Traveling Repairman Problem for Optical Network Recovery to Restore Virtual Networks After a Disaster [Invited]

Chen Ma, Jie Zhang, Yongli Zhao, M. Farhan Habib, S. Sedef Savas, and Biswanath Mukherjee

***Abstract*—Virtual networks mapped over a physical network can suffer disconnection and/or outage due to disasters. After a disaster occurs, the network operator should determine a repair schedule and then send repairmen to repair failures following the schedule. The schedule can change the overall effect of a disaster by changing the restoration order of failed components. In this study, we introduce the traveling repairman problem to help the network operator make the schedule after a disaster. We measure the overall effect of a disaster from the damage it caused, and we define the damage as the numbers of disconnected virtual networks, failed virtual links, and failed physical links. Our objective is to find an optimal schedule for a repairman to restore the optical network with minimum damage. We first state the problem; then a mixed integer linear program (MILP) and three heuristic algorithms, namely dynamic programming (DP), the greedy algorithm (GR), and simulated annealing (SA), are proposed. Finally, simulation results show that the repair schedules using MILP and DP results get the least damage but the highest complexity; GR gets the highest damage with the lowest complexity, while SA has a good balance between damage and complexity.**

***Index Terms*—Disaster survivability; Optical network; Traveling repairman problem; Virtual networks.**

I. INTRODUCTION

Several recent disasters have shown that the risk of large-scale failures in communication networks due to disasters such as earthquakes, hurricanes, and tsunamis is on the rise [1–3]. For instance, around 30,000 km of optical fiber and 4000 telecommunications offices were destroyed by the 2008 China Sichuan Earthquake [3], and more than 2200 telecom buildings were damaged in the 2011 Great East Japan Earthquake and Tsunami [4]. After a disaster, how to restore network connectivity/capacity, especially in the virtual layer, is an important issue for network operators.

In the recent past, several large telecom companies have established disaster management systems and trained repairmen for failures [5–7]. For instance, AT&T's Network Disaster Recovery (NDR) program was formed to respond to natural and man-made disasters that may damage telecom networks [7]. This program includes support trailers, recovery engineering software applications, and a response team with both full-time and volunteer members. After a disaster occurs, telecom offices need to quickly determine the location and type of failures, and make a recovery schedule for the repairmen. Then, the repairmen are sent to repair the damaged network elements with recovery vehicles and equipment.

A disaster may cause failures of multiple network elements, and they cannot be restored at the same time because of the limitation of the recovery resources, such as repairmen, recovery vehicles, and repaired network materials for replacement. Network operators tend to divide the disaster area into several subareas with multiple failures, and each repairman is in charge of one area [8]. Thus, the repair process employs multiple steps after a major disruption [8–11]. Although much work has been done by network operators regarding equipment and repairman training, how to optimally make the repair schedule after a disaster is an open problem. Note that a disaster may affect both communications and transportation networks. In the Sichuan Earthquake, some recovery vehicles were not able to reach failure locations, and repairmen went there on foot because roads were damaged, while for some mountain areas without ground access after the earthquake, army helicopters transported repairmen and equipment [3]. Hence, the repair schedule must consider the travel time in the disaster zone, as it may vary and affect the repair schedule optimization decisions.

With the introduction of network virtualization, virtual network requests are more prevalent in optical networks [12–14], and the damage to an optical network with virtual networks can be more destructive than before. In a conventional network, damage can be measured by the amount of disrupted end-to-end connection requests [8–10]. But in the

Manuscript received June 2, 2015; revised August 28, 2015; accepted August 31, 2015; published October 9, 2015 (Doc. ID 242161).

C. Ma (e-mail: machen1888@bupt.edu.cn) is with the State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China. He was a visiting student when this work was done at the University of California, Davis, California 95616, USA.

J. Zhang and Y. Zhao are with the State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China.

M. F. Habib is with Intel, Folsom, California 95630, USA.

S. S. Savas and B. Mukherjee are with the Networks Lab at the University of California, Davis, California 95616, USA.

http://dx.doi.org/10.1364/JOCN.7.000B81

physical network that supports network virtualization, even a single failure may cause multiple failures in overlay networks, and disconnect them. Thus, damage metrics for network virtualization should consider the damage to both the physical network and virtual networks caused by a disaster.

In this study, we focus on the traveling repairman problem (TRP) for the network operator to schedule repair order after a disaster. Our objective is to minimize the total damage in an optical network caused by a disaster failure in terms of connectivity loss and capacity loss by providing a repair schedule that considers both the repair time and the time spent to reach the disaster zone. We model the TRP in optical networks and calculate the damage cost in terms of disconnected virtual networks (DVNs), failed virtual links (FVLs), and failed physical links (FPLs).

An auxiliary graph is introduced to simplify the problem. In the auxiliary graph, nodes are the failure locations in the optical network, which are related to the failed links of virtual networks, while the weights of links are the travel time between two failure locations and the repair time of the failure. The output of the TRP is the optimal repair schedule of the repairman that results in minimum damage. Based on the TRP model, a mixed integer linear program (MILP), the greedy algorithm (GR), dynamic programming (DP), and simulated annealing (SA) are all proposed for the problem.

The rest of this study is organized as follows. Section II reviews related work. We model the problem, introduce the auxiliary graph, and propose metrics to calculate the damage in Section III. The MILP formulation for the TRP is introduced in Section IV. In Section V, we present the proposed heuristic algorithms. Section VI presents the simulation results, and Section VII concludes the study.

II. Related Work

In telecom networks, various methods are proposed to improve their survivability against disasters, which can be categorized as protection and restoration. In protection, backup paths are precalculated, and backup resources are preallocated for the requests [15–17], while in restoration, backup paths are computed and established after a disaster occurs [18–20]. Although restoration technologies try to reprovision connections and virtual networks with the remaining resources, the scarce resources may not be enough to fully recover them, if the network is not overprovisioned enough. Thus, a solid repair plan is required to restore the network.

To the best of our knowledge, few works have addressed repair schedules for telecom networks. In [8], an efficient recovery schedule was proposed to determine the best repair sequence for point-to-point requests. Reference [9] introduced a schedule for a worst case, where recovered resources are not enough for all the failures, so the network operator should decide which failures to repair to minimize the damage of connections. In [10], the schedule not only repairs the failures but also reprovisions new lightpaths for disrupted connections. But the schemes in [8–10] have the same drawback that the time consumption of each failure is fixed. In practice, the time consumption of each failure consists of travel time and repair time. Travel time depends on the distance between vehicle locations and the failure zones, road conditions, and so on [3]. Repair times vary between 2 and 24 h based on the type of failures [21–23]. Hence, time consumption is a very important factor to determine the repair schedule.

The traveling salesman problem (TSP) is a graph problem, whose main idea is to find the shortest route that visits each node once and returns to the origin node, and its complexity is proved to be NP-hard [24–26]. Over the past few decades, several solutions have been proposed for the TSP, such as the tour improvement process, dynamic programming, the genetic algorithm, and simulated annealing [27–29]. The vehicle routing problem (VRP), which is also named the delivery man problem [30–32] and TRP [33–35], is also a graph problem. Its main idea is to find an optimal route to travel to all the nodes in the graph with minimum damage, and the solutions consider dynamic and static scenarios [36–40]. However, the cost of each node is fixed in TSP and VRP, while in network virtualization, the benefit of repairing each node can vary in the repair process. Hence, in this study, we define a new problem, which also considers the network virtualization aspects.

III. Traveling Repairman Problem

In this section, we model the TRP for network virtualization, and then propose three metrics to evaluate the damage in the network.

A. Problem Statement

In normal operation, virtual networks are already provisioned on the physical network. Each virtual node from the same virtual network request is assigned to a different physical node by a mapping scheme. Each virtual link is mapped to a physical path between the corresponding physical nodes that host the virtual end nodes of that virtual link.

Several physical links may get damaged due to a disaster, which may affect the carried virtual links, and hence affect the virtual networks. Repairmen are needed to restore physical failures caused by the disaster.

