Scheme for Optical Network Recovery Schedule to Restore Virtual Networks after a Disaster

Presented by Chen Ma

24 Oct. 2014

Background

- In network virtualization, multiple virtual networks are mapped over one physical network.
- A single component failure in the physical layer may destroy one ore more virtual networks, while a disaster

How to find a sequence to recover failed physical links with minimizing damage?

- multiple steps, and during each step, the restored resources will be allocated to recover some of failed components to arrive a partially recovered physical network.

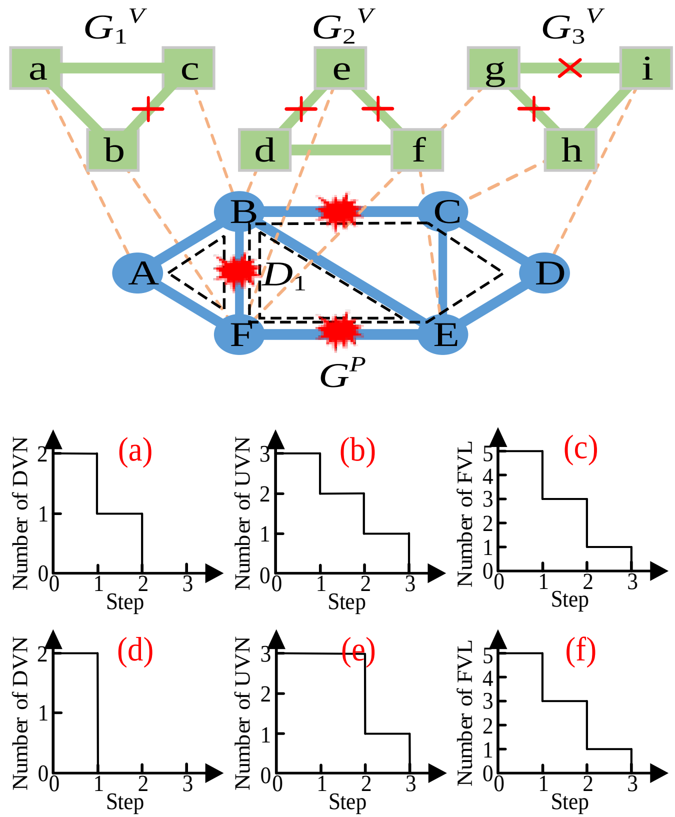
Scenario

- In our model, only one physical links is recovered in each step.
- Three indices are used to evaluate the statements of each failure:

✓ The number of disconnected virtual networks (DVN)

✓ The number of un-full virtual networks (UVN)

✓ The number of failed virtual links (FVL)



(a)~(c): *BF, EF, BE*
(d)~(e): *EF, BF, BE*

Objective

- We select to minimize the DVN as our objective, where the capacity of active virtual links can be improved to guarantee the QoS of a connected virtual network.
- Assumed the number of DVN in each step k is ϕ_k, so the total number of DVN in the recovery schedule is:
  - $\Phi = \sum_{k \in \{1..K\}} \phi_k$
- If we give a weight $w_{s,k}$ to denote the s-th virtual network in the k step, and a binary variable $r_{s,k}$ to describe the s-th virtual network is connected or not in the step k., and the weight of the objective is:
  - $W = \sum_{k \in \{1..K\}} \sum_{s \in \{1..S\}} w_{s,k} \times r_{s,k}$

Problem Statement

Given:

- $G^P(V^P, E^P)$: Physical network, where V^P is the set of physical nodes, and E^P is the set of physical links.
- $G^V\{G_s^V\}$: The set of virtual networks, which includes $G_1^V, G_1^V, \ldots, G_S^V$.
- $G_s^V(V_s^V, E_s^V)$: Virtual network, where s is the sequence of virtual network, V_s^V is the set of virtual nodes and E_s^V is the set of virtual links in the s-th virtual network.
- $D(E^D)$: Disaster, where E^D is the set of physical links in the disaster area.
- $M_{(i,j),(m,n),s}$: Mapping relation, where virtual link (m,n) in the s-th virtual network is mapped to physical link (i,j).
- $w_{s,k}$: Weight of s-th virtual network in step k.
- K: Number of physical links in D.
- S: Number of virtual networks.

