2005

A New Survivable Mapping Problem in IP-over-WDM Networks

Chang Liu
Iowa State University

Lu Ruan
Iowa State University

Follow this and additional works at: http://lib.dr.iastate.edu/cs_techreports

Part of the OS and Networks Commons

Recommended Citation
http://lib.dr.iastate.edu/cs_techreports/207
A New Survivable Mapping Problem in IP-over-WDM Networks

Abstract
We introduce a new version of the widely studied survivable mapping problem in IP-over-WDM networks. The new problem allows augmenting the given logical topology and is described as follows: given a physical topology and a logical topology, compute a survivable logical topology that contains the given logical topology such that the minimal survivable mapping cost for the result logical topology is minimized. The problem is significant for two reasons: 1) If there does not exist a survivable mapping for the given logical topology, we can add logical links to the given logical topology to make it survivable; 2) Even if a survivable mapping for the given logical topology can be found, it is still possible to reduce the minimal survivable mapping cost by adding logical links selectively. We first prove the existence of a solution to the problem, then provide a straightforward Integer Linear Program (ILP) formulation for the problem. Moreover, we present a theoretical result that leads to a simple NP-hardness proof of the problem and an improved ILP formulation. Simulation results demonstrate the significance of both the new survivable mapping problem and the theoretical result.

Disciplines
OS and Networks

This article is available at Iowa State University Digital Repository: http://lib.dr.iastate.edu/cs_techreports/207
A New Survivable Mapping Problem in IP-over-WDM Networks

Chang Liu and Lu Ruan
Department of Computer Science
Iowa State University, Ames, IA 50011
e-mail: {liuc, ruan}@cs.iastate.edu

Abstract—We introduce a new version of the widely studied survivable mapping problem in IP-over-WDM networks. The new problem allows augmenting the given logical topology and is described as follows: given a physical topology and a logical topology, compute a survivable logical topology that contains the given logical topology such that the minimal survivable mapping cost for the result logical topology is minimized. The problem is significant for two reasons: 1) If there does not exist a survivable mapping for the given logical topology, we can add logical links to the given logical topology to make it survivable; 2) Even if a survivable mapping for the given logical topology can be found, it is still possible to reduce the minimal survivable mapping cost by adding logical links selectively.

We first prove the existence of a solution to the problem, then provide a straightforward Integer Linear Program (ILP) formulation for the problem. Moreover, we present a theoretical result that leads to a simple NP-hardness proof of the problem and an improved ILP formulation. Simulation results demonstrate the significance of both the new survivable mapping problem and the theoretical result.

Index Terms—Network survivability, survivable mapping, IP-over-WDM

I. INTRODUCTION

There is a growing consensus that the next generation Internet will employ an IP-over-WDM architecture [1]. In this architecture, IP routers are attached to a WDM optical network, which consists of optical cross-connects (OXCs) interconnected by multi-wavelength optical fibers. The IP routers are interconnected by lightpaths, which are circuit-switched optical connections provisioned by the optical network. Each lightpath spans one or more fibers and occupies one wavelength channel in every fiber along its route. The IP routers and the lightpaths interconnecting them form a logical topology. The OXCs and the optical fibers interconnecting them form a physical topology. Routers and OXCs are called logical nodes and physical nodes respectively. Lightpaths and optical fibers are called logical links and physical links respectively. Given a logical topology and a physical topology, the mapping problem is to find a path for every logical link in the physical topology. There are different ways to map a logical topology onto a physical topology. For example, consider the logical topology and the physical topology shown in Fig. 1. One mapping could route the logical link $a - c$ over the physical path $A - B - C$, while another mapping could route $a - c$ over the physical path $A - E - C$.

Survivability is an important issue in IP-over-WDM networks since a network failure such as fiber cut can cause tremendous data loss. Many lightpath protection and restoration schemes have been proposed to achieve survivability in WDM optical networks [2][3][4][5][6][7]. And many MPLS-based protection and restoration schemes have been proposed to achieve survivability in IP networks [8][9][10][11][12]. IP layer failure recovery is possible only if a network failure does not disconnect the IP topology. However, a fiber cut, which is the predominant form of failures in optical networks [13], may cause the IP topology to be disconnected because all the lightpaths using the failed fiber will be disrupted. Therefore, it’s desirable to solve the following survivable mapping problem: given a logical topology and a physical topology, map the logical topology onto the physical topology such that the logical topology remains connected in case of any single physical link failure. In general, there are different ways to map a logical topology onto a physical topology and not all of them are survivable. For example, consider the logical topology and the physical topology given in Fig. 1. One possible mapping is to map $a - b$ to $A - B$, $a - c$ to $A - B - C$, $b - d$ to $B - D$, $b - e$ to $B - A - E$, $c - e$ to $C - E$, and $d - e$ to $D - E$. This is not a survivable mapping since the failure

Fig. 1. Example logical topology and physical topology.
of physical link $A-B$ will cause logical links $a-b$, $a-c$, and $b-e$ to fail, leaving the logical topology disconnected. A survivable mapping can be obtained by mapping $a-b$ to $A-E-D-B$ instead.

The survivable mapping problem has been studied in [14][15][16][17][18]. In [14], it is proven that determining whether a survivable mapping is possible for a logical topology on a given physical topology is NP-Complete. [15] gives a necessary and sufficient condition for a mapping to be survivable and an Integer Linear Program (ILP) formulation to solve the problem with the objective of minimizing the cost of the mapping. Necessary conditions for ring logical topologies being survivably routed are also presented in [15]. Various heuristic algorithms for the survivable mapping problem are proposed in [16][17][18].

In this paper, we introduce a new version of the survivable mapping problem, which allows adding logical links to the given logical topology in order to find a minimal cost survivable routing. This is different from the original survivable mapping problem that does not allow the logical topology to be changed. The significance of the new problem is two-fold:

1) If there does not exist a survivable mapping for the given logical topology, adding some logical links to the given logical topology will enable a survivable mapping to be obtained.