Then, TRP is described as follows: the network operator should determine a repair schedule for a repairman after a disaster occurs. The schedule determines the steps of repair starting from the first node visited by the repairman, and gives a sequence of repairs. The objective of this schedule is to minimize the overall damage caused by the disaster. Assuming transportation by road and one repairman, TRP can be formally stated as follows.

Input:

- G^T: Transportation network, which shows distance and time between vehicle and disaster zone.

- $G^P(N^P, L^P)$: Physical network, carrier of virtual networks, where N^P is the set of physical nodes and L^P is the set of physical links.
- $G_s^V(N_s^V, L_s^V)$: Virtual network, requested virtual network topology, where s is the index of a virtual network, and N_s^V and L_s^V are sets of virtual nodes and links in the sth virtual network. G^V denotes a set of S virtual networks.
- $G^D(N^D, L^D)$: Disaster zone, which is destroyed by the disaster. N^D and L^D are sets of failed nodes and links in the area. G^D is a subgraph of G^P, where $N^D \subset N^P$ and $L^D \subset L^P$.
- n_I^P: Physical node in physical network, where $n_I^P \in N^P$.
- l_{IJ}^P: Physical link with two physical end nodes n_I^P and n_J^P, where $l_{IJ}^P \in L^P$.
- $n_{i,s}^V$: Virtual node in G_s^V, where $n_{i,s}^V \in N_s^V$.
- $l_{ij,s}^V$: Virtual link with two virtual end nodes $n_{i,s}^V$ and $n_{j,s}^V$, where $l_{ij,s}^V \in L_s^V$.

Output:

- Recovery schedule with minimum damage.

Figure 1 shows an illustration of TRP. The optical network G^V (solid lines) and transportation network (dashed lines) G^T coexist in an area such as in Fig. 1(b). We assume that fiber optic cables are laid along rights-of-way, such as highways and railroads. Before a disaster occurs, three virtual networks G_1^V, G_2^V, and G_3^V in Fig. 1(a) are mapped to the physical network G^P in Fig. 1(b) by the mapping relationship in Table I. The disaster zone in the network is the solid line oval, where l_{BE}^P, l_{BF}^P, and l_{CE}^P are FPLs, and P_1, P_2, and P_3 are the exact positions of the failures. FPLs cause the failures of virtual links; e.g., $l_{ab,1}^V$ in G_1^V is affected by the failure of l_{BF}^P. After the disaster occurs, the repairman leaves his office in node n_A^P to repair all the failed links. One possible repair schedule is $A \to P_1 \to P_2 \to P_3$, where the travel time and repair time are shown in Table II.

In the TRP, we assume there are K failures after a disaster occurs, and the number of candidate repair schedules is $K!$. The network operator should select the optimal schedule from the candidates. Hence, the time complexity of the TRP is $K!$, which is a NP-hard problem.

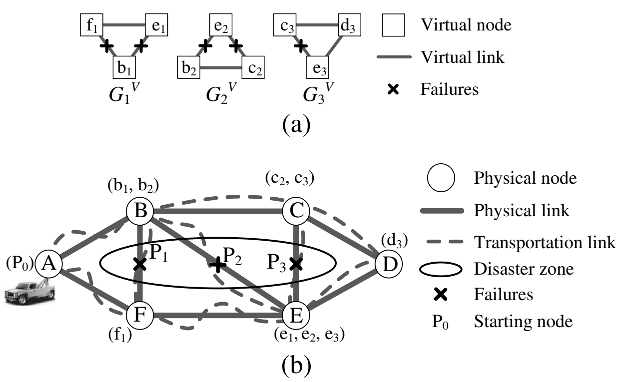

Fig. 1. Illustration of TRP. (a) Virtual networks G^V. (b) Physical network G^P and transportation network G^T.

TABLE I
VIRTUAL MAPPING OF ILLUSTRATION

VNs	Virtual Link	Physical Link	Failure Location
G_1^V	$f_1 - b_1$	F-B	P_1
	$b_1 - e_1$	B-E	P_2
	$e_1 - f_1$	E-F	
G_2^V	$b_2 - e_2$	B-E	P_2
	$e_2 - c_2$	E-C	P_3
	$c_2 - b_2$	C-B	
G_3^V	$c_3 - e_3$	C-E	P_3
	$e_3 - d_3$	E-D	
	$d_3 - c_3$	D-C	

TABLE II
TIME CONSUMPTION OF ILLUSTRATION

Notations	Time	Work
t_1^T	1.5 h	Travel path $P_0(A) \to P_1$
t_1^R	2.2 h	Repair P_1
t_2^T	1.2 h	Travel path $P_1 \to P_2$
t_2^R	2.4 h	Repair P_2
t_3^T	2.0 h	Travel path $P_2 \to P_3$
t_3^R	2.0 h	Repair P_3

B. Auxiliary Graph

We assume that the network operator knows the exact location of failed links, the travel time between any two failure locations, and the mapping relationships of the physical network and existing virtual networks. Based on this information, we can merge the transportation network, physical network, and disaster network into an auxiliary network, which is described as follows:

Auxiliary graph $G^A(N^A, L^A)$ is a fully connected graph that shows the points the repairman needs to visit. Node P_0 in N^A is the office of the repairman, such as node n_A^P in Fig. 1(b), which is also his starting node. Other nodes represent failure locations in the disaster area. A link in the auxiliary graph is unidirectional, whose weight is the sum of the travel time to the failure location and the repair time of each failure.

The mapping relationship of the auxiliary graph and the virtual network can be derived from the relationship of G^V and G^P, and G^P and G^A. One virtual link is mapped to a set of physical links; one FPL is mapped to one auxiliary node; hence, one FVL is mapped to a set of auxiliary nodes.

An illustration of the auxiliary graph is shown in Fig. 2. Node P_0 in Fig. 2 is mapped from node n_A^P in Fig. 1(b). Auxiliary nodes P_1, P_2, and P_3 are the failure locations in the transportation network. Since virtual link $l_{ab,1}^V$ is mapped to physical link l_{FB}^P, and l_{FB}^P is related to auxiliary node P_1, $l_{ab,1}^V$ is mapped to P_1. The relationship between virtual links and auxiliary nodes is shown in Table I. The weights of auxiliary links are decided by the travel time and the repair time of failure in the destination node. Take

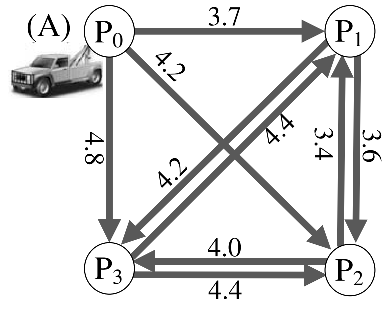

Fig. 2. Auxiliary graph.

$P_1 \to P_2$ for instance, where the travel time from P_1 to P_2 is 1.2 h, and the repair time of P_2 is 2.4 h; hence, the weight of $P_1 \to P_2$ is $1.2 + 2.4 = 3.6$ h. For $P_2 \to P_1$, its weight is $1.2 + 2.2 = 3.4$ h, which is different with $P_1 \to P_2$.

Finally, the TRP can be redefined as follows: find an optimal repair scheme with minimum damage, which travels through all the nodes in the auxiliary graph.

C. Survivability Metrics for Optical Networks

In the conventional scheme, the repair times of various failures are assumed to be fixed; hence the damage is mainly related to the number of failed requests in each step [7]. But in network virtualization, the damage of virtual networks is different from the conventional scheme in two aspects. First, the time consumption of each step is different depending on the travel time and repair time. Second, the influence of failed virtual networks is different and should be evaluated at two levels. At link level, the damage is related to the failed links, such as the five failed links in Fig. 1(a), and at network level, the damage is related to the number of DVNs, such as G_1^V and G_2^V in Fig. 1(a). In some situations, several physical links that do not carry virtual links are also damaged by the disaster. Thus, the damage of virtual networks should include DVNs, FVLs, and FPLs. The objectives of TRP are shown in descending order in terms of importance:

- Minimize damage caused by the DVN;
- Minimize damage caused by the FVL;
- Minimize damage caused by the FPL.