Ojbective:

- Minimize total weight of DVN.

Output:

- Restored sequence of failed physical links.

ILP Model-1

To simplify the model, we only consider the links in the physical network, and we don't allocate the spectrum resources on each link.

Variables:

- $y_{(i,j),k}$: Binary variable. 1, physical link (i,j) is restored in step k; 0, otherwise.
- $c_{(i,j),k}$: Binary variable. 1, physical link (i,j) is up in step k; 0, otherwise.
- $l_{(i,j),s,k}$: Binary variable. 1, the virtual link (m,n) of the s-th virtual network is disconnected in step k; 0, otherwise.
- $r_{s,k}$: Binary variable. 1, the s-th virtual network is disconnected in step k; 0, otherwise.
- $x_{(p,q),(m,n),s,k}$: Binary variable. 1, the path from virtual node p to virtual node q uses virtual link (m,n) in s-th virtual network in step k; 0, otherwise.
- $h_{(p,q),(m,n),s,k}$: Binary variable. 1, the path from virtual node p to virtual node q uses virtual link (m,n) and (m,n) is disconnected in s-th virtual network in step k; 0, otherwise.

Objective:

$$\min \sum_{k \in \{1..K\}} \sum_{s \in \{1..S\}} w_{s,k} \times r_{s,k} \tag{4}$$

ILP Model-2

$$\sum_{k\in\{1..K\}} y_{(i,j),k} \quad \forall (i,j)\in E^D \tag{5}$$

$$\sum_{(i,j)\in E^D} y_{(i,j),k} = 2 \quad \forall k \in 1..K \tag{6}$$

$$y_{(i,j),k} = y_{(j,i),k} \quad \forall k\in\{1..K\},(i,j)\in E^D \tag{7}$$

$$c_{(i,j),k=1} = 0 \quad \forall k\in\{1..K\},(i,j)\in E^D \tag{8}$$

$$c_{(i,j),k} = \sum_{k'\in\{1..k\}} y_{(i,j),k'} \quad \forall k\in\{2..K\},(i,j)\in E^D \tag{9}$$

$$l_{(m,n),s,k} = \bigvee_{(i,j)\in E^D} M_{(i,j),(m,n),s}\times(1-c_{(i,j),k}) \quad \forall (m,n)\in E^V_s, k\in\{1..K\} \tag{10}$$

$$\sum_{(m,n)\in E^D} x_{(p,q),(m,n),s,k} - \sum_{(n,m)\in E^D} x_{(p,q),(n,m),s,k} = \begin{cases} 1 & n=p \\ -1 & n=q \\ 0 & other \end{cases} \quad \forall p,q,m\in V^V_s, p\neq q, (m,n)\in E^V_s, k\in\{1..K\} \tag{11}$$

$$h_{(p,q),(m,n),s,k} = x_{(p,q),(m,n),s,k} \bigwedge l_{(m,n),s,k} \quad \forall p,q\in V^V_s, p\neq q,(m,n)\in E^V_s, k\in\{1..K\} \tag{12}$$

$$r_{s,k} = \bigvee_{p,q\in E^V_s, p\neq q,(m,n)\in E^V_s} h_{(p,q),(m,n),s,k} \quad \forall s\in\{1..S\}, k\in\{1..K\} \tag{13}$$

Constraint (5) means that each physical link is repaired one and only one time in the whole recovery schedule. Constraints (6) and (7) denote that one and only one physical link is repaired at each step.

Constraint (8) and (9) find the statements of physical links in each step.

Constraint (10) figures out the statements of virtual links.

Constraint (11) finds the path from any two virtual nodes of one virtual network.

Constraint (12) finds the statement of each virtual link employed by two virtual node.

Constraint (13) finds the virtual network is disconnected or not.

Heuristic Algorithm-1

Algorithm 1 Connectivity of virtual network

Input:

- $G^P(V^P, E^P)$: Physical network;
- $G_s^V(V_s^V, E_s^V)$: The s-th virtual network;
- $c_{(i,j),k}$: Statement of physical link (i,j) in step k.
- $M_{(i,j),(m,n),s}$: Mapping relationship.

Output:

- $G_s^V(V_s^V, E_s^V)$: The s-th virtual network.

