

NYU TANDON SCHOOL OF ENGINEERING

NETWORK DESIGN AND ALGORITHMS

PROJECT

*TOPIC: Comparison of delay variation for different network resiliency network models*

Submitted By:

Prasad P. Netalkar : N15310512

Sneheil Raz : N18411738

Table of Contents

1. Acknowledgement
2. Abstract

Chapter 1: Introduction

Chapter 2: Proposed Network Resiliency Models

              2.1: Congestion Delay

              2.2: Link Path Re-establishment

              2.3: Demand Flow Re-establishment

Chapter 3: Topology Overview

              3.1: Topology 1- Simple four node scenario

              3.2: Topology 2- Complex real network scenario

Chapter 4: Simulation and Results

              4.1: Experimental Setup

              4.2: Results

                 4.2.1: Flow Allocation (x)

                 4.2.2: Link Capacity (y)

                 4.2.3: Congestion Delay

Chapter 5: GNS3 Practical Implementation

              5.1: Topology 1

Chapter 6: Conclusion

Chapter 7: References

1. ACKNOWLEDMENT
2. ABSTRACT

Resiliency is the ability of a network to provide and maintain an acceptable level of service in the time of failure. Here we are mainly concentrating mainly on the link failures. There can be several reasons for link breakdown, can be due to attack (DoS) or may be due to natural disasters. In the event of failure it is always desirable to have a backup path to fulfil certain demand. We have basically two restoration design models - Link Path and Demand Flow. These restoration design are well established and used within an autonomous system(AS).We tried comparing two restoration models with respect to their flow allocation, Link capacity and delay .Our model aims at minimizing  the congestion delay in the network in the event of failure.

**Keywords:** *Resiliency, Failure, AMPL, Link Capacity.*

CHAPTER 1: INTRODUCTION

To make a Network robust and available all the time, a new dimension needs to be added to the design problems which is only for normal Network operation. This new dimension which takes care of different failure situations is called network resiliency. The different failure situations are specified by the availability status of the links and nodes and, additionally, by (possibly decreased) demand volumes requested for a particular situation. The resulting optimization problems will be referred to as restoration design problems. Its main goal is to protect demand volumes

This project is about formulation and comparison of two re-establishment mechanism based on congestion delay. The mechanisms which we are comparing are Link re-establishment and Demand flow re-establishment. These re-establishment mechanisms can use either dedicated (reserved) or shared capacity. *Dedicated capacity* is used as the term for the spare capacity required to re-establish a link (or path) that is reserved exclusively for re-establishing this link (or path), and that cannot be used for re-establishing other links (or paths). Shared capacity means that a common pool of spare resources is used for re-establishing, typically for restoration, broken links or paths, and that the same capacity unit may be used for protecting different resources in different situations.

We have compared the two mechanisms which we have formulated using Ampl and found out certain inferences. Also, we have used GNS3 to simulate the same Network topology, routers running OSPF and verify the results of Ampl with actual Cisco router results in case of normal operation as well as in case of a link failure.

CHAPTER 2: PROPOSED NETWORK RESILIENCY MODELS

We proposed two network resiliency models which minimizes congestion delay in the network

2.1: Congestion Delay

Congestion in network degrades quality of service and occurs when the link is carrying more than its potential. Typical effects include increase of queuing delay, retransmission and blocking. In the case of network failure it always advisable to use a backup path having least congestion, this in turn reduces burden on the path if it was carrying a flow. Path chosen with lower congestion delay has higher survival rate and reduces overall delay in the network.

Congestion delay is given by: (Convex Function):-

Also , can be called as link load delay.

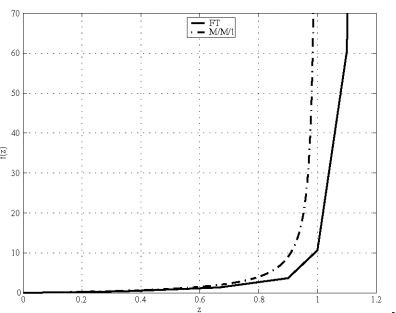
ye: Variable link capacity

ce: Fixed capacity of link

Since it is non linear we use piecewise linear approximation to make it linear. Here our main task is minimizing total congestion delay in the network.

The above function is converted to linear model as shown below:

**Graph displaying delay function and the approximate piecewise model:**

****

The above linear model obtained is added to our formulation with the main goal of reducing congestion in the network.

**Objective Function:**

Where re is the delay in each link.