Now, we first introduce the damage caused by the DVN. There are K FPLs caused by a disaster, so the repairman needs K steps to repair the optical network. We define that the damage c_k^{DVN} of the DVN in step $k \in \{1, 2, \ldots, K\}$ is the product of the number of DVNs $\omega_{k-1}^{\mathrm{DVN}}$ and the time duration of the steps, which shows the overall damage as

$$c_k^{\mathrm{DVN}} = \omega_{k-1}^{\mathrm{DVN}} \times (t_k^T + t_k^R). \tag{1}$$

The total damage caused by the DVN in the entire repair process is the sum of the damage caused by the DVN in each step, i.e.,

$$C^{\mathrm{DVN}} = \sum_{k \in \{1,\ldots,K\}} c_k^{\mathrm{DVN}} = \sum_{k \in \{1,\ldots,K\}} \omega_{k-1}^{\mathrm{DVN}} \times (t_k^T + t_k^R). \tag{2}$$

For instance, we consider that the repair schedule $P_0 \to P_1 \to P_2 \to P_3$ in Figs. 2 and 3 shows the entire repair process. From Fig. 3, the number of DVNs is two, and the duration is 3.7 h in the first step, so the damage caused by the DVNs is $c_1^{\mathrm{DVN}} = 2 \times 3.7 = 7.4$ in this step. Similarly, the damage caused by the DVNs is $1 \times 3.6 = 3.6$ and 0×4 in the second and third steps, respectively. Hence, the total damage caused by the DVN is $7.4 + 3.6 + 0 = 11.0$ in this example.

Similarly, damages of FVL and FPL are shown below:

$$c_k^{\mathrm{FVL}} = \omega_{k-1}^{\mathrm{FVL}} \times (t_k^T + t_k^R), \tag{3}$$

$$C^{\mathrm{FVL}} = \sum_{k \in \{1,\ldots,K\}} c_k^{\mathrm{FVL}} = \sum_{k \in \{1,\ldots,K\}} \omega_{k-1}^{\mathrm{FVL}} \times (t_k^T + t_k^R), \tag{4}$$

$$c_k^{\mathrm{FPL}} = \omega_{k-1}^{\mathrm{FPL}} \times (t_k^T + t_k^R), \tag{5}$$

$$C^{\mathrm{FPL}} = \sum_{k \in \{1,\ldots,K\}} c_k^{\mathrm{FPL}} = \sum_{k \in \{1,\ldots,K\}} \omega_{k-1}^{\mathrm{FPL}} \times (t_k^T + t_k^R). \tag{6}$$

IV. MILP Model

In this section, we formulate the repair scheduling problem, and it turns out to be an MILP.

A. Notations and Variables

The notations and variables are introduced as follows.

Notations:

- $G^A(N^A, L^A)$: Auxiliary network, where N^A and L^A are sets of auxiliary nodes and auxiliary links.
- $G^V = \{G_1^V, \ldots, G_S^V\}$: Set of virtual networks existing in the network.

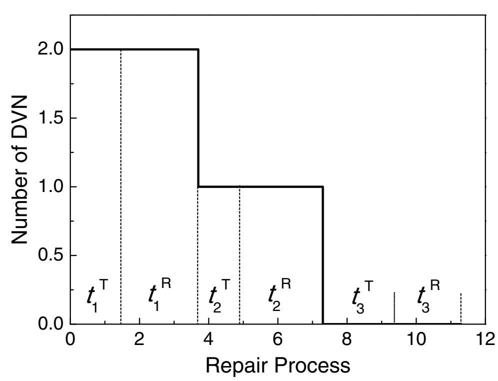

Fig. 3. Damage caused by the DVN.

- $G_s^V(N_s^V, L_s^V)$: sth virtual network, where N_s^V and L_s^V are sets of virtual nodes and links, respectively.
- t_{uv}: Weight of auxiliary link $P_u \to P_v$, which is the sum of the travel time from P_u to P_v and the repair time of P_v.
- Π^{AV}: Relationship of G^A and G^V. If $\pi^{AV}_{u,(i,j),s} = 1$, virtual link $l^V_{ij,s}$ is mapped to failure P_u in N^A; 0, otherwise.
- K: Number of FPLs.
- S: Number of virtual networks.
- $P_0 \in N^A$: Start node of repairman.

Variables:

- C: Integer variable, total damage in the whole process.
- C^{DVN}: Integer variable, damage caused by the DVN.
- C^{FVL}: Integer variable, damage caused by the FVL.
- C^{FPL}: Integer variable, damage caused by the FPL.
- $\phi^A_{u,k}$: Binary variable, repair status of auxiliary node P_u in step k. If $\phi^A_{u,k} = 1$, P_u is repaired in step k; 0, otherwise.
- t_k: Nonnegative variable, which is the time consumption in step k.
- $h_{u,k}$: Binary variable, status of auxiliary node P_u in step k. If $h = 1$, P_u is active in step k; 0, otherwise.
- $\eta_{(i,j),s,k}$: Binary variable, status of virtual link $l^V_{ij,s}$ in step k. If $\eta = 1$, $l^V_{ij,s}$ in step k is active; 0, otherwise.
- $x_{(p,q),(i,j),s,k}$: Binary variable. If $x = 1$, the path from virtual node $n^V_{p,s}$ to $n^V_{q,s}$ uses virtual link $l^V_{ij,s}$ in step k; 0, otherwise.
- $y_{(p,q),(i,j),s,k}$: Binary variable. If $y = 1$, the path from virtual node $n^V_{p,s}$ to $n^V_{q,s}$ uses virtual link $l^V_{ij,s}$ and $l^V_{ij,s}$ is disconnected in step k; 0, otherwise.
- $r_{s,k}$: Binary variable. If $r = 1$, the sth virtual network is disconnected in step k; 0, otherwise.

B. Objective

For the MILP, we use Eq. (7) to express the total damage in the optical network, and our objective is to minimize the total damage:

$$\min C = \min(\alpha C^{\mathrm{DVN}} + \beta C^{\mathrm{FVL}} + \gamma C^{\mathrm{FPL}}), \tag{7}$$

where α, β, and γ are weights of damage with DVN, FVL, and FPL, which can be dynamically adjusted by the network operator.

C. Constraints

The constraints are divided into four parts. The first is to find the path on G^A, and the other three minimize the total damage caused by DVNs, FVLs, and FPLs, respectively.

1) *Finding Path on G^A:*

$$\phi^A_{u=0,k=0} = 1, \tag{8}$$

$$\sum_{P_u \in N^A} \phi^A_{u,k} = 1 \quad \forall\, k \in \{0, \ldots, K\}, \tag{9}$$

$$\sum_{k \in \{0,\ldots,K\}} \phi^A_{u,k} = 1 \quad \forall\, P_u \in N^A. \tag{10}$$

In this model, we need to find the repair sequence of auxiliary nodes, because G^A is a fully connected graph; i.e., an auxiliary link connects any node pair. Constraint (8) denotes that the first point is the start position of the repairman. Constraint (9) ensures that, in each step k, one failure is repaired. Constraint (10) denotes that one failure is repaired one time in the entire process.

$$t_k = \sum_{P_u \in N^A} \sum_{P_v \in N^A} t_{uv} \times (\phi^A_{u,k} \wedge \phi^A_{v,k-1}) \quad \forall\, k \in \{1, \ldots, K\}. \tag{11}$$

Then, we determine the time of step k by constraint (11).

2) *Determining C^{FPL}:*

$$C^{\mathrm{FPL}} = \sum_{k \in \{1,\ldots,K\}} \sum_{P_u \in N^A} \phi^A_{u,k-1} \times t_k. \tag{12}$$

In each step, only one FPL can be repaired; hence, the total damage caused by the FPL (C^{FPL}) is calculated by Eq. (12).

