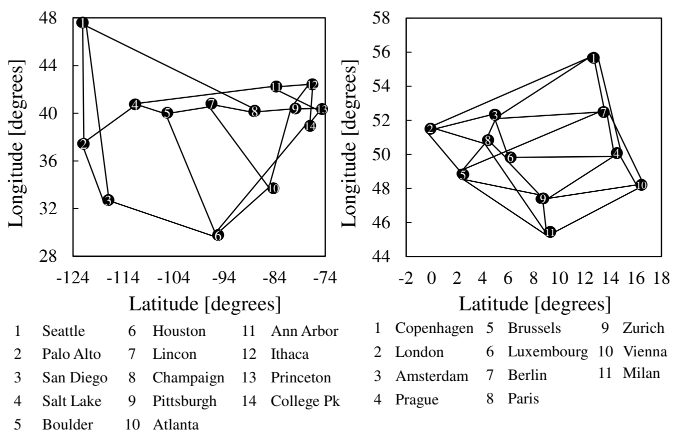


Fig. 2: Full procedure of proposed scheme



(a) NSFNET (b) COST239

Fig. 3: NSFNET and COST239

route. Because the number of checks varies depending on the location of the faulty link, we select the faulty links so that each route of the flowchart in Fig.2 is followed. The problem formulated in Section III-D3 is solved using CPLEX [8].

B. Example of failure identification

Fig. 4 shows an example of identifying the failure link in COST239. First, in Step 1, the optimal number of divisions of the COST239 (a=3) is determined using (1a) based on the number of links in COST239 (26 links). In Step 2, the three check routes are determined using (3a)–(3k) based on the number of divisions (a=3) obtained by Step 1. In Step 3, a failure is detected only in the blue check route as shown in Fig. 2(a). Because the number of links (8 links) exceeds the number of divisions (a=3), go back to Step 1 and check the blue check route as the target network. A failure is detected only in the orange check route in Steps 1 to 3, as shown in Fig. 2(b). Because the number of links (three) is less than or equal to the number of divisions (a=3), go to Step 5, and all links are checked in the orange check route. Now, let us compare the number of error detection packet checks. The proposed scheme has 9 error detection packet checks, while the conventional HANMOC (all-link check) has 28 (the same as the number of links).

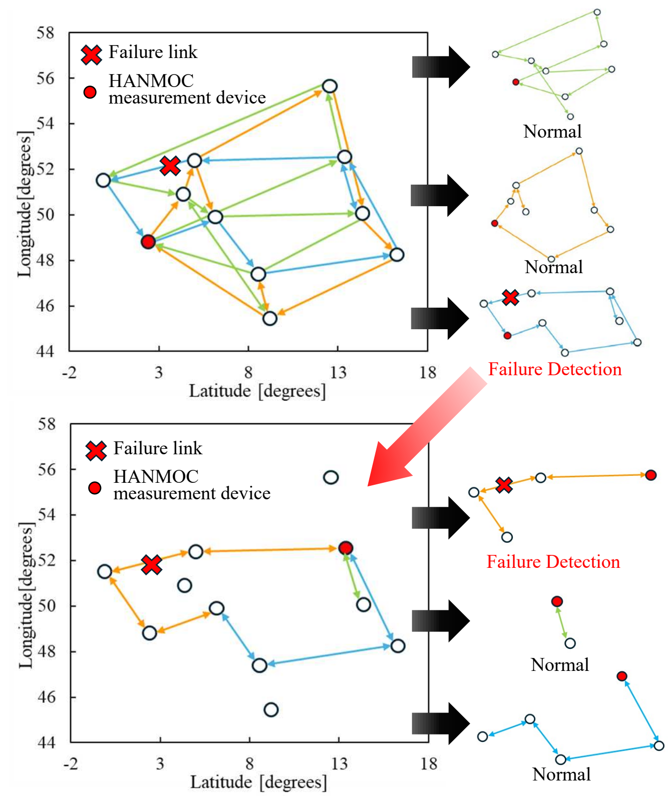
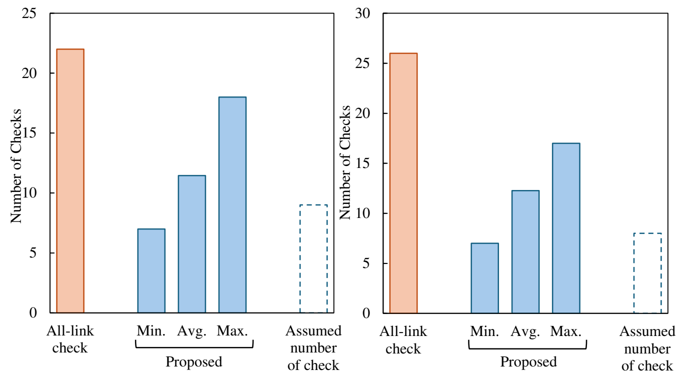


Fig. 4: Example of identifying failure link in COST239

C. Results

The numbers of checks for the conventional scheme (all-link check) and proposed scheme in NSFNET and COST239 are shown in Fig. 5. Table II shows the detailed results of the number of links per the number of checks. Here, “Number of Checks” denotes the number of checks performed until the failure link is identified, and “Count of Failure Links” represents the number of links that required the corresponding number of checks. For example, in NSFNET, there were two links that detected the failure with seven checks. The assumed number of checks in Fig. 5 is the minimum number of checks obtained in Section III-B. This value is treated as reference data for evaluation.