2) Even if a survivable mapping for the given logical topology can be found, adding some logical links to the given logical topology may reduce the minimal survivable mapping cost. (The cost of a mapping is the total cost of all logical links, where the cost of a logical link is the number of hops in its physical path.) This effect can be illustrated using the example physical and logical topologies in Figure 1. As shown in Figure 2, the minimal survivable mapping cost for the logical topology in Figure 1(a) is 10. After adding a logical link $a-e$, the minimal survivable mapping cost for the new logical topology becomes 9.

![A minimal cost survivable mapping](image1)

![A minimal cost survivable mapping after adding logical link a-e](image2)

The rest of the paper is organized as follows. In section II, we define the necessary terms and notations and give the formal problem definition. In section III, we present the existence proof of a solution to the new survivable mapping problem and a brute force ILP solving the problem. In section IV, we first prove a theorem, then we give an NP-hardness proof of the problem and an improved ILP solution to the problem with the help of this theorem. Simulation results are discussed in section V. A conclusion is given in section VI.

II. TERMINOLOGY AND PROBLEM DEFINITION

A. Terminology

A logical topology and a physical topology are represented by undirected graphs $G_l = (V, E_l)$ and $G_p = (V, E_p)$ respectively, where $V$ denotes the set of logical and physical nodes, $E_l$ and $E_p$ denote the set of logical links and physical links respectively. Although both logical topology and physical topology are represented by undirected graphs, sometimes it is useful to treat a link $ij$ in $E_p$ as two directional links, $ij$ and $ji$. For convenience, the bi-directional edge set derived from $E_p$ by attaching both directions to each undirected edge will be denoted as $E^b_p$, where “bd” stands for “bi-directional”.

For a graph $G = (V,E)$, $\forall S \subseteq V$ and $S \neq \emptyset$, an edge cut of $G$ defined by $S$, denoted by $EC_G(S,V-S)$, is the set of edges in $G$ with one endpoint in $S$ and the other endpoint in $V - S$. Clearly, the removal of $EC_G(S,V-S)$ will disconnect $G$.

For $s,t \in V$ and $s \neq t$, a path from $s$ to $t$ in the physical topology $G_p$ is denoted as $P_{st} = \{s = \cdots = t\}$. $P_{st}$ denotes the set of all paths from $s$ to $t$ in $G_p$. A mapping from $G_l$ to $G_p$ is a function $M : E_l \rightarrow \bigcup_{s,t \in V} P_{st}$. That is, $M$ maps each logical link $st \in E_l$ onto a path from $s$ to $t$ in the physical topology.

The load set of a physical link $ij \in E_p$ under a mapping $M$, denoted as $l_M(ij)$, is the set of all logical links whose physical paths traverse $ij$, i.e., $l_M(ij) = \{st \in E_l | ij \text{ is in } M(st)\}$. The remaining logical topology in case of the failure of $ij \in E_p$ under $M$ is defined as $G^M_l(ij) = (V,E_l - l_M(ij))$. $ij \in E_p$ is a critical link under $M$ if $G^M_l(ij)$ is not a connected graph.

$M$ is a survivable mapping from $G_l$ to $G_p$ if under $M$, the failure of any physical link will not disconnect the logical topology, i.e., $\forall ij \in E_p$, $G^M_l(ij)$ is a connected graph. In other words, $M$ is a survivable mapping if there is no critical link in the physical topology under $M$. $G_l$ is a survivable logical topology on a physical topology $G_p$ if there exists a survivable mapping $M$ from $G_l$ to $G_p$.

$st \in E_l$ is a reflective logical link if there is a physical link between $s$ and $t$ in the physical topology, i.e., $st \in E_l \cap E_p$. $st \in E_l \cap E_p$ is a reflectively-routed logical link under a mapping $M$ if $M(st) = (s-t)$. $M$ is a reflectively-routed mapping if all reflective logical links are reflectively-routed, i.e., $\forall st \in E_l \cap E_p$, $M(st) = (s-t)$.

The cost of a mapping $M$, denoted by $cost(M)$, is the total capacity (wavelengths) used in the physical topology to route all the logical links. It can be computed as $cost(M) = \sum_{st \in E_l} |M(st)|$, where $|M(st)|$ is the hop count of $M(st)$.

The cost of a survivable logical topology $G_l$, denoted by $cost(G_l)$, is the minimal cost of all survivable mappings from $G_l$ to $G_p$, i.e., $cost(G_l) = \min M$ is a survivable mapping from $G_l$ to $G_p$. The survivable mapping $M$ that achieves the minimal cost is called a minimal cost survivable mapping from $G_l$ to $G_p$.

A graph is 2-edge-connected if the minimum number of edges whose removal disconnect the graph is 2. Given a logical topology $G_l = (V,E_l)$ and a 2-edge-connected physical topology $G_p = (V,E_p)$, a minimal cost survivable logical topology that contains $G_l$ on $G_p$.
is a survivable logical topology $G'_l = (V, E'_l)$ such that $E_l \subseteq E'_l$ and $\text{cost}(G'_l)$ is minimized. We denote this minimized cost as $\text{MIN-COST}G_l$, then $\text{MIN-COST}G_l = \min G$ is survivable and $E_l \subseteq E(G) \text{cost}(G)$.

B. Problem Definition

The new survivable mapping problem we study in this paper is the following: given a logical topology $G_l = (V, E_l)$ and a 2-edge-connected physical topology $G_p = (V, E_p)$, compute a survivable logical topology $G'_l = (V, E'_l)$ and a survivable mapping $M$ from $G'_l$ to $G_p$ such that $E_l \subseteq E'_l$ and $\text{cost}(M) = \text{MIN-COST}G_l$. In other words, the goal is to find a minimal cost survivable logical topology that contains $G_l$ and a mapping $M$ that achieves the minimal cost.

Note that in practice, WDM network topologies are required to be 2-edge-connected so that traffic restoration is possible when a fiber cut occurs in the network.

III. A Brute Force ILP FORMULATION

First, we prove that a solution to the new survivable mapping problem always exists.

**Theorem 1:** Given a logical topology $G_l = (V, E_l)$ and a 2-edge-connected physical topology $G_p = (V, E_p)$, there exists a survivable logical topology that contains $G_l$ on $G_p$.