3) *Determining C^{FVL}:*

$$h_{u,k} = \sum_{k' \in \{0,\ldots,k\}} \phi^A_{u,k'} \quad \forall\, P_u \in N^A, k \in \{0, \ldots, K\}. \tag{13}$$

The damage caused by the FVL (C^{FVL}) includes the number and duration of inactive virtual links. Constraint (13) ensures that the failure P_u is repaired.

$$\eta_{(i,j),s,k} = \bigvee_{P_u \in N^A} (1 - h_{u,k}) \wedge \pi^{AV}_{u,(i,j),s} \quad \forall\, n^V_{i,s}, n^V_{j,s} \in N^V_s, s \in \{1, \ldots, S\}, k \in \{1, \ldots, k\}. \tag{14}$$

Then, constraint (14) finds the status of virtual link $l^V_{ij,s}$ in the sth virtual network.

$$C^{\mathrm{FVL}} = \sum_{s \in \{1,\ldots,S\}} \sum_{l^V_{ij,s} \in L^V_s} \sum_{k \in \{1,\ldots,K\}} \eta_{(i,j),s,k-1} \times t_k. \tag{15}$$

The damage caused by the FVL is obtained by constraint (15).

4) *Determining C^{DVN}:*

$$\sum_{l^V_{ij,s} \in L^V_s} x_{(p,q),(i,j),s,k} - \sum_{l^V_{ij,s} \in L^V_s} x_{(p,q),(j,i),s,k} = \begin{cases} 1 & j = p \\ -1 & j = q \\ 0 & \text{otherwise} \end{cases} \quad \forall\, n^V_{p,s}, n^V_{q,s}, n^V_{i,s} \in N^V_s, k \in \{1, \ldots, K\}, \tag{16}$$

$$y_{(p,q),(i,j),s,k} = x_{(p,q),(i,j),s,k} \wedge \eta_{(i,j),s,k} \quad \forall\, n^V_{p,s}, n^V_{q,s} \in N^V_s, l^V_{ij,s} \in L^V_s, k \in \{1,\ldots,K\}, \tag{17}$$

$$r_{s,k} = \bigwedge_{n^V_{p,s}, n^V_{q,s} \in N^V_s, l^V_{ij,s} \in L^V_s} y_{(p,q),(i,j),s,k} \, \forall\, s \in \{1,\ldots,S\}, k \in \{1,\ldots,K\}. \tag{18}$$

The damage caused by the DVN in the entire process is based on the status of virtual networks. Constraint (16) finds the path from virtual node p to q of the sth virtual network in step k by flow conservation. Constraint (17) ensures that the path is connected. Constraint (18) denotes that the sth virtual network is connected.

$$C^{\mathrm{DVN}} = \sum_{k\in\{1,\ldots,K\}} \sum_{s\in\{1,\ldots,S\}} r_{s,k-1} \times t_k. \tag{19}$$

Finally, the total damage caused by the DVN is calculated by Eq. (19).

V. Heuristic Algorithms

In this section, three heuristic algorithms are proposed to solve the TRP for large problem instances for which MILP is not suitable. They are the greedy algorithm (GR), dynamic programming (DP), and simulated annealing (SA) methods. Then, we analyze their time complexities.

A. Greedy Algorithm

The main idea of GR is to find the minimum damage in each step. From Eqs. (1), (3), and (5), the damage of step k by repair of failure P_u can be determined by Eq. (20):

$$c_{k,u} = \alpha c^{\mathrm{DVN}}_{k,u} + \beta c^{\mathrm{FVL}}_{k,u} + \gamma c^{\mathrm{FPL}}_{k,u}. \tag{20}$$

ϕ_k denotes the repaired physical link in the kth step, Φ_k denotes the repair scheme from step 0 to step k, and Φ_K is the final repair scheme. First, damages of all auxiliary nodes $P_u \in N^A$ are calculated, and the node with minimum damage is selected as the first repaired failure and added to Φ_0. In step k, node $P_u \in N^A - \Phi_{k-1}$ with minimum damage is the repaired failure and is added to Φ_{k-1}. Finally, all the failures are repaired after K steps by GR. The pseudo code of GR is shown in Algorithm 1.

Algorithm 1 Greedy Algorithm

Input:
- $G^A(N^A, L^A)$: Auxiliary network;
- G^V: Set of virtual networks;
- Π^{VA}: Relationship of auxiliary and virtual network;
- P_0: Start node of repairman;
- $\Phi_0 = \{P_0\}$: Original status of repair scheme;
- M: A large number (e.g., 10^5).

Output:
- Φ_K: Restored sequence of FPLs, where ϕ_k is the number of restored failure in step k.

```
for all k ∈ {1, …, K} do
    c_k = M, φ_k = null
    for all P_v ∈ N^A do
        Calculate c_{k,v} by G^A, G^V, and Π^{VA}
        if c_{k,v} < c_k then
            φ_k = P_v, c_k = c_{k,v}
        end if
    end for
    Φ_k = Φ_{k−1} + φ_k
    N^A = N^A − φ_k
end for
return Φ_K
```

B. Dynamic Programming

Dynamic programming (DP) is a method for solving complex problems by breaking them down into simpler subproblems. DP can be used to solve the TRP also.

Here, ϕ_k denotes the repaired physical links in step k. $\Phi_{k,v}$ denotes the first k steps of a repair scheme with total damage MC_k, and auxiliary node P_v is the latest repaired in $\Phi_{k,v}$, i.e., $\phi_k = P_v$. In the $k+1$-th step, P_u is repaired, and the damage from P_v to P_u is c_{uv}. Hence, the total damage of step $k+1$ is $MC_k + c_{uv}$. Then, we analyze the damage of DP in each step.

In the first step, the repairman leaves P_0 to P_u. The damage of the first step is

$$MC_1[P_u, \Phi_{0,0}] = c_{u,0} \quad \forall\, P_u \in N^A. \tag{21}$$

In the second step, failure P_v is repaired after P_u, and the damage of the second step is determined by Eq. (22):

$$MC_2[P_u, \Phi_{1,v}] = MC_1[P_v, \Phi_{0,0}] + c_{uv} \quad \forall\, u, v \in \{1,\ldots,K\}. \tag{22}$$

In the kth ($k \in \{3,\ldots,K\}$) step, the damage of repaired failure P_u can be calculated by the damage of repair scheme $\Phi_{k-1,v}$ and c_{uv} as shown in Eq. (23):

$$MC_k[P_u, \Phi_{k-1,v}] = \min_{P_d\in\Phi_{k-1,v}} \{MC_{k-1}[P_d, \Phi_{k-1,v} - P_d] + c_{du}\} \quad \forall\, k \in \{1,\ldots,K\}. \tag{23}$$

Finally, we calculate the result in the Kth step, and return the sequence with minimum damage. The pseudo code of DP is shown in Algorithm 2.

All possible options are exhaustively searched in DP, which gives us the optimal result in the end (which is the same as the MILP result).

Algorithm 2 Dynamic Programming

Input:
- $G^A(N^A, L^A)$: Auxiliary network;
- G^V: Set of virtual networks;
- Π^{VA}: Relationship of auxiliary and virtual network;
- P_0: Start node of repairman;
- $\Phi_{0,0} = \{P_0\}$: Original status of repair scheme.

Output:
- Φ_K: Restored sequence of FPLs, where ϕ_k is the number of restored failure in step k.

for all $P_u \in N^A$ **do**
  Calculate $c_{u,0}$ by G^A, G^V, and Π^{VA}
  Calculate $MC_0[P_u, \Phi_{0,0}]$ by Eq. (21)
end for
for all $P_u \in N^A$ **do**
  for all $P_v \in N^A - P_u$ **do**
    Calculate $c_{u,v}$ by G^A, G^V, and Π^{VA}
    Calculate $MC_2[P_u, \Phi_{1,v}]$ by Eq. (22)
  end for
end for
for all $k \in \{3, \ldots, K\}$ **do**
  for all $P_u \in N^A$ **do**
    for all $P_v \in N^A - P_u$ **do**
      Calculate $c_{u,v}$ by G^A, G^V, and $\pi^{VA}_{N,(i,j),s}$
      Calculate $MC_k[P_u, \Phi_{k-1,v}]$ by Eq. (23)
    end for
  end for
end for
$\Phi_K = \Phi_{K,u}$ with minimum damage C $\forall\, P_u \in N^A$
return Φ_K

C. Simulated Annealing

Although DP can obtain the optimal repair scheme for the repairman, the complexity of DP is high, so we introduce simulated annealing (SA) to determine a possible suboptimal solution quickly. First, we find an initial repair sequence Φ_{initial}, and then we preconfigure the initial temperature Θ_{initial} and stop temperature Θ_{stop}. A sequence set S_Φ is used to record the candidate solutions for the repairman. Φ_{temp} and C_{temp} are used to record the temporary sequence and its damage.