(a) NSFNET (b) COST239

Fig. 5: Number of checks identifying failure link

TABLE II: Distribution of number of checks

(a) NSFNET

Number of Checks	Count of Failure Links
7	2
8	1
9	3
10	8
13	2
14	1
15	1
16	1
17	2
18	1

(b) COST239

Number of Checks	Count of Failure Links
7	2
8	1
9	1
11	3
12	1
13	12
14	5
17	1

As shown in Fig. 5, In NSFENT and COST239, compared with the conventional scheme, the proposed scheme can detect failure links with an average of approximately 50% checks. Some links require many checks, for example, link 1-14 (link between nodes 1 and 14) in NSFNET needs 18 checks. This was because failures were frequently detected in areas with a high number of links within the divided areas, making it difficult to narrow down the area where the failure link existed. In COST239 (e.g., Link 3–5), the failure can be detected with fewer checks than the assumed number. This is because the process branched into Step 4-2 in Fig. 2, resulting in the check of overlapping links, and the failure link is identified while the number of divisions is still very small. This result is the effect of the first objective function, which minimizes the overlapping links.

V. Conclusion and Future Work

In this paper, we proposed a high-speed normality check scheme based on the lump link check for a silent failure, which is a failure that is not notified or causes only a large increase in latency. The proposed scheme is improved to check normality with the lump link check for divided networks to reduce the number of checks. The proposed scheme consists of two processes. The first process determines the number of divisions to minimize the number of checks from the number of links in the network. The second process is a check route decision process in each divided network based on the number of divisions obtained in the first process. The check route decision problem in the second process is formulated as an ILP problem improved from the VRP. In evaluating the proposed scheme in NSFNET and COST239, the number of checks to identify the failure link was compared with that of the all-link check. Numerical results show that in all failure patterns in both networks, the proposed scheme can reduce the average number of checks to approximately 50% compared to that of all-link check. These results indicate that introducing the proposed scheme will speed up normality checks in large-scale networks.

As future work, we plan to reduce the computation time (the computation time of the evaluation in Section IV takes roughly within 5 minutes) for applying larger-scale networks . We consider that by dividing the network into multiple regions in advance and running the proposed scheme in parallel for each region, it is possible to speed up the process through parallel processing. In addition, we will investigate extensions for scenarios where multiple link failures occur simultaneously or where only one-way link failures occur.

Acknowledgement

This work was supported by ROIS NII Open Collaborative Research 251S1-22586 and JSPS KAKENHI JP23K16867. The authors would like to thank the Information Media Center (IMC) at Toyohashi University of Technology for providing computer resources and the kind support of its staff.

References

[1] S. Previdi, B. Decraene, S. Litkowski, and R. Shakir, “RFC 8660: Segment routing with the MPLS data plane,” 2019.

[2] “High-accuracy network monitoring and control: HANMOC,” https://www.rd.ntt/e/research/NIC0008.html, online; accessed 3 April 2025.

[3] G. Laporte, and Y. Nobert, “A branch and bound algorithm for the capacitated vehicle routing problem,” Operations-Research-Spektrum, vol. 5, no. 3, pp. 77–85, 1983.

[4] K. C. Claffy, G. C. Polyzos, and H. W. Braun, “Traffic characteristics of the T1 NSFNET backbone,” *IEEE Infocom’93 The Conference on Computer Communications, Proceedings*, vol. 2, pp. 885–892, 1993.

[5] M. O’MAHONY, “Ultrahigh capacity optical transmission network: European research project cost 239,” *Information, Telecommunications, Automata Journal*, vol. 12, pp. 33–45, 1993.

[6] J. Moy, “RFC 2328: OSPF Version 2,” 1998.

[7] C. E. Miller, A. W. Tucker, and R. A. Zemlin, “Integer programming formulations and traveling salesman problems,” *Journal of the ACM (JACM)*, vol. 7, no. 4, pp. 326–329, 1960.

[8] “IBM ILOG CPLEX,” https://www.ibm.com/jp-ja/products/ilog-cplex-optimization-studio/cplex-optimizer, online; accessed 3 April 2025.