**Proof.** Let $G'_l = (V, E_l \cup E_p)$. Clearly, $G'_l$ contains $G_l$. We will prove that $G'_l$ is a survivable logical topology by showing that any reflectively-routed mapping from $G'_l$ to $G_p$ is survivable.

Let $M$ be a reflectively-routed mapping from $G'_l$ to $G_p$. Under $M$, when any link $ij \in E_p$ fails, among the logical links in $E(G'_l)$, at least those links in $E_p - \{ij\}$ will stay in the remaining logical topology because they are reflectively-routed and therefore not affected by the failure of $ij$. That is, the remaining logical topology contains $E_p - \{ij\}$. And since $G_p$ is 2-edge-connected, the remaining logical topology must be connected. Thus, $M$ is a survivable mapping.

Let $K_n$ denote the undirected complete graph on the vertex set $V$, where $n = |V|$. A straightforward Integer Linear Program (ILP) formulation for the new survivable mapping problem is given below.

Variables to be solved:
- $f_{ij}^{st}$: takes value 1 if logical link $st$ is routed on physical link $ij$, 0 otherwise.
- $x^{st}$: takes value 1 if $st$ is included in the result logical topology, 0 otherwise.

Objective function:

\[
\text{Minimize } \sum_{ij \in E_p, st \in E(K_n)} f_{ij}^{st}
\]

Subject to:

(a). Flow conservation constraints:

\[
\sum_{j \text{ s.t. } ij \in E_p^{bd}} f_{ij}^{st} - \sum_{j \text{ s.t. } ji \in E_p^{bd}} f_{ji}^{st} = \begin{cases} 
    x^{st} & \text{if } s = i \\
    -x^{st} & \text{if } t = i \\
    0 & \text{otherwise}
\end{cases}
\]

(b). Survivability constraints:

\[
\sum_{(e \in S \land e \notin V-S)} f_{ij}^{st} + f_{ji}^{st} < \sum_{(e \in V \land e \notin S)} x^{st}, \forall ij \in E_p, \forall S \subset V.
\]

(c). Deletion of existing logical links is not allowed:

\[
x^{st} = 1, \forall st \in E_l.
\]

(d). Integer constraints:

\[
f_{ij}^{st} \in \{0, 1\}, \forall ij \in E_p^{bd}, \forall st \in E(K_n).
\]

\[
x^{st} \in \{0, 1\}, \forall st \in E(K_n).
\]

In the flow conservation constraints in (a), the number of flow units is determined by $x^{st}$, which ensures that a logical link is routed only when it is included in the result logical topology, i.e., $x^{st} = 1$. In the survivability constraints in (b), the right hand side is the number of edges in the edge cut $EC_G_i(S, V-S)$ and the left hand side is the number of logical links in $EC_G_i(S, V-S)$ that are routed on $ij \in E_p$ in either direction, which equals $|M(ij) \cap EC_G_i(S, V-S)|$, where $G_l$ is the result logical topology and $M$ is the result mapping from $G_l$ to $G_p$. It is proved in [15] that $M$ is survivable if and only if $|M(ij) \cap EC_G_i(S, V-S)| < |EC_G_i(S, V-S)|$, $\forall ij \in E_p$, $\forall S \subset V$ ($S \neq \emptyset$). Therefore, the constraints in (b) ensure that the result mapping is survivable. Constraints in (c) guarantee that logical links in the given logical topology must stay in the result logical topology.

For convenience, this ILP formulation will be referred to as ILP1.

IV. A THEOREM AND ITS APPLICATIONS

ILP1 provides a straightforward method for solving the new survivable mapping problem, which considers all links not in the given logical topology as candidate links to be added to the given logical topology. In this section, we present a theorem about the problem, which shows that we can find a solution to the problem by adding only reflective logical links to the given logical topology, and the result logical topology has a reflectively-routed mapping that achieves the minimal cost. The theorem can be used to prove that the new survivable mapping problem is NP-hard. It can also be used to improve ILP1 in two ways. First, the candidate logical links to be added to the given logical topology is confined to reflective logical links that are not in the given logical topology instead of all non-existing links in the given logical topology. Second, the existence of the minimal cost reflectively-routed survivable mapping for the result logical topology makes the mapping job easier since the physical paths for those reflective logical links can be determined right away because they are reflectively-routed.
A. The Theorem

The theorem is stated as follows: given a logical topology $G_t$ and a 2-edge-connected physical topology $G_p$, it is always possible to get a minimal cost logical topology that contains $G_t$ on $G_p$ by only adding reflective logical links, and there exists a survivable reflectively-routed mapping that achieves the minimal cost for the result logical topology.

The following lemma will be used to prove the theorem.

Lemma 1: Given a physical topology $G_p = (V, E_p)$, for any survivable logical topology $G_t = (V, E_t)$ on $G_p$, there exists a set $E' \subseteq E_p - E_t$ such that $G'_t = (V, E_t \cup E')$ has a survivable reflectively-routed mapping $M'$ and $\text{cost}(M') \leq \text{cost}(G_t)$.

Proof. Pick a minimal cost survivable mapping $M$ from $G_t$ to $G_p$. If $M$ is reflectively-routed, just let $E' = \emptyset$, then $G'_t = G_t$ has a survivable reflectively-routed mapping $M' = M$ and $\text{cost}(M') = \text{cost}(G_t)$.

If $M$ is not reflectively-routed, we will call the procedure REFLECTIVATE($G_t, G_p, M$), after which $M$ and $G'_t$ will be transformed so that $G'_t$ is obtained by adding links in $E_p - E(G'_t)^{old}$ to $G'_t$ and $M$ is a survivable reflectively-routed mapping from $G_t$ to $G_p$, with $\text{cost}(M) \leq \text{cost}(G'_t)^{old}$. ($M'^{old}$ and $G'^{old}$ are used to denote the old mapping and logical topology before the transformation.) The pseudocode of the procedure REFLECTIVATE is given below. The correctness proof of the procedure follows the pseudocode.