2.2: Link Path Re-establishment-

Link re- establishment assumes single, total failures of individual links, and restores the entire (or part of) capacity of the failed link on one or several paths between the two end nodes of the link in question (in this way individual flows are re- established indirectly). In link re -establishment the spare capacity is shared. In our model we are assuming link to have a fixed given capacity and our model minimizes total congestion in the network.

We are implementing the following design model:

**LP: DR/CF/BR/CC/LIN/LR+BR** Design with Link Restoration

Link-Path Formulation

**Indices**

d = 1, 2, ..., D demands

p = 1, 2, ..., Pd candidate paths for demand d

e,l = 1,2,...,E links

q = 1, 2, ..., Qe candidate restoration paths for link e

**Constants**

δedp = 1 if link e belongs to path p realizing demand d; 0, otherwise

hd volume of demand d

ξe unit cost of link e

βleq = 1 if link l belongs to path q restoring link e; 0, otherwise

ce Capacity of each link.

**Variables**

xdp0 (non-negative) normal flow allocated to path p of demand d

ye (non-negative) normal capacity of link e

zeq (non-negative) flow restoring normal capacity of link e on restoration path q (non-negative) spare, protection capacity of link e (not used in the

ye’ (non-negative) spare, protection capacity of link e (not used in the normal network operating state)

**Objective**

Minimize (2.2 a)

**Constraints**

*d :1, 2, ..., D*(2.2 b)

*< = ye e: 1,2 ,...,E* (2.2 c)

*e: 1,2,...,E* (2.2 d)

*l: 1,2,...,E e:1,2,...,E* (2.2 e)

2.3: Demand Flow Re-establishment

The path re-establishment mechanisms deal directly with demand flows. Contrary to link re-establishment, the PR mechanisms restore individual flows rather than link capacities. Also, the PR schemes use not only shared spare capacity, but also are also able to reuse the capacity released by the failed flows on those links of the broken paths, which survive the failure. Even here we are assuming link to have a fixed given capacity and our model minimizes total congestion in the network.

**LP: DR/CF/BR/CC/LIN/PR+UR {DR-U} PR with Unrestricted Reconfiguration**

**Link-Path Formulation**

**Indices**

d = 1, 2, ..., D demands

p = 1, 2, ..., Pd candidate paths for demand d

e = 1,2,...,E links

s = 0, 1, ..., S situations

**Constants**

δedp = 1 if link e belongs to path p realizing demand d; 0, otherwise

hd volume of demand d

χds demand coefficient of demand d in state s, hds=Xds\*hd

ξe unit cost of link e

ce Fixed Capacity of each link.

V**ariables**

αes fractional availability coefficient of link e in state s (0 ≤ αes ≤ 1)

Xdps flow allocated to path p of demand d in state s

ye (non-negative) normal capacity of link e

**Objective**

Minimize *(2.3 a)*

**Constraints**

*d = 1, 2, ..., D s = 0, 1, ..., S* *(2.3 b)*

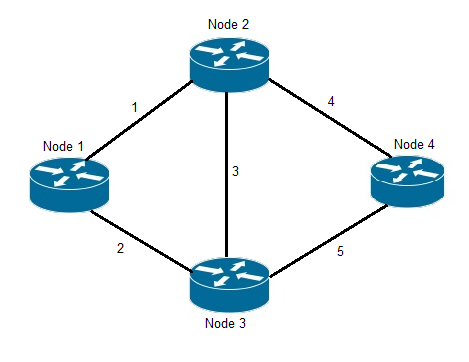
*< = αes\*ye e = 1,2,...,E s = 0, 1, ..., S (2.2 c)*

2.4 Failure Situation:

CHAPTER 3: TOPOLOGY OVERVIEW

We have two ampl model files. The dat files are different for different topologies. In the case of link re-establishment model we provide candidate demand paths and also link backup path in case of failure. Whereas in case of demand flow restoration we give only candidate demand path and in case of failure the whole demand gets rerouted.