$r_{s,k} \Leftarrow 0$
for all $(m,n) \in E_s^V$ **do**
  for all $(i,j) \in E^P$ **do**
    if $c_{(i,j),k} \bigvee M_{(i,j),(m,n),s} = 1$ **then**
      $l_{(m,n),s,k} = 1$;
      break
    end if
  end for
end for
for all $p, q \in V_s^V, p \neq q$ **do**
  Find the virtual path from p to q by Dijkstra Algorithm using $l_{(m,n),s,k}$ as the weight.
  The statement of path from p to q is $L_{(p,q)} = \bigvee_{(m,n)\in E_s^V} l_{(m,n),s,k} \bigvee h_{(p,q),(m,n),s,k}$
  if $L_{(p,q)} = 1$ **then**
    $r_{s,k} \Leftarrow 1$
    break
  end if
end for
return $r_{s,k}$

Algorithm 1: Connectivity of virtual network

Main Idea:

Find the shortest path between any two virtual nodes. If the one of the paths is disconnected, the virtual network is disconnected.

Heuristic Algorithm-2

Algorithm 2 MW-DVN Algorithm

Input:

- $G^P(V^P, E^P)$: Physical network.
- $G^V\{G_s^V\}$: The set of virtual networks, where G_s^V is the s-th virtual network.
- $w_{s,k}$: The weight of s-th virtual network in step k.
- E^D: The set of physical links in D.
- K: Number of physical links in D

Output:

- $y_{(i,j),k}$: Restored sequence of failed physical links in D.
- W: Total connectivity weight.

$y_{(i,j),k} \Leftarrow 0, \forall (i,j) \in E_d^D, k \in \{1..K_d\}, W \Leftarrow 0$
for all $k \in \{1..K\}$ **do**
  $(i,j) \Leftarrow$ ***null***, $W_k' \Leftarrow$ ***max***
  for all $(i',j') \in E_d$ **do**
    $c_{(i',j'),k} \Leftarrow 1, \Delta_{(i',j'),s,k} \Leftarrow 0$
    for all $s = \{1...S\}$ **do**
      Find the $r'_{(i',j'),s,k}$ by **Alg.1** and calculate $\delta_{(i',j'),s,k} = w_{s,k} \times r'_{(i',j'),s,k}$
      $\Delta_{(i',j'),k} = \Delta_{(i',j'),k} + \delta_{(i',j'),s,k}$
    end for
    if $\Delta_{(i',j'),s,k} < W_k'$ **then**
      $(i,j) \Leftarrow (i',j')$, $W_k \Leftarrow \Delta_{(i,j),s,k}$
    end if
  end for
  $y_{(i,j),k} \Leftarrow 1$, $W = W + W_k'$
end for
return $y_{(i,j),k}$, W

Algorithm 2: Minimum Weight DVN Algorithm

Main Idea:

In each step,

1. We recover one of the failed link, and then calculate the weight of virtual network.
2. We do this for all failed links in this step, and calculate their weight.
3. We select the failed link with minimum weight as the restored link in this step.

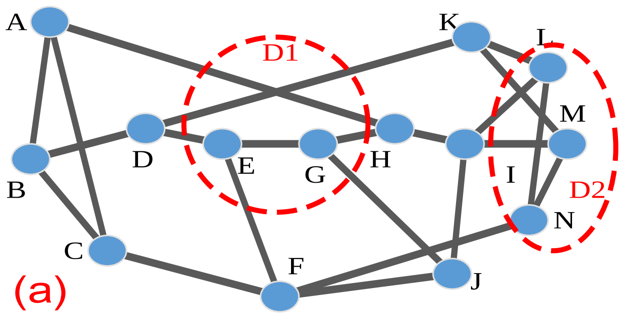
Simulation Results - ILP

Preparation (OFC)

1. NSFNet with two disaster areas as the physical network.
2. 10 virtual networks with 4 nodes as the virtual networks.
3. The connected probability of each two nodes in a virtual network is 50%.
4. The virtual nodes are randomly mapped to the physical nodes.
5. The virtual links are mapped to the physical network by the shortest physical path between its two endpoints.