REFLECTIVATE($G_t, G_p, M$)

$G_t$, $M$: in
$G_p$: in
1. for each non-reflectively-routed $st \in E_t \cap E_p$ do
2. let $M(st) = (s - t)$;
3. DE-CRITICALIZE($G_t, G_p, M, st$);

DE-CRITICALIZE($G_t, G_p, M, st$)

$G_t$, $M$: inout
$G_p$, $st$: in
1. Pick $s' \in V(C_1)$, $t' \in V(C_2)^*$ such that $s't' \in E_p - E_t$;
2. if such $s't'$ exists then
3. let $E_t = E_t \cup \{s't'\}$; $M(st') = (s - t')$;
4. else
5. Pick $x \in V(C_1)$, $y \in V(C_2)$ such that $xy \in E_t \cap E_p$ and $xy \neq st$;
6. let $M(xy) = (x - y)$;
7. DE-CRITICALIZE($G_t, G_p, M, xy$);

*: As stated in Claim 1 (given later), upon entering procedure DE-CRITICALIZE, $M$ is a non-survivable mapping with $st \in E_p$ being the only critical link, whose failure will disconnect $G_t$ into two connected components. For convenience, we call these two connected components $C_1$ and $C_2$.

During the execution of REFLECTIVATE, we have following observations.

Observation 1: All links in $G_t^{old}$ are kept in $G_t$. And in the result logical topology, newly added links are all from $E_p - E(G_t^{old})$.

By going through procedure REFLECTIVATE and procedure DE-CRITICALIZE, we can see that there is no logical link deletion anywhere. And whenever a new logical link is added, it is added from $E_p$.

Observation 2: The cost of $M$ never increases.

Every time before DE-CRITICALIZE is called (either line 2 of REFLECTIVATE or line 6 of DE-CRITICALIZE), a non-reflectively routed logical link is rerouted from a multi-hop physical path to a single-hop physical path, which decreases the cost of $M$ by at least 1. On the other hand, within the procedure DE-CRITICALIZE, at most one logical link is added and the newly added logical link is a reflectively logical link rerouted on a single-hop physical path (line 3 of DE-CRITICALIZE), which increases the cost of $M$ by 1. The overall effect is that the cost of $M$ does not increase.

In the procedure REFLECTIVATE, if $M$ is a survivable mapping from $G_t$ to $G_p$ at the beginning of an iteration of the for loop, then

Claim 1: after line 2 is executed and before DE-CRITICALIZE is called, $M$ becomes a non-survivable mapping from $G_t$ to $G_p$ with $st \in E_p$ being the only critical link, whose failure will disconnect $G_t$ into 2 connected components in which $s$ and $t$ will be separated.

Claim 2: after DE-CRITICALIZE returns (we will prove that DE-CRITICALIZE always returns), $M$ is a survivable mapping from $G_t$ to $G_p$. And no new non-reflectively-routed link is introduced.

If Claim 1 and Claim 2 are true, then each iteration of the for loop in REFLECTIVATE eliminates at least one non-reflectively-routed link and end up with a survivable mapping $M$ from $G_t$ to $G_p$ with no new non-reflectively-routed link being introduced. Since we have a finite number of non-reflectively-routed logical links, the procedure REFLECTIVATE will always terminate with a survivable reflectively-routed mapping $M$ from $G_t$ to $G_p$. Together with Observation 1 and Observation 2, we have a result logical topology $G_t$ that has links only added from $E_p - E(G_t)^{old}$, and $G_t$ has a survivable reflectively-routed mapping $M$ with cost no greater than $\text{cost}(G_t^{old})$. So if we can prove Claim 1 and Claim 2, the proof of Lemma 1 is done.

Proof of Claim 1: Assume $M$ is still survivable after we let $M(st) = (s - t)$. $M$ has a smaller cost now because $st$ is rerouted from a multi-hop path to a single-hop path. This contradicts the condition that $M$ is a minimal survivable mapping from $G_t$ to $G_p$ before the reroute. So $M$ must be non-survivable after the reroute. Moreover, $st \in E_p$ will become the only critical link in $M$ because $st \in E_p$ is the only physical link whose load set expands because of the reroute ($st \in E_t$ is newly included in the load set of $st \in E_p$ due to the reroute). Therefore, $st \in E_t$ must be a bridge in the remaining logical topology in case of the failure of $st \in E_p$ under the mapping before the reroute. Since removing a bridge in a connected graph will disconnect the graph into two connected components, under the new mapping, the failure of $st \in E_p$ will disconnect the logical topology into 2 connected components, and $s$ and $t$ will be separated in different connected components.

(End of Proof of Claim 1)

Proof of Claim 2: The correctness of Claim 2 lies in the following facts about the procedure DE-CRITICALIZE. Each
statement of these facts will be followed by a proof.

Fact 1: Line 5 always succeeds in finding such $xy$. And $xy \in E_l \cap E_p$ is a non-reflectively-routed logical link.

Since we enter the “else” branch, we must have

$$\forall s \in V(C_1), t \in V(C_2), s't \in E_p \Rightarrow s't \in E_t \quad (*)$$

On the other hand, because of the 2-edge-connectivity of the physical topology, the edge cut $EC_{G_p}(V(C_1), V(C_2))$ must contain at least one more physical link $xy \neq st$ besides $st$. By $(*)$, $xy$ must also be in $E_l$, i.e., $xy \in E_l \cap E_p$, and $xy \neq st$. Since $xy \in EC_{G_p}(V(C_1), V(C_2))$, which is a subset of the load set of $st$ because $st$ is currently a critical link, it means that $xy \in E_l$ is routed on $st \in E_p$ instead of $xy \in E_p$. So $xy \in E_l \cap E_p$ is a non-reflectively-routed logical link.

Fact 2: After line 6 is executed, $st$ is not a critical link. Moreover, $xy$ becomes the only critical link whose failure will disconnect the logical topology into 2 connected components, in which $x$ and $y$ are separated.

