On network traffic concentration and updating interval for proactive recovery method against large-scale network failures

Takuro Horie∗, Go Hasegawa†, Satoshi Kamei‡, Masayuki Murata∗
∗Graduate School of Information Science and Technology, Osaka University
†Cybermedia Center, Osaka University
‡NTT Service Integration Laboratories, NTT Corporation

ATNAC 2009

Contents

- Background
- Our earlier study
- Problem definition
  - Network traffic concentration after recovering from network failures
  - Updating interval of routing configurations against the growth of the network
- New network element addition algorithm
- Evaluation results
- Conclusion

Contents

- Background
- Our earlier study
- Problem definition
  - Network traffic concentration after recovering from network failures
  - Updating interval of routing configurations against the growth of the network
- New network element addition algorithm
- Evaluation results
- Conclusion

Background

- Computer networks
  - Essential infrastructure like water and gas utilities
  - Weak in large-scale network failures caused by earthquakes, terrorist attacks, and software bugs

- Recovering from multiple simultaneous failures and ensuring network connectivity are important challenges

Network recovery methods

- Reactive recovery methods:
  - Recalculate and propagate new routing configurations when failures occur
  - **Merit:** Accommodate to various kinds of network failures
  - **Demerit:** Require long time for routing convergence
- Proactive recovery methods:
  - Calculate and share recovery settings by assuming possible failures before failures occur
  - Select one of the settings to correspond to the detected failures when failures occur
  - **Merit:** Fast recovery by switching the alternative setting
  - **Demerit:**
    Require exact prediction to failures to recover completely

Contents

- Background
- Our earlier study
- Problem definition
  - Network traffic concentration after recovering from network failures
  - Updating interval of routing configurations against the growth of the network
- New network element addition algorithm
- Evaluation results
- Conclusion

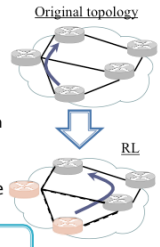
Our earlier study [8]

- Proposed a recovery method from large-scale network failures
  - Based on multiple routing configurations [9]
  - Assume simultaneous network failures
  - Construct network topologies from the original topology
    - Avoid using failed network equipment

[8] T. Horie, G. Hasegawa, S. Kamei, and M. Murata, "A new method of proactive recovery mechanism for large-scale network failures," in Proceedings of AINA-09, May 2009.
[9] A. Hansen, A. Kvalbein, T. Cicic, and S. Gjessing, "Resilient routing layers for network disaster planning," Lecture notes in computer science, vol. 3421, pp. 1097-1105, Apr. 2005.

RRL: Resilient Routing Layers [9] (1/2)

- Calculate topology (RL: Routing Layer) from the original topology
  - Assume the failure of the nodes
    - Called as *safe nodes*
  - Isolate safe nodes from network
    - Set the weight of safe nodes' links to maximum
- Calculate and share routing table according to the RL between all nodes
  - Use the RL and routing table when some safe nodes are down

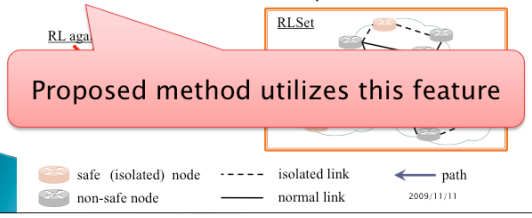


Safe nodes NEVER intermediate between the route of any node pair

All transmitted packets can avoid the failure of safe nodes

RRL: Resilient Routing Layers [9] (2/2)

- Construct and share set of some RLs (RLSet)
  - Isolate each node in one RL
  - Guarantee the recovery from any single-failure
  - Recover from the failure of multiple safe nodes



Proposed method utilizes this feature

Proposed method [8] (1/2)

- Construct effective RLSet in recovering multiple failures
  - Hub-based construction
    - A high-degree (hub) node and as many of its adjacent nodes are isolated in a single RL
  - Attribute-based construction
    - The same attribute* nodes are isolated in a single RL
- Select a RL in two ways
  - Static RL selection:
    - Select a RL by source node packet-by-packet
    - Lower recovery flexibility and smaller overhead
  - Dynamic RL selection:
    - Select a RL by source node and intermediate node hop-by-hop
    - Higher recovery flexibility and larger overhead

* e.g., location, vendor name, version of node OS, etc…

Proposed method [8] (2/2)

- Evaluation results
  - (when 8% of the all network nodes are down)
  - Improve the network reachability from 89% to 99%
  - Keep the average path length sufficiently small
- Two left problems
  - Network traffic concentration after recovering failures
  - Updating interval of routing configurations against the network growth

Objectives

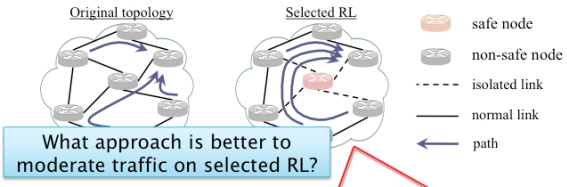
- Suggest some solutions to network traffic concentration after recovering from failures
  - Evaluate network traffic after recovery
- Prolong the updating interval of RLSet
  - Propose a light-weight and distributed algorithm w/o overall RLSet reconstruction

Contents

- Background
- Our earlier study
- Problem definition
  - Network traffic concentration after recovering from network failures
  - Updating interval of routing configurations against the growth of the network
- New network element addition algorithm
- Evaluation results
- Conclusion

Network traffic concentration after recovering from failure

Traffic on the original topology is balanced



What approach is better to moderate traffic on selected RL?