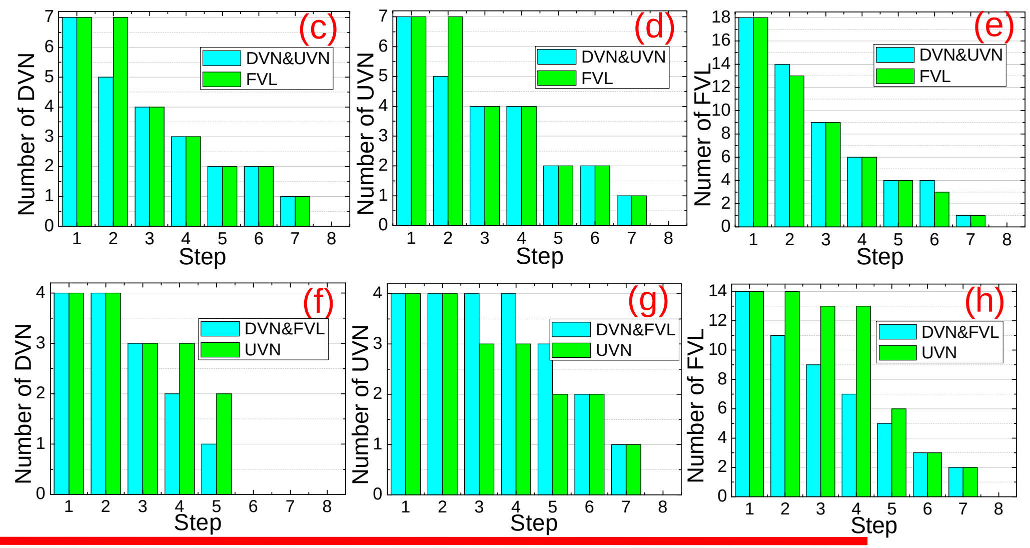
Objective

1. Minimum number of DVN
2. Minimum number of UVN
3. Minimum number of FLV



(b)

D1	DVN / UVN	(A,H)(D,K)(D,E)(E,F)(E,G)(G,H)(G,J)
	FLV	(D,K)(A,H)(D,E)(E,F)(G,H)(E,G)(G,J)
D2	DVN / FLV	(I,L)(I,M)(K,M)(M,N)(F,N)(L,N)(K,L)
	UVN	(K,M)(M,N)(I,M)(F,N)(I,L)(K,L)(L,N)



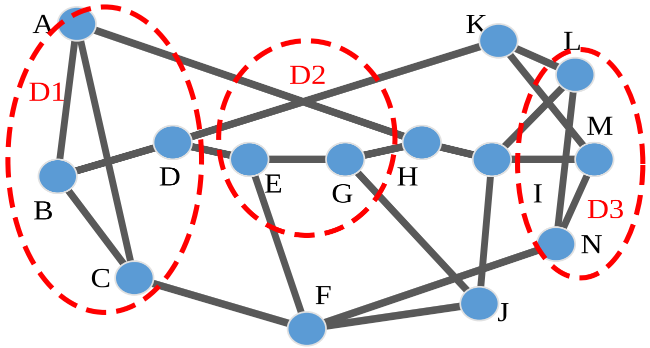
Simulation Results-Heuristic Algorithm

Preparation

1. NSFNet with 3 disaster areas as the physical network, and the capacity of each physical link is 400 spectrum slots.
2. Spectrum consecutiveness and continuity constraints are considered here
3. Virtual networks with 4 nodes as the virtual networks are dynamically generated by Poisson distribution with 1 time period.
4. Bandwidth of each virtual link is randomly selected from 2~4 spectrum slots, and the guard band is 1 spectrum slot.
5. Disasters are generated by uniformly distribution with 100 time periods.

Objective

1. Minimum weight of DVN (future)
2. Minimum number of DVN (future)
3. Minimum number of UVN (future)
4. Minimum number of FVL (future)



Traffic Load		100Erlang		
type	w_dvn	n_dvn	n_lfv	n_uvn
w_dvn	46.951	330.5	1280.76	1171.06
n_dvn	47.126	328.87	1265.88	1165.31
n_uvn	48.3811	337.39	1237.6	1189.33
n_fvl	48.1718	336.18	1271.79	1105.79

Thanks!
Q&A