After line 6 is executed, $xy \in E_l$ is no longer in the load set of $st \in E_p$. And because $x \in V(C_1)$ and $y \in V(C_2)$, the failure of $st \in E_p$ will not disconnect the logical topology now. So $st$ is not a critical link. Moreover, $xy \in E_p$ has to be critical now. If not, on the one hand, the cost of $M$ decreases because $xy \in E_l$ has been rerouted from a multi-hop path to a single-hop path; on the other hand, there is no critical link under $M$. This means that we get a survivable mapping with lower cost, which contradicts the condition that $M^{old}$ is a minimal cost survivable mapping. Also, $xy \in E_p$ is the only critical link in $M$ because $xy \in E_p$ is the only physical link whose load set expands because of the reroute ($xy \in E_l$ is newly included in the load set of $xy \in E_p$ due to the reroute). Therefore, $xy \in E_l$ must be a bridge in the remaining logical topology in case of the failure of $xy \in E_p$ under the mapping before the reroute. Since removing a bridge in a connected graph will disconnect the graph into two connected components, under the new mapping, the failure of $xy \in E_p$ will disconnect the logical topology into 2 connected components, and $x$ and $y$ will be separated in different connected components.

Fact 3: After line 3 is executed, $s't \neq st$ and neither $st \in E_p$ nor $s't \in E_t$ is a critical link.

Before line 3 is executed, $st \in E_l$ and $s't \notin E_l$, so it must be the case that $s't \neq st$. After line 3 is executed, the newly added logical link $s't$ is not routed on $st$, so the failure of $st \in E_p$ will not affect $s't \in E_l$. And since $s' \in V(C_1)$ and $t' \in V(C_2)$, in the remaining logical topology in case of the failure of $st \in E_p$, $s't \in E_l$ will bridge $C_1$ and $C_2$ together as a connected graph, which means that $st \in E_p$ is not a critical link now. As of $s't \in E_p$, there is one more logical link, $s't \in E_l$, in the new load set of $s't \in E_p$. Assume that $s't \in E_p$ is critical now, it must have been critical too before $s't \in E_p$ is added to the logical topology, which contradicts the condition that $st(\neq s't') \in E_p$ is the only critical link at that point of time (this condition is enforced by Claim 1 if DE-CRITICALIZE is called from line 3 of REFLECTIVATE, and by Fact 2 if DE-CRITICALIZE is called from line 7 of itself). Thus, $s't \in E_p$ is not a critical link now.

Fact 4: After $st$ becomes non-critical in DE-CRITICALIZE, it will never become critical again all the way till the end of REFLECTIVATE. Also, for each newly added $s't \in E_l$, corresponding $s't \in E_p$ will never become critical either.

$s \in E_l$ is now reflectively-routed. The load set of $st \in E_p$ will never include other logical links till the end of REFLECTIVATE because

1. All newly added logical links will be routed on the corresponding single-hop paths.
2. We only reroute non-reflectively-routed logical links onto corresponding single-hop paths.

So $st$ will never become critical.

Because of the same reasons, $s't \in E_p$ will never become critical either.

After DE-CRITICALIZE starts, if it enters the “then” branch (i.e., $s't$ can be found), Fact 3 tells us that $st \in E_p$ will become a non-critical link, and for the newly added logical link $s't \in E_l$ ($s't \neq st$), the corresponding $s't \in E_p$ is not critical either. If it enters the “else” branch, Fact 1 and Fact 2 tell us that $st$ will become non-critical and another physical link $xy$ will become critical. Either way, DE-CRITICALIZE eliminates one critical link and may introduce another critical link. Fact 4 guarantees that de-criticalized links as well as $s't \in E_p$ for those newly added reflective logical links $s't \in E_l$ will never become critical in the future. Since we have a finite number of physical links (i.e., potential critical links), DE-CRITICALIZE will always return with a situation where no critical link exists, which means that the result mapping is a survivable mapping from the result logical topology to the physical topology. In addition, it’s easy to verify that no new non-reflectively-routed link is introduced in DE-CRITICALIZE.

(End of Proof of Claim 2)

We now give the formal statement of the theorem and its proof.

Theorem 2: Given a logical topology $G_l = (V, E_l)$ and a 2-edge-connected physical topology $G_p = (V, E_p)$, there exists an edge set $E'' \subseteq E_p - E_l$ such that $G''_l = (V, E_l \cup E'')$ is a minimal cost survivable logical topology that contains $G_l$ on $G_p$. Moreover, there is a reflectively-routed mapping $M''$ from $G''_l$ to $G_p$ such that $M''$ achieves the minimal cost, i.e., $cost(M'') = MIN-COST_{G_l}$.

Proof. Arbitrarily pick one minimal cost survivable logical topology $G = (V, E)$ that contains $G_l$ on $G_p$. Let $M$ be a minimal cost survivable mapping from $G$ to $G_p$, i.e., $cost(M) = cost(G) = MIN-COST_{G_l}$.

Case 1: Logical links added in $E$ (if any) are all from $E_p - E_l$, i.e., $E - E_l \subseteq E_p - E_l$.

1. If $M$ is a reflectively-routed mapping, then $E'' = E - E_l$, $G''_l = G$, and $M'' = M$ are the edge set, the logical topology, and the mapping we are looking for.
2. If $M$ is not a reflectively-routed mapping from $G$ to $G_p$, then by Lemma 1, there exists $G' = (V, E \cup E')$ ($E' \subseteq E_p - E$) that has a survivable reflectively-routed mapping $M'$ from $G'$ to $G_p$ such that $cost(M') = cost(M)$ (it is impossible to get $cost(M') < cost(M)$ since $cost(M) = MIN-COST_{G_l}$). Then $E''' = (E \cup E') - E_l$, $G'''' = G'$, and $M''' = M'$ are the edge
set, the logical topology, and the mapping we are looking for. Note that \(E' \subseteq E_p - E_t\) because \(E'' = (E \cup E') - E_t = (E - E_t) \cup E'\), where \(E - E_t \subseteq E_p - E_t\) and \(E' \subseteq E_p - E \subseteq E_p - E_t\).

CASE II: There are non-reflective logical links added in \(E\), i.e., \(\exists s \in E - E_t\) such that \(st \notin E_p\).