3.1: Topology 1- Simple four node scenario



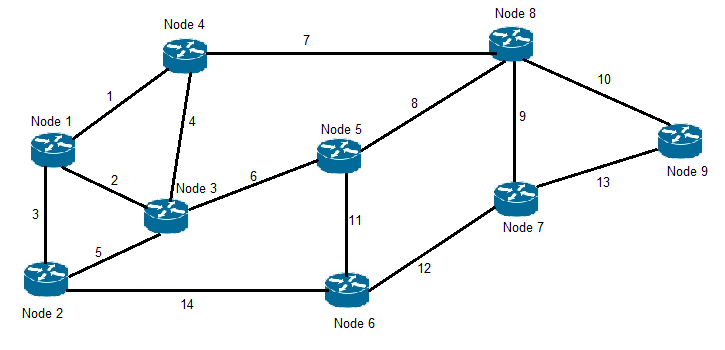
Candidate Demand Path:

|  |  |  |
| --- | --- | --- |
| Demand | Volume | Path |
| Node 2 – Node 3 | 6 | 1. (3) 2. (1 2) 3. (4 5) |
| Node 2 – Node 4 | 4 | 1. (4) 2. (3 5) 3. (1 2 5) |
| Node 1 – Node 3 | 5 | 1. (2) 2. (1 3) 3. (1 4 5) |

Link Restoration Path: For Link Re-establishment Models.

|  |  |
| --- | --- |
| Link | Restoration Path |
| 1 | 1. (2 3) 2. (2 5 4) |
| 2 | 1. (1 3) 2. (1 4 5) |
| 3 | 1. (1 2) 2. (4 5) |
| 4 | 1. (3 5) 2. (1 2 5) |
| 5 | 1. (3 4) 2. (2 1 4) |

3.2: Topology 2- Complex real network scenario



Candidate Demand Path:

|  |  |  |
| --- | --- | --- |
| Demand | Volume | Path |
| Node 1 – Node 5 | 6 | 1. (2 6) 2. (1 7 8) 3. (3 14 11) |
| Node 1 – Node 9 | 3 | 1. (1 7 10) 2. (3 14 12 13) 3. (2 6 8 10) |
| Node 4 – Node 6 | 4 | 1. (4 6 11) 2. (1 2 5 14) 3. (7 8 11) |
| Node 8 – Node 2 | 4 | 1. (8 6 5) 2. (9 12 14) 3. (7 1 3) |

Link Restoration Path: For Link Re-establishment Model.

I have given just two re-establishment path.

|  |  |
| --- | --- |
| Link | Restoration Path (Link no) |
| 1 | 1. (2 4) 2. (3 5 4) |
| 2 | 1. (1 4) 2. (3 5) |
| 3 | 1. (2 5) 2. (1 4 5) |
| 4 | 1. (2 1) 2. (1 3 5) |
| 5 | 1. (3 2) 2. (6 11 14) |
| 6 | 1. (4 7 8) 2. (5 14 11) |
| 7 | 1. (4 6 8) 2. (1 2 6 8) |
| 8 | 1. (7 4 6) 2. (11 12 9) |
| 9 | 1. (8 11 12) 2. (10 13) |
| 10 | 1. (9 12) 2. (8 11 12 13) |
| 11 | 1. (6 5 14) 2. (8 9 12) |
| 12 | 1. (11 8 9) 2. (14 5 6 8 9) |
| 13 | 1. (9 10) 2. (12 11 8 10) |
| 14 | 1. (5 6 11) 2. (3 1 7 8 11) |

CHAPTER 4: SIMULATION AND RESULTS

4.1: Experimental Setup

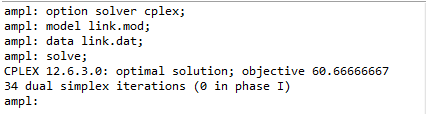
We are using AMPL to formulate our model. CPLEX is used as a optimization tool. Both the model and data files were added to the optimizer and the following results were obtained.

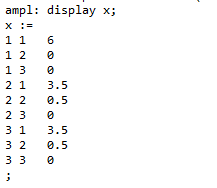
4.2: Results

***Topology 1: 4 Node Scenarios:***

4.2.1a: Flow Allocation (x)

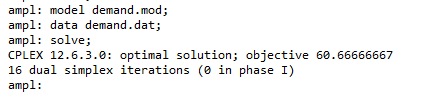
*4 Node Scenarios- Link Path Restoration:*



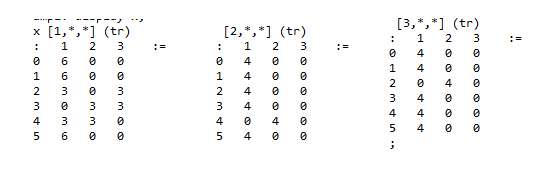


We can see that demand 1 takes the direct path, whereas demand 2 and demand 3 splits their flows between path 1 and 2 respectively. Path 2 is given fewer shares (0.5) because it involves link 3 which is already used .It’s not taking path 3 as it involves three links and to reduce cost that link is not used.