For temperature Θ_{temp} with damage C_{temp}, we randomly generate a new sequence Φ_{new} with damage C_{new}, and then compare it with Θ_{temp}. The acceptance probability AP_{temp} is defined as

$$AP_{\text{temp}} = \begin{cases} \exp(-\Delta C/\Theta_{\text{temp}}) & \Delta C > 0 \\ 1 & \Delta C \le 0 \end{cases}, \tag{24}$$

where $\Delta C = C_{\text{new}} - C_{\text{temp}}$. Then, a random value $\text{random}(0,1)$ between 0 and 1 is uniformly generated. If $\text{random}(0,1) > AP_{\text{temp}}$ or $AP_{\text{temp}} = 1$, sequence Φ_{new} is used as the candidate solution.

Two loops are considered in the algorithm. For the internal loop, if continuous $\omega_{\text{in}}^{\text{stop}}$ sequences are not accepted as the candidates, new temperature Θ_{new} will be generated by the current temperature Θ_{temp} and cooling function as shown in Eq. (25):

$$\Theta_{\text{new}} = \lambda \times \Theta_{\text{temp}}, \tag{25}$$

where λ is the weight of cooling, whose range is (0,1].

For the external loop, if continuous $\omega_{\text{ex}}^{\text{stop}}$ temperatures do not have solutions or $\Theta_{\text{temp}} < \Theta_{\text{stop}}$, the process of SA is finished.

Finally, we choose the sequence $\Phi_{\min}$ with minimum damage in S_Φ as the solution for the repairman. The pseudo code of SA is shown in Algorithm 3.

Algorithm 3 Simulated Annealing

Input:
- $G^A = (N^A, L^A)$: Auxiliary network;
- G^V: Set of virtual networks;
- Π^{VA}: Relationship of auxiliary and virtual network;
- P_0: Start node of repairman;
- $\Theta_{initial}$ and Θ_{stop}: Initial and stop temperatures;
- ω_{ex}^{stop} and ω_{in}^{stop}: continuous sequence numbers of external and internal loops.

Output:
- Φ_K: Restored sequence of FPLs, where ϕ_k is the number of restored failure in step k.

Get $\Phi_{initial}$, $C_{initial}$ from GR
Initial Φ_{temp}, C_{temp}, Θ_{temp}, S_Θ, ω_{ex}, and ω_{in}
while $(\Theta_{temp} > \Theta_{stop} || \omega_{ex} < \omega_{ex}^{stop})$ **do**
  $\omega_{in} = 0$, $F_{ex} = true$
  while $\omega_{in} < \omega_{in}^{stop}$ **do**
    Randomly select ϕ_m and ϕ_n in Φ_{temp} to get Φ_{new}
    Calculate damage of new scheme C_{new} and ΔC
    Calculate AP by Eq. (24)
    if $\Delta C \le 0 || AP > random(0,1)$ **then**
      $S_\Phi += \Phi_{new}$, $\Phi_{temp} = \Phi_{new}$, $C_{temp} = C_{new}$, $\omega_{in} = 0$, $\omega_{ex} = 0$, $F_{ex} = false$.
    else
      $\omega_{in}++$
    end if
  end while
  if $F_{ex} == true$ **then**
    $\omega_{ex}++$
  end if
  $\Theta_{temp} = \lambda \times \Theta_{temp}$
end while
$\Phi_K = \{\Phi \in S_\Phi$ with minimum damage $C\}$
return Φ_K

D. Time-Complexity Analysis

In this subsection, we analyze the time complexities of GR, DP, and SA.

1) Time Complexity of GR: For GR, two loops are used in the algorithm. The first loop decides the kth node in the sequence, and K failures in the entire network. In the second loop, we calculate the damage to all the other failure locations, and then find the one with minimum cost. In the

kth step, $k-1$ nodes have been decided, and $K-(k-1)$ nodes are calculated in this step. The total time of GR is calculated by Eq. (26):

$$T^{\mathrm{GR}} = \sum_{k=1,\ldots,K} K \times (K-k+1) \times t = (K(K+1)/2)t. \quad (26)$$

Hence, the complexity of GR is $O(K^2)$.

2) Time Complexity of DP: For DP, the total time consumption is calculated by each step. In the first step, there are K possibilities for the first repair node, so the time is $K \times t$. For the second step, the possibility is $K \times (K-1)$, which means that the total time of the second step is $K \times (K-1) \times t$. For the kth ($k \geq 3$) step, the total time is $K \times \binom{k-1}{K-1} \times (k-1) \times t$. Then, the total time of DP is obtained by Eq. (27):

$$T^{\mathrm{DP}} = K(K-1)t + K(K-1)t + \sum_{k \in \{3,\ldots,K\}} K \binom{k-1}{K-1} (k-1)t = (K + K(K-1)2^{K-2})t. \quad (27)$$

Hence, the complexity of DP is $O(K^2 2^{K-2})$.

3) Time Complexity of SA: For SA, it is hard to determine its exact complexity because its accepted probability is randomly generated [27], but we can qualitatively analyze the complexity of SA considering its convergence [30].

The total time consumption of SA is related to the start temperature Θ_{initial}, the stop temperature Θ_{stop}, and the annealing coefficient λ. Two loops are designed in the SA algorithm: the first finds the temporary temperature Θ_{temp}. We assume that the number of the external loop is ϵ_{ex}. For the external loop, we can use Eq. (28) to determine the ϵ_{ex}:

$$\Theta_{\text{initial}} \times \lambda^{\epsilon_{\text{ex}}-1} \leq \Theta_{\text{stop}} \leq \Theta_{\text{initial}} \times \lambda^{\epsilon_{\text{ex}}}. \quad (28)$$

For the internal loop, we use ϵ_{in} to present the number of sequences in the first step. Because the loop is converged, and the acceptance probability becomes smaller in other steps, for the other step, the number of sequences gets smaller than ϵ_{in}. Hence, the upper bound for SA can be determined by Eq. (29):

$$T^{\mathrm{SA}} \leq \epsilon_{\text{ex}} \times \epsilon_{\text{in}} \times K \times t. \quad (29)$$

Hence, the complexity of SA is $O(\epsilon K)$, where $\epsilon = \epsilon_{\text{ex}}\epsilon_{\text{in}}$.

VI. Illustrative Numerical Examples

In this section, we conduct simulations on the network topology as in Fig. 4. Fourteen nodes, nineteen links, and three disaster zones ($\{l^P_{DE}, l^P_{EF}, l^P_{EG}, l^P_{GH}, l^P_{GJ}\}$, $\{l^P_{AC}, l^P_{BC}, l^P_{CD}, l^P_{CF}, l^P_{EF}, l^P_{FJ}\}$, and $\{l^P_{FJ}, l^P_{GJ}, l^P_{IJ}, l^P_{JN}, l^P_{MN}\}$) are in the network. Because the results of the three disaster zones are similar, we only show the results of Disaster 1. Virtual networks are randomly generated with three nodes, and the probability of any two virtual nodes being connected is 60%. MILP, GR, DP, and SA with $\lambda = 0.9$ and $\lambda = 0.8$ are employed in the simulations. In this study, we do not generate the transportation network with length; instead we randomly generate the travel times between any two nodes in the auxiliary graph between 2 and 8 h. Now, we introduce the results of simulations in terms of damage versus number of virtual networks, and damage versus number of FPLs.

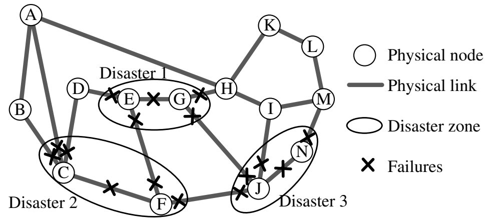

Fig. 4. Network topology with disaster.