In this case, we will call the procedure PURIFY\((G_t, G_p, G, M)\), after which \(G = (V, E)\) will be a minimal cost survivable logical topology that contains \(G_t\) on \(G_p\), where \(E - E_t \subseteq E_p - E_t\). And \(M\) is a reflectively-routed mapping from \(G\) to \(G_p\) that achieves the minimal cost MIN-COST\(_{G_t}\). Below is the pseudocode of PURIFY followed by its correctness proof.

\[
\text{PURIFY}(G_t, G_p, G, M) \\
G_t, G_p; \text{ in} \\
G, M; \text{ out} \\
1. \text{ if } M \text{ is not reflectively-routed then} \\
2. \text{ Find } G' = (V, E \cup E') \text{ and } M' \text{ such that } E' \subseteq E_p - E \text{ and } M' \text{ is a survivable reflectively-routed mapping from } G' \text{ to } G_p \text{ and } \text{cost}(M') = \text{cost}(M); \\
3. \text{ let } G = G'; M = M'; \\
4. \text{ for each } st \in (E - E_t) - E - P \text{ do} \\
\quad \text{//}(E - E_t) - E - P \text{ is the set of} \\
\quad \text{// all added non-reflective links in } G. \\
5. \text{ let } E = E - \{st\}; \\
6. \text{ for each } i j \in M(st) \text{ do} \\
\quad \text{ let } E = E \cup \{ij\}; \\
7. \text{ let } M(ij) = (i - j); \\
\]

During the execution of PURIFY, we have the following observations.

**Observation 3:** All logical links in \(G_t\) are kept in \(G\), and in the result logical topology, newly added links are all from \(E_p - E_t\).

It can be seen that the procedure that removal of logical links only occurs in line 5, where \(st \in (E - E_t) - E - P\) are removed. Thus, all logical links in \(G_t\) are kept in \(G\). Addition of logical links occurs in line 2 and line 7, and it’s easy to verify that the added logical links are all from \(E_p - E_t\).

**Observation 4:** The cost of \(M\) never increases.

Before entering the for loop in line 4, it is required that \(M\) is a survivable reflectively-routed mapping from \(G\) to \(G_p\). If this is not satisfied, by Lemma 1, there exists \(G' = (V, E')\) where \(E' \subseteq E_p - E\) that has a survivable reflectively-routed \(M'\) from \(G'\) to \(G_p\) such that \(\text{cost}(M') = \text{cost}(M)\) (it is impossible to get \(\text{cost}(M') < \text{cost}(M)\) since \(\text{cost}(M) = \text{MIN-COST}_{G_t}\)). So line 2 always succeeds and cost of \(M\) does not increase in line 2.

In each iteration of the for loop in line 4, on the one hand, \(st \in (E - E_t) - E - P\) is removed from \(G\), which decreases the cost of \(M\) by \(|M(st)|\); on the other hand, at most \(|M(st)|\) reflectively-routed logical links are added, which increases the cost of \(M\) by at most \(|M(st)|\). Thus, the cost of \(M\) does not increase in the for loop.

Overall, the cost of \(M\) never increases in PURIFY.

Within one iteration of the for loop in line 4, we use \(M_{before}\) and \(G_{before}\) to denote the mapping/logical topology before removing \(st\) and \(after\) adding \(ij\)’s and mapping them reflectively. At the end of an iteration of the for loop in line 4, we have:

**Claim 3:** \(\forall ij \in M_{before}(st), ij\) is not a critical link under \(M_{after}\).

**Claim 4:** \(\forall ij \in E_p - M_{before}(st), ij\) is not a critical link under \(M_{after}\).

If Claim 3 and Claim 4 are true, each iteration of the for loop in line 4 will eliminate exactly one added non-reflective logical link \(st \in (E - E_t) - E - P\) without breaking the survivability or introducing non-reflectively-routed logical links. Since we have a finite number of added non-reflective logical links \((E - E_t) - E - P\) is a finite set), the procedure PURIFY always terminates. And by Observation 3 and Observation 4, the result logical topology and mapping are what we are looking for. So if we can prove Claim 3 and Claim 4, the proof of Theorem 2 is done.

**Proof of Claim 3:** \(\forall ij \in M_{before}(st), l_{M_{after}}(ij) = (l_{M_{before}}(ij) - \{st\}) \cup \{ij\}\). Assume that \(ij \in E_p\) is critical under \(M_{after}\), then the failure of \(ij \in E_p\) will disconnect the logical topology \(G_{after}\) into 2 connected components \(C_1\) and \(C_2\), which separate \(i\) and \(j\). The reason is that \(ij \in E_p\) is not critical under \(M_{before}\) and the only new logical link appears in \(l_{M_{after}}\). \(ij \in E(G_{after})\).

On the other hand, all logical links in \(E(G_{after})\) (except \(ij \in E(G_{after})\)) along the logical path corresponding to \(M_{before}(st)\) are reflectively-routed onto corresponding single-hop physical paths, which implies that these logical links will not be affected by the failure of \(ij \in E_p\) under \(M_{after}\). In other words, only \(ij \in E(G_{after})\) is broken on the logical path corresponding to \(M_{before}(st)\). Without loss of generality, suppose \(s, i \in V(C_1)\) and \(t, j \in V(C_2)\). As can be seen from Figure 3, \(ij \in E_p\) must have been critical under \(M_{before}\) because removing \(l_{M_{before}}(ij) = (l_{M_{before}}(ij) - \{ij\}) \cup \{st\}\) from \(G_{before}\) would have disconnected \(G_{before}\). This contradicts the condition that \(M_{before}\) is survivable. Thus, \(ij \in E_p\) is not critical under \(M_{after}\).