*Demand Flow Restoration:*



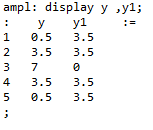
Here total cost for demand flow and link re-establishment is same as we have considered all link failure situations.



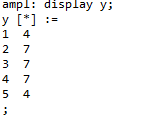
We can see that demand 1 takes the direct path, for link failure situation 0, 1, 5. It splits traffic equally for link failure situation 2, 3, 4 as those links are not available, so it has to split traffic. Also in the case of demand 2 and demand 3, Most of the time it uses direct link but it changes path for link failure 4 and 2 respectively. It is always cheaper to take direct path. Our model works in this case works as follows, consider the link failure situation of 2 where link 2 is failed. Demand 1 uses path 1 and 3 basically uses link (3) and (4 5). In the case of demand 2 no change is observed (direct link 4). And in the case of demand 3 uses path 2 (1 3 links).We can see that link 3 has already been used by demand 1 and also it has been used by demand 3, So these two demands share link 3 and also they try to minimise the congestion on the link 3 by allocating less traffic on link 3.

4.2.2a: Link Capacity (y)

*Link Path Restoration:*



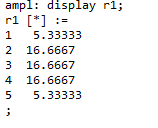
*Demand Flow Restoration:*



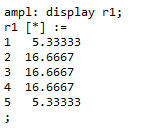
Here we can see that in the case on link path restoration link 3 is given maximum share so it has not been allocated any spare capacity as it reduces congestion cost. Also we noticed that link path and demand flow restoration have same capacity (Flow allocated on links).

4.2.3a: Congestion Delay

*Link Path Restoration:*



*Demand Flow Restoration:*

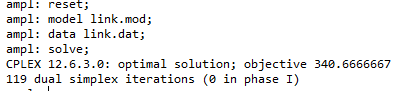


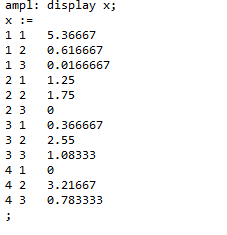
Both the models have same congestion delay.

***Topology 2: Complex real network scenario***

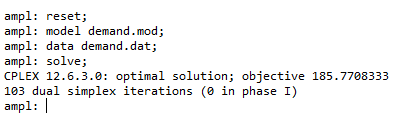
4.2.1b: Flow Allocation (x)

*Link Path Restoration:*

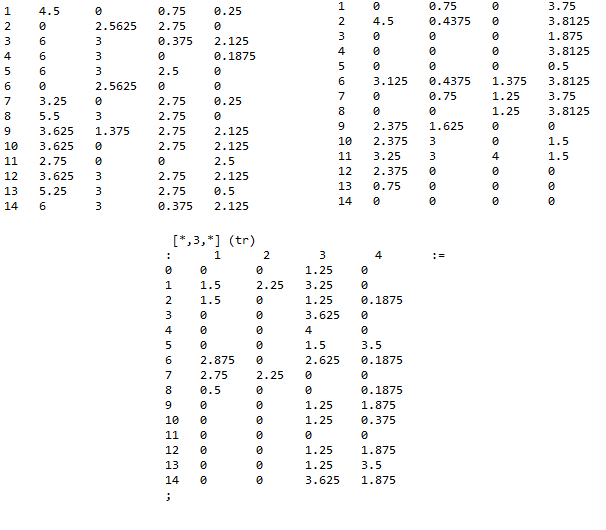
**

**

*Demand Flow Restoration:*

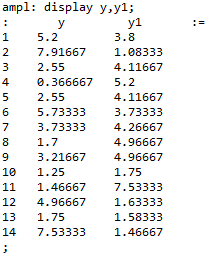
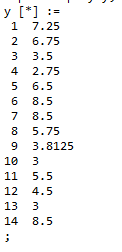
**

We observe a lower congestion cost in the case of Demand flow restoration.



4.2.2b: Link Capacity (y)

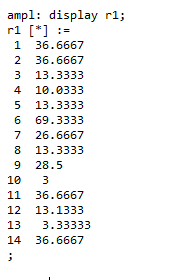
*Link Path Restoration: Demand Flow Restoration:*

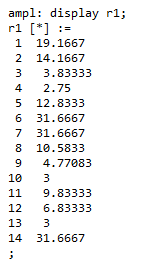
Lower capacity is observed in the case of demand flow restoration.

4.2.3b: Congestion Delay

*Link Path Restoration:*



*Demand Flow Restoration:*



Lower congestion delay is observed in the case of demand flow restoration.