A. Damage Versus Number of Virtual Networks

For the damage (Disaster 1) versus the number of virtual networks, five physical links are all failed at each time, and the number of virtual networks generated ranges from 10 to 30.

The total damage caused by the DVN increases with the number of virtual networks as shown in Fig. 5, because virtual networks are uniformly distributed in the physical network. Another conclusion is that the result of GR is the worst, and MILP and DP are the best. For SA, the damage caused by the DVN is less than that of GR because its initial sequence is the result of GR. Another conclusion is that the result of SA with higher λ is better than that with lower λ, because the higher one means more temperatures than the lower one, so the calculation sequences of the higher one are more than those with lower λ. Figure 6 shows that the average damage caused by the DVN for each network is stable, which is also because virtual networks are uniformly distributed in the optical network.

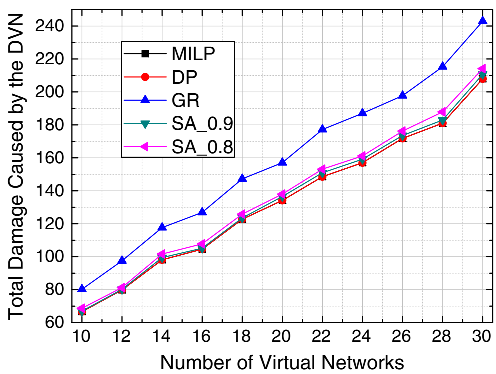

Fig. 5. Total damage caused by the DVN versus number of virtual networks.

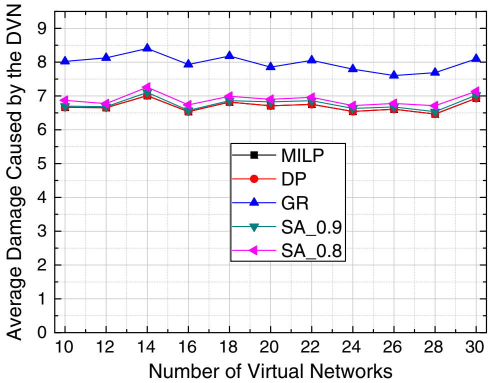

Fig. 6. Average damage caused by the DVN versus number of virtual networks.

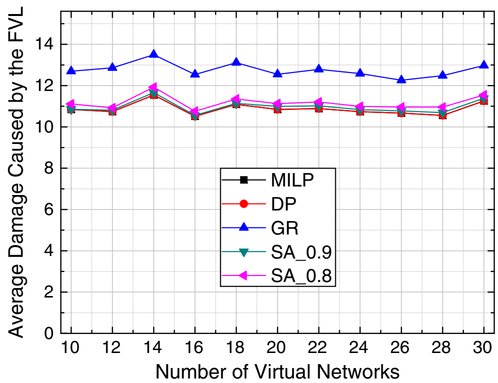

Fig. 8. Average damage caused by the FVL versus number of virtual networks.

The reason Fig. 6 does not show steady behavior is that the virtual networks are randomly generated and mapped over physical networks, and this randomization causes fluctuations in the graphs.

Figures 7 and 8 show the total damage caused by the FVL, and the average damage for each virtual network by different algorithms. We first note that the damage caused by both DVN and FVL (in Figs. 5 and 7, respectively) follows a similar pattern. This is because the number of virtual links is proportional to the number of virtual networks (two nodes in each virtual network are connected with 60% probability). Compared with DVN, the total damage caused by the FVL is larger, for two reasons: one DVN has more than one FVL, and the connected virtual networks may also have FVLs.

Different from DVN and FVL, the total damage caused by the FPL does not change much with a different number of virtual networks in Fig. 9. In each step, one FPL is repaired; hence the effect of FPL is related to the time consumption. The damage caused by the FPL of GR is the lowest, because GR consumption is the least in the algorithms. Curves of the total time for each algorithm are shown in Fig. 10, and show similarity with the damage caused by the FPL.

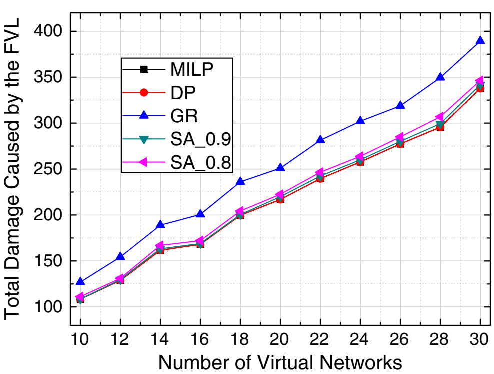

Fig. 7. Total damage caused by the FVL versus number of virtual networks.

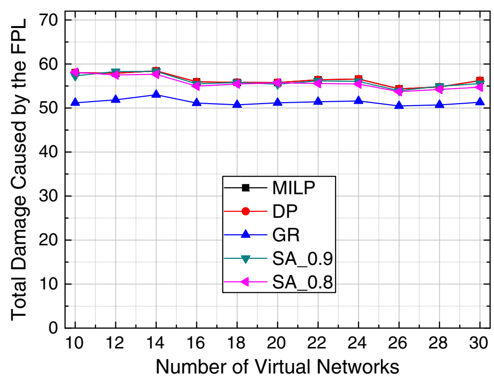

Fig. 9. Total damage caused by the FPL versus number of virtual networks.

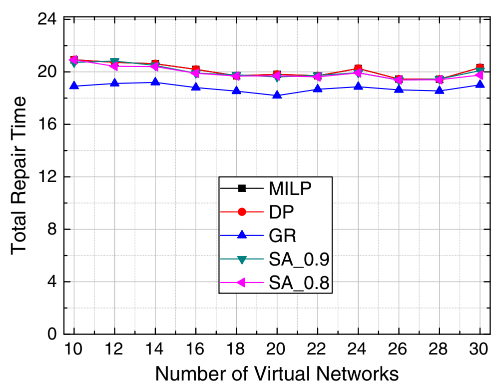

Fig. 10. Total repair time versus number of virtual networks.

B. Damage Versus Number of Failed Physical Links

For the damage (Disaster 1) versus the number of FPLs, the number of virtual networks is 30. $\{l_{EG}\}$, $\{l_{EG}, l_{EF}\}$, $\{l_{EG}, l_{EF}, l_{GJ}\}$, $\{l_{EG}, l_{EF}, l_{GJ}, l_{DE}\}$, and $\{l_{EG}, l_{EF}, l_{GJ}, l_{DE}, l_{GH}\}$ are failed when the numbers of FPLs are 1–5, respectively.

Figures 11 and 12 show the total damage caused by the DVN and FVL versus the number of FPLs, respectively. We note that the curves of DVN and FVL increase with the number of FPLs because more FPLs will cause more DVN and FVLs. We also observe that GR has the worst results, and MILP and DP have the best results, because damage caused by the DVN and FVL are the most important metrics of TRP, and the MILP and DP yield better schedules for the repairman.

Similar to the total damage caused by the DVN and FVL, the total damage caused by the FPL and repair time also increases with the number of FPLs in Figs. 13 and 14 because more FPLs need more repair time, which will lead to more damage caused by the FPL. But the results of GR are better than other algorithms, because GR is a local optimization of time in each step, while other algorithms make a good balance between the damage caused by the DVN, FVL, and FPL. Hence, GR has good performance in repair time and damage caused by the FPL.

From the above analysis, we note that the scheme with minimum repair time may not be the optimal choice for the repairman. MILP and DP can obtain the best solution for the repairman with minimum damage, GR can get the repair scheme with the minimum travel time and simplest complexity, and SA provides a balance between them.

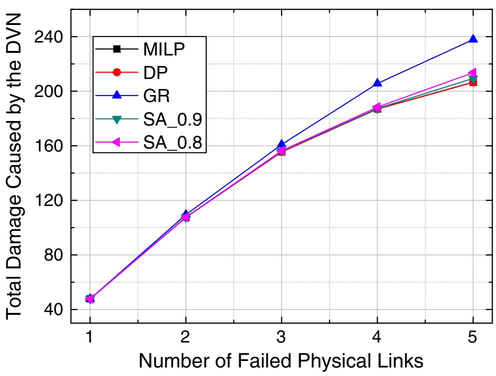

Fig. 11. Total damage caused by the DVN versus number of FPLs.