(End of Proof of Claim 3)

**Proof of Claim 4:** \(\forall ij \in E_p - M_{before}(st), assume \ ij \ becomes critical under \ M_{after}\). Since the load set of \(ij\) under \(M_{after}\) is the same as under \(M_{before}\), the only possible reason to make it critical under \(M_{after}\) is the loss of \(st \in E(G_{before})\) in \(G_{after}\). So, the failure of \(ij\) would disconnect the logical topology \(G_{after}\) into 2 connected components, which separates \(s\) and \(t\). However, this is impossible because there is a path in \(G_{after}\) from \(s\) to \(t\) when \(ij\) fails since all logical links along the logical path corresponding to \(M_{before}(st)\) are reflectively-routed and not affected by the failure of \(ij\). Thus, \(\forall ij \in E_p - M_{before}(st), ij\) is not a critical link under \(M_{after}\).

(End of Proof of Claim 4)

**B. NP-hardness of the New Survivable Mapping Problem**

With the help of Theorem 2, we can prove that the new survivable mapping problem is NP-hard by a reduction from the Minimum 2-Edge-Connected Spanning Subgraph problem that has been proven to be NP-hard [19].

First, we formulate the corresponding decision problems as follows.
Two groups of logical topologies are used: a subgraph of links in $G[st, G_{st}](V(C_3), V(C_2))$. The dashed line is $st \in E(G_{st}^{\text{before}})$. The dotted lines denote the logical path from $s$ to $i$ and the logical path from $j$ to $t$ in the logical topology $G[st]^{\text{before}}$. This diagram shows that if the removal of $l_{\text{before}}(ij)$ disconnected the logical topology $G[st]^{\text{before}}$, then the removal of $l_{\text{before}}(ij) = (l_{\text{before}}(ij) - \{ij\}) \cup \{st\}$ would have also disconnected the logical topology $G_{st}^{\text{before}}$.

\[ M_{2\text{ECSS}} = \{ G, k \} | G \text{ has a 2-edge-connected spanning subgraph that contains } \leq k \text{ edges.} \]

\[ \text{NSM} = \{ G_t, G_p, c \} | \text{There is a survivable logical topology } G_t \text{ that contains } G_t \text{ on } G_p \text{ such that } cost(G_t) \leq c. \]

**Theorem 3:** NSM is NP-hard.

**Proof.** We will show that $M_{2\text{ECSS}} \leq_p \text{NSM}$. Given any instance $\{ G, k \}$ of $M_{2\text{ECSS}}$, we construct an instance of NSM as follows:

- Let $G_t = (V(G), \emptyset)$, denoted as $\text{Empty}_t$.
- Let $G_p = G$.
- Let $c = k$.

Clearly, the construction is polynomial-time computable.

$\langle G, k \rangle \in M_{2\text{ECSS}} \Rightarrow \langle \text{Empty}_t, G, k \rangle \in \text{NSM}$:

Suppose $G'$ is a 2-edge-connected spanning subgraph of $G$ and $|E(G')| \leq k$. Consider $G'$ as a logical topology on the physical topology $G$. Let $M$ be the reflectively-routed mapping from $G$ to $G'$. Under $M$, any single link failure in $G$ will affect at most one logical link in $G'$. And since $G'$ is 2-edge-connected, the failure will not disconnect $G'$. Therefore, $M$ is survivable. Since $M$ is a reflectively routed mapping, $cost(M) = |E(G')| \leq k$. Hence, there is a survivable logical topology $(G')$ that contains $\text{Empty}_t$ on $G$ such that $cost(G') = cost(M) \leq k$, i.e., $\langle \text{Empty}_t, G, k \rangle \in \text{NSM}$.

$\langle G, k \rangle \in M_{2\text{ECSS}} \Leftarrow \langle \text{Empty}_t, G, k \rangle \in \text{NSM}$:

Since there is a survivable logical topology that contains $\text{Empty}_t$ on $G$ such that its cost $\leq k$, MIN-COST$\text{Empty}_t \leq k$ holds. By Theorem 2, we can build a minimal cost survivable logical topology (denoted as $G_{\text{min}}$) that contains $\text{Empty}_t$ on $G$ by only adding reflective logical links. So the logical links in $G_{\text{min}}$ are all from $E(G)$ and the reflectively-routed mapping from $G_{\text{min}}$ to $G$ achieves the minimal cost, i.e., $|E(G_{\text{min}})| = \text{MIN-COST}_{\text{Empty}_t}$. Since $\text{MIN-COST}_{\text{Empty}_t} \leq k$, we have $|E(G_{\text{min}})| \leq k$. Therefore, $G_{\text{min}}$ is a spanning subgraph of $G$ with $\leq k$ edges. Also, $G_{\text{min}}$ must be 2-edge-connected because it is survivable. So, there is a 2-edge-connected spanning subgraph with $\leq k$ edges for $G$. Hence $\langle G, k \rangle \in M_{2\text{ECSS}}$.

**C. An Improved ILP Formulation**

By Theorem 2, we can find a solution to the new survivable mapping problem by only adding links in $E_p - E_t$ to the given logical topology. This helps to decrease the number of variables in ILP1. Instead of considering all possible pairs $st \in E(K_n)$, only $st \in E_t \cup E_p$ need to be considered as potential logical links in the result logical topology, which leads to the following improved ILP formulation.

\[
\text{Minimize } \sum_{st \in E^d_p} f_{ij}^st \quad \text{Subject to:}
\]

(a) Flow conservation constraints:

\[
\sum_{ij \in E^d_p} f_{ij}^st - \sum_{ji \in E^d_p} f_{ji}^st = \begin{cases} 1 & \text{if } s = i \\ 0 & \text{if } t = i \\ -1 & \text{otherwise} \end{cases} \\
\forall i \in V, \forall st \in E_t - E_p.
\]

(a'). Reflectively-routed constraints:

\[
f_{ij}^st = x^st, \forall st \in E_p. \quad (1)
\]

\[
f_{ij}^st = 0, \forall st \in E_p, ij \in E^d_p \text{ and } (i \neq s \lor j \neq t). \quad (2)
\]

(b) Survivability constraints: Same as those in ILP1.

(c). Deletion of existed logical links is not allowed: Same as those in ILP1.