Traffic on the selected RL is concentrated on same links

Contents

- Background
- Our earlier study
- Problem definition
  - Network traffic concentration after recovering from network failures
  - Updating interval of routing configurations against the growth of the network
- New network element addition algorithm
- Evaluation results
- Conclusion

Accommodating to the growth of network

- Mechanism of proposed method
  - Calculate and share RLSet beforehand
  - Difficult to accommodating to the network growth
- Reconstruction of overall RLSet every network growth
  - Ideal for accommodating to the network growth
  - Exhaust network resources

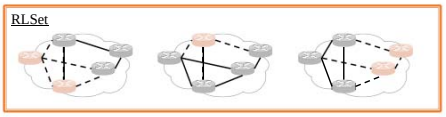
Propose new network element addition algorithms w/o overall RLSet reconstruction

Contents

- Background
- Our earlier study
- Problem definition
  - Network traffic concentration after recovering from network failures
  - Updating interval of routing configurations against the growth of the network
- New network element addition algorithm
- Evaluation results
- Conclusion

Link addition algorithm

1. Add new link to all RLs in RLSet
2. Isolate the new link in the RL in which the new link connects to at least one safe node
3. Recalculate the route between the nodes connected by the new link



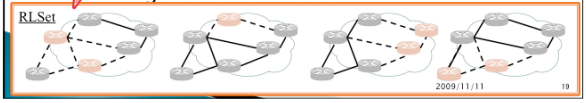
Node addition algorithm

The packets generated by new node MUST pass one safe node in this RL

This algorithm can degrade recovery performance

Evaluation with network growth is necessary

RLSet



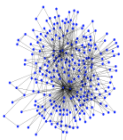
Contents

[3] A. Barabasi and R. Albert, "Emergence of scaling in random networks," Science, vol. 286, no. 5439, pp. 509-512, Oct. 1999.

Evaluation settings

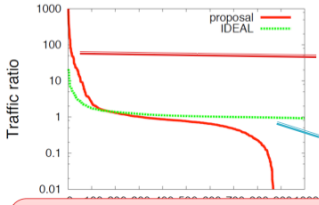
- Topology
  - AS-level network topology administrated by JPNIC for traffic evaluation
    - 259 nodes, 1162 links
  - BA model topology [3] for performance evaluation with network growth
    - Start with 259 nodes and 1030 links
    - Add a new node with 4 links until 359 nodes and 1430 links
- Failure
  - The failure of directly interconnected nodes
- All following evaluations:
  - Use hub-based RLSet
  - With dynamic RL selection mechanism
  - With two node failures

JPNIC



BA model

Traffic ratio after recovering



Traffic of proposed method is extremely concentrated on specific links

IDEAL: The result with the routing table recalculated after failure detection

Need different approaches since traffic are too concentrated

e.g., packet priority control, network bandwidth design

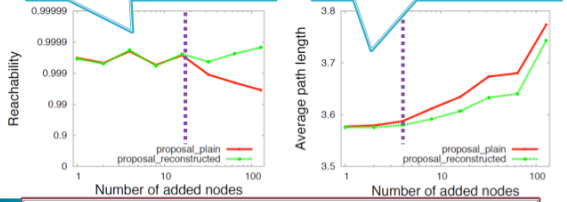
Traffic amount: The number of paths through the link

Performance with network growth

RLSet should be reconstructed every network growth by 5-10% to keep reachability and path length

Keep reachability w/o overall reconstruction until join of 16 nodes

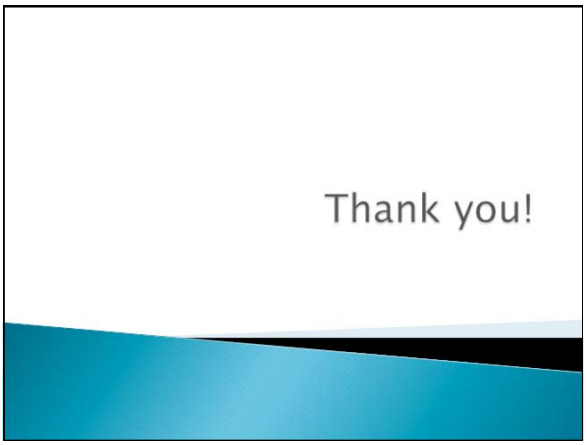
Keep average path length w/o overall reconstruction until join of 8 nodes



plain: NOT reconstructed RLSet
reconstructed: RLSet that is reconstructed every new node join

Conclusion

- Conclusion
  - Need different approaches to moderate network traffic concentration
    - e.g., packet priority control, network bandwidth design
  - Propose the algorithm to add nodes and links to routing configurations w/o overall reconstruction
    - Routing configurations should be recalculated every network growth by 5-10% to keep performance
- Future works
  - Evaluate proposed method with different networks and different conditions
  - Apply proposed method to the routing in the overlay networks


Thank you!