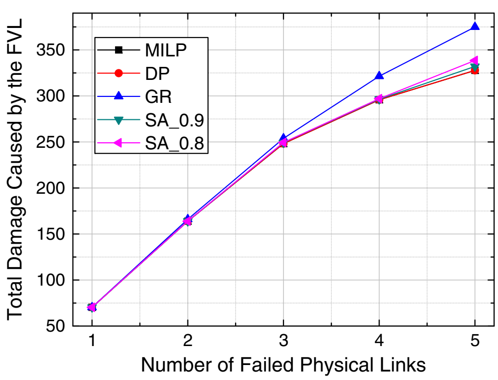

Fig. 12. Total damage caused by the FVL versus number of FPLs.

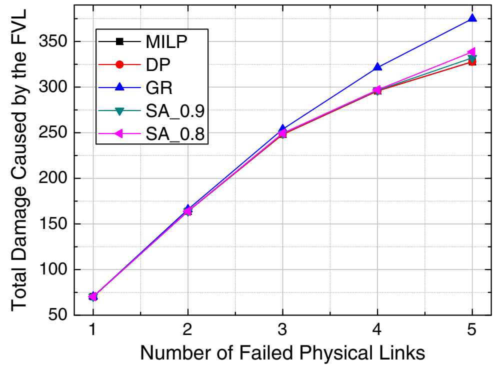

Fig. 13. Total damage caused by the FPL versus number of FPLs.

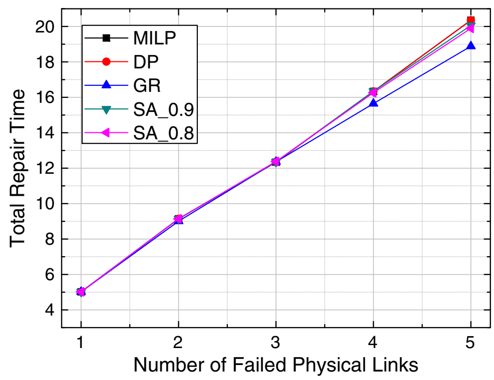

Fig. 14. Total repair time versus number of FPLs.

VII. Conclusion

In this study, we focused on how to repair FPLs in an optical network after a disaster to restore virtual networks. The TRP is proposed to model the problem, and an auxiliary graph is proposed to simplify the model. Then, the MILP model and heuristic algorithms including GR, DP, and SA are introduced to solve the problem. Simulation results show that our proposed model and algorithms can achieve a good solution for the repairman with minimum damage. The proposed heuristic solutions can be a good fit for larger-scale problems with small deviations from the optimal solution, given their comparably low complexity.

As an extension of TRP, multiple repairmen with different means of transport (such as helicopters) should be considered. Besides repairing virtual networks through physical repairs, we can also restore virtual networks using intelligent network management systems by using the excess capacity in the network.

Acknowledgment

This work is supported by a 863 program (2015AA015503), an NSFC project (61271189, 61201154), an RFDP Project (20120005120019), the Ministry of Education of the China Mobile Research Foundation (MCM20130132), the Fund of the State Key Laboratory of Information Photonics and

Optical Communications (BUPT), the China Scholarship Council (CSC), and the Networks Lab at the University of California, Davis, USA. Parts of this work were presented at the Optical Fiber Communication Conference (OFC), Los Angeles, CA, USA, March 2015.

References

[1] M. F. Habib, M. Tornatore, M. D. Leenheer, F. Dikbiyik, and B. Mukherjee, "Design of disaster-resilient optical datacenter networks," *J. Lightwave Technol.*, vol. 30, no. 16, pp. 2563–2573, Aug. 2012.

[2] M. F. Habib, M. Tornatore, F. Dikbiyik, and B. Mukherjee, "Disaster survivability in optical communication networks," *Comput. Commun.*, vol. 36, no. 6, pp. 630–644, Mar. 2013.

[3] Y. Ran, "Considerations and suggestions on improvement of communication network disaster countermeasures after the Wenchuan earthquake," *IEEE Commun. Mag.*, vol. 49, no. 1, pp. 44–47, Jan. 2011.

[4] T. Adachi, Y. Ishiyama, Y. Asakura, and K. Nakamura, "The restoration of telecom power damaged by the Great East Japan Earthquake," in *Proc. IEEE Telecommunications Energy Conf.*, Amsterdam, The Netherlands, Oct. 2011, pp. 1–5.

[5] S. Chandana and H. Leung, "A system of systems approach to disaster management," *IEEE Commun. Mag.*, vol. 48, no. 3, pp. 138–145, Mar. 2010.

[6] N. D. Tura, S. M. Reilly, S. Narasimhan, and Z. J. Yin, "Disaster recovery preparedness through continuous process optimization," *Bell Labs Tech. J.*, vol. 9, no. 2, pp. 147–162, Apr. 2004.

[7] K. T. Morrison, "Rapidly recovering from the catastrophic loss of a major telecommunications office," *IEEE Commun. Mag.*, vol. 49, no. 1, pp. 28–35, Jan. 2011.

[8] J. Wang, C. Qiao, and H. Fu, "On progressive network recovery after a major disruption," in *Proc. INFOCOM*, Shanghai, China, Apr. 2011, pp. 1925–1933.

[9] H. Yu and C. Yang, "Partial network recovery to maximize traffic demand," *IEEE Commun. Lett.*, vol. 15, no. 12, pp. 1388–1390, Dec. 2011.

[10] S. Kamaura, D. Shimazaki, Y. Uematsu, K. Genda, and K. Sasayama, "Multi-staged network restoration from massive failures considering transition risks," in *IEEE Int. Conf. on Communications (ICC)*, Sydney, Australia, Jan. 2014, pp. 1308–1313.

[11] C. Ma, J. Zhang, Y. Zhao, and M. F. Habib, "Scheme for optical network recovery schedule to restore virtual networks after a disaster," in *Optical Fiber Communications Conf. and Exhibition (OFC)*, Los Angeles, CA, Mar. 2015, paper M3I.4.

[12] M. F. Habib, M. Tornatore, and B. Mukherjee, "Fault-tolerant virtual network mapping to providing content connectivity in optical networks," in *Optical Fiber Communications Conf. and Exhibition (OFC)*, Anaheim, CA, Mar. 2013, paper OTh3E.4.

[13] Z. Ye, A. N. Patel, P. N. Ji, and C. Qiao, "Survivable virtual infrastructure mapping with dedicated protection in transport software-defined networks [Invited]," *J. Opt. Commun. Netw.*, vol. 7, no. 2, pp. A183–A189, Feb. 2015.

[14] W. Xie, J. P. Jue, Q. Zhang, Q. She, P. Palacharla, and M. Sekiya, "Survivable impairment-constrained virtual optical network mapping in flexible-grid optical networks," *J. Opt. Commun. Netw.*, vol. 6, no. 11, pp. 1008–1017, Nov. 2014.

[15] S. Huang, B. Guo, X. Li, J. Zhang, Y. Zhao, and W. Gu, "Pre-configured polyhedron based protection against multi-link failures in optical mesh networks," *Opt. Express*, vol. 22, no. 3, pp. 2386–2402, Jan. 2014.

[16] C. Ma, J. Zhang, Y. Zhao, J. Yuan, Y. Shi, S. Huang, and W. Gu, "Pre-configured ball (p-Ball) protection method with minimum backup links for dual-link failure in optical mesh networks," *IEEE Commun. Lett.*, vol. 19, no. 3, pp. 363–366, Mar. 2015.

[17] R. Asthana, Y. N. Singh, and W. D. Grover, "p-Cycles: an overview," *IEEE Commun. Surv. Tutorials*, vol. 12, no. 3, pp. 97–111, Feb. 2010.

[18] C. C. Meixner, F. Dikbiyik, M. F. Habib, M. Tornatore, C. Chuah, and B. Mukherjee, "Disaster-survivable cloud-network mapping," *Photonic Netw. Commun.*, vol. 27, no. 3, pp. 141–153, June 2014.