(d) Integer constraints:

\[
f_{ij}^st \in \{0, 1\}, \forall ij \in E^d_p, \forall st \in E_t \cup E_p.
\]

\[
x^st \in \{0, 1\}, \forall st \in E_t \cup E_p.
\]

The flow conservation constraints in (a) are only used for logical links in $E_t - E_p$ because other logical links are reflective and will be reflectively-routed, which is guaranteed by reflective-routed constraints in (a'). And in (a'), constraints (1) ensure that reflective logical links are routed only when they are included in the result logical topology (i.e., $x^st = 1$). The existence of a result logical topology and corresponding reflectively-routed mapping is guaranteed by Theorem 2.

For convenience, the improved ILP formulation will be denoted as ILP2.

**V. NUMERIC RESULTS**

In our simulation study, a 12-node 18-link random graph shown in Figure 4 is used as the physical topology (PHY TOP). Two groups of logical topologies are used: one consists of 100 12-node 15-link random topologies (GROUP1), the other consists of 100 12-node 18-link random topologies (GROUP2). All logical topologies as well as the physical topology are 2-edge-connected. All simulations are...
run on a Sun Ultra 10 workstation with a 440MHz CPU, 256MB RAM, and 4GB virtual memory. CPLEX8.1 is used as the ILP solver.

A. Comparison of Solutions to the New and Old Survivable Mapping Problems

In section I, we have discussed that the solution to the new survivable mapping problem not only can fix a non-survivable logical topology, but also may reduce the cost of the minimal survivable mapping. To illustrate these effects, we run ILP1 (or ILP2 since both ILP1 and ILP2 solve the new survivable mapping problem optimally) and the ILP provided in [15], denoted as ILP\_ORIG, on GROUP1 and GROUP2 over PHY\_TOP. ILP\_ORIG solves the original survivable mapping problem that does not allow adding new links to the given logical topology. That is, ILP\_ORIG finds a minimal cost survivable mapping for a given logical topology and physical topology if a survivable mapping exists.

Table I shows the improvement made by ILP1 (or ILP2) over ILP\_ORIG. As shown in the table, in GROUP1, 53 out of 100 logical topologies are not survivable, which can be fixed by adding logical links. Among the survivable ones, 28 out of 47 (about 60.0%) can achieve smaller cost by introducing new logical links. Moreover, among these 28 improved logical topologies, the maximum/average cost saving ratio is 17.4%/7.0%. While in GROUP2, we have 90 survivable logical topologies, among which 28 (about 31.1%) can be improved by adding new logical links. And the maximum/average cost saving ratio is 10.4%/3.7%. It can be seen that the overall improvement on GROUP2 is less than that on GROUP1, which suggests that the new survivable mapping problem exhibits more significance on sparser logical topologies than on denser ones. This is intuitive because denser logical topologies are generally closer to survivable, and the room to improve survivable mapping cost is generally smaller in denser logical topologies.

B. Performance Comparison between ILP1 and ILP2

To evaluate the running time improvement made by ILP2 over ILP1, we run both ILP1 and ILP2 on GROUP1 over PHY\_TOP. The average running time over 100 logical topologies on PHY\_TOP taken by ILP1 and ILP2 are measured. As shown in Table II, the average running time taken by ILP2 is much less than that taken by ILP1, and the average speedup achieved by ILP2 over ILP1 is $544.06sec/27.89sec \approx 20$. Table II also compares the number of variables/flow conservation constraints/survivability constraints in ILP1 and ILP2. It can be seen that ILP2 has less variables and flow conservation constraints than ILP1 does, which explains why ILP2 runs faster than ILP1.

VI. CONCLUSION

We propose the following new survivable mapping problem: given a physical topology and a logical topology, compute a minimal cost survivable logical topology that contains the given logical topology as well as the corresponding minimal cost survivable mapping. The problem is significant for two reasons: 1) If the given logical topology is not survivable, we can augment the given logical topology to make it survivable; 2) If the given logical topology is survivable, we may reduce the cost of the survivable mapping by augmenting the given logical topology. We have proved that a solution to the new survivable mapping problem always exists and provided a straightforward ILP formulation (ILP1) to solve the problem. Furthermore, we have proved that we can find a solution to the problem by only adding reflective logical links to

![PHY\_TOP: the 12-node 18-link random physical topology.](image)

**Fig. 4.** PHY\_TOP: the 12-node 18-link random physical topology.

### Table I

**IMPROVEMENT BY ILP1 (OR ILP2) OVER ILP\_ORIG**

<table>
<thead>
<tr>
<th></th>
<th>GROUP1</th>
<th>GROUP2</th>
</tr>
</thead>
<tbody>
<tr>
<td># non-survivable logical topologies fixed by ILP1 (or ILP2)</td>
<td>53</td>
<td>10</td>
</tr>
<tr>
<td># survivable logical topologies improved by ILP1 (or ILP2)</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>Max(avg) cost saving ratio* among improved survivable logical topologies</td>
<td>17.4%(7.0%)</td>
<td>10.4%(3.7%)</td>
</tr>
</tbody>
</table>

* cost saving ratio is defined as cost computed by ILP\_ORIG - cost computed by ILP1 (or ILP2)/cost computed by ILP\_ORIG

### Table II

**PERFORMANCE COMPARISON OF ILP1 AND ILP2**

<table>
<thead>
<tr>
<th></th>
<th>ILP1</th>
<th>ILP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variables</td>
<td>$n(n-1)/2 + n(n-1)/2$</td>
<td>$</td>
</tr>
<tr>
<td>Flow conservation constraints</td>
<td>$n^2(n-1)/2$</td>
<td>$</td>
</tr>
<tr>
<td>Survivability constraints</td>
<td>$</td>
<td>E_p</td>
</tr>
<tr>
<td>Average running time (second)</td>
<td>544.06</td>
<td>27.89</td>
</tr>
</tbody>
</table>
the given logical topology, and the result logical topology has a reflectively-routed survivable mapping that achieves the minimal cost. This result helped us get a simple NP-hardness proof for the new survivable mapping problem and an improved ILP formulation (ILP2) that solves the problem more efficiently. Simulation results demonstrate the significance of the new survivable mapping problem as well as the speedup of ILP2 over ILP1.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation under Grant No. 0237592.

REFERENCES