[19] B. Mukherjee, M. F. Habib, and F. Dikbiyik, "Network adaptability from disaster disruptions and cascading failures," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 230–238, May 2014.

[20] Y. Zhao, X. Li, H. Li, X. Wang, J. Zhang, and S. Huang, "Multi-link faults localization and restoration based on fuzzy fault set for dynamic optical networks," *Opt. Express*, vol. 21, no. 2, pp. 1496–1511, Jan. 2013.

[21] D. Turner, K. Levchenko, A. C. Snoeren, and S. Savage, "California fault lines: Understanding the causes and impact of network failures," *ACM SIGCOMM Comp. Commun. Rev.*, vol. 40, no. 4, pp. 315–326, Oct. 2010.

[22] J. Zhang and B. Mukherjee, "A review of fault management in WDM mesh networks: Basic concepts and research challenges," *IEEE Netw.*, vol. 18, no. 4, pp. 41–48, Mar. 2014.

[23] J. M. Simmons, "Catastrophic failures in a backbones network," *IEEE Commun. Lett.*, vol. 16, no. 8, pp. 1328–1331, Aug. 2012.

[24] C. H. Papadimitrou, "The Euclidean traveling salesman problem is NP-complete," *Theor. Comput. Sci.*, vol. 4, no. 3, pp. 237–244, June 1977.

[25] G. Laporte, "The traveling salesman problem: An overview of exact and approximate algorithms," *Eur. J. Oper. Res.*, vol. 59, no. 2, pp. 231–247, June 1992.

[26] R. Bellman, "Dynamic programming treatment of the traveling salesman problem," *J. ACM*, vol. 9, no. 1, pp. 61–63, Jan. 1962.

[27] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[28] P. Larranaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic, "Genetic algorithms for the traveling salesman problem: A review of representation and operators," *Artif. Intell. Rev.*, vol. 13, no. 2, pp. 129–170, Apr. 1999.

[29] V. Granville, M. Krivanek, and J. P. Rasson, "Simulated annealing: A proof of convergence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 652–656, Aug. 2002.

[30] G. Laporte, "The vehicle routing problem: An overview of exact and approximate algorithms," *Eur. J. Oper. Res.*, vol. 59, no. 3, pp. 345–358, June 1992.

[31] V. Pillac, M. Gendreau, C. Gueret, and A. L. Medaglia, "A review of dynamic vehicle routing problems," *Eur. J. Oper. Res.*, vol. 225, no. 1, pp. 1–11, Feb. 2013.

[32] M. Pavone, E. Frazzoli, and F. Bullo, "Adaptive and distributed algorithms for vehicles routing in a stochastic and dynamic environment," *IEEE Trans. Autom. Control*, vol. 56, no. 6, pp. 1259–1274, June 2011.

[33] A. Garcia, P. Jodra, and J. Tejel, "A note on the traveling repairman problem," *Networks*, vol. 40, no. 1, pp. 27–31, Aug. 2002.

[34] T. Dewilde, D. Cattrysse, S. Coene, F. C. R. Spieksma, and P. Vansteenwegen, "Heuristics for the traveling repairman problem with profits," *Comput. Oper. Res.*, vol. 40, no. 7, pp. 1700–1707, July 2013.

[35] S. Irani, X. Lu, and A. Regan, "On-line algorithms for the dynamic traveling repair problem," *J. Schedul.*, vol. 7, no. 3, pp. 243–258, May 2004.

[36] I. Petrakis, C. Hass, and M. Bichler, "On the impact of real-time information on field service scheduling," *Decis. Support Syst.*, vol. 53, no. 2, pp. 282–293, May 2012.

[37] D. J. Bertsimas and G. V. Ryzin, "A stochastic and dynamic vehicle routing problem in the Euclidean plane," *Oper. Res.*, vol. 39, no. 4, pp. 601–615, Aug. 1991.

[38] J. D. Papastavrou, "A stochastic and dynamic routing policy using branching processes with state dependent immigration," *Eur. J. Oper. Res.*, vol. 95, no. 1, pp. 167–177, Nov. 1996.

[39] G. Mosheiov, "The traveling salesman problem with pick-up and delivery," *Eur. J. Oper. Res.*, vol. 79, no. 2, pp. 299–310, Dec. 1994.

[40] P. K. Sheridan, E. Gluck, Q. Guan, T. Pickles, B. Bacioglu, and B. Benhabib, "The dynamic nearest neighbor policy for the multi-vehicle pick-up and delivery problem," *Transp. Res. Part A*, vol. 49, pp. 178–194, Mar. 2013.

Chen Ma is currently pursuing his Ph.D. degree at the State Key Laboratory of Information Photonics and Optical Communications at Beijing University of Posts and Telecommunications, Beijing, China. He received his B.S. and M.S. degrees from the School of Electrical and Electronic Engineering, North China Electric Power University, Beijing, China, in 2009 and 2012. He was also a visiting student at the University of California, Davis. His research interests include network survivability, network virtualization, and management and control of multi-layer optical transport networks.

Jie Zhang is currently a professor and the vice dean of the Information Photonics and Optical Communications Institute at Beijing University of Posts and Telecommunications (BUPT), China. He earned his bachelor's degree in communication engineering and a Ph.D. in electromagnetic field and microwave technology from BUPT. He has published more than 300 technical papers, authored 8 books, and submitted 17 ITU-T recommendation contributions and 6 IETF drafts. Also he holds 17 patents. He has served as a TPC member for a number of conferences such as ACP, OECC, PS, ONDM, COIN, and ChinaCom. His research focuses on optical network architecture, protocols, and standards.

Yongli Zhao is an associate professor at the Institute of Information Photonics and Optical Communications at Beijing University of Posts and Telecommunications (BUPT). He received the B.S. degree in communication engineering and the Ph.D. degree in electromagnetic field and microwave technology from BUPT, in 2005 and in 2010, respectively. More than 140 articles have been published. His research focuses on software-defined optical networking, elastic optical networks, packet transport networks, and so on.

M. Farhan Habib is currently working as a software engineer at Intel. He received his Ph.D. in computer science from the University of California, Davis, in 2014. His research interests include network survivability, network optimization, and compilers.

S. Sedef Savas is currently pursuing her Ph.D. degree in computer science from the University of California at Davis. She received her B.S. in computer science and engineering from Sabanci University, Istanbul, Turkey, in 2009 and her M.S. in computer science from Koc University, Istanbul, Turkey, in 2011. Her research interests include adaptability and survivability of communication networks against disasters, QoS-aware service provisioning schemes, and software-defined networking (SDN).

Biswanath Mukherjee is Distinguished Professor at the University of California, Davis, where he was Chairman of Computer Science during 1997–2000. He received the B.Tech. degree from the Indian Institute of Technology, Kharagpur, in 1980 and the Ph.D. from the University of Washington, Seattle, in 1987. He was General Co-Chair of the IEEE/OSA Optical Fiber Communications (OFC) Conference 2011, Technical Program Co-Chair of OFC 2009, and Technical Program Chair of the IEEE INFOCOM 1996 conference. He is Editor of Springer's Optical Networks Book Series. He has served on eight journal editorial boards, most notably *IEEE/ACM Transactions on Networking* and *IEEE Network*. In addition, he has guest edited special issues of the *Proceedings of the IEEE*, the IEEE/OSA *Journal of Lightwave Technology*, the *IEEE Journal on Selected Areas in Communications*, and *IEEE Communications*. He has supervised 63 Ph.D.s to completion and currently mentors 18 advisees, mainly Ph.D. students. He is the winner of the 2004 Distinguished Graduate Mentoring Award and the 2009 College of Engineering Outstanding Senior Faculty Award at UC Davis. He is the co-winner of Optical Networking Symposium Best Paper Awards at IEEE GLOBECOM 2007 and 2008. He is the author of the graduate-level textbook *Optical WDM Networks* (Springer, January 2006). He served a 5-year term on the Board of Directors of IPLocks, a Silicon Valley startup company (acquired by Fortinet). He has served on the Technical Advisory Board of several startup companies, including Teknovus (acquired by Broadcom). He is an IEEE Fellow.