INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
www.ijrcar.com

Vol.2 Issue.12, Pg.: 41-47
December 2014

INTERNATIONAL JOURNAL OF
RESEARCH IN COMPUTER
APPLICATIONS AND ROBOTICS
ISSN 2320-7345

FASTEST METHOD TO FIND ALTERNATIVE RE-ROUTE

[1]**M.JothiLakshmi,** M.Sc., M.Phil. [2]**C.Theebendra,** M.Sc., M.Phil. [3]**M.K.Pavithra,** [4]**S.Yuvabarathi**

Asst. Professor[1+2], Research scholar[3+4], Department of Computer Science, Vivekanandha College Of Arts & Sciences For Women (Autonomous), Elayampalayam, TamilNadu, India[1+2+3+4.]

Abstract

Routing is the concept of sending the packets from the source to destination through the certain paths. In some cases before the Interior Gateway Protocol (IGP) there may be a link failure between the source and destination. To reach the target the regional sub network service provider encountered to find a new path which already exists. Certainly multicast nodes are involved and spare links are used. In our scheme **Ring Topology** are used to overcome the single path failure. We illustrate the method and prove that it will find a path if one exists.

Keywords*:* Ring Topology, Routing Protocols, Alternate Routing, Interior Gateway Protocol, Node Failure.

1. INTRODUCTION

A Routing Protocol select route between two nodes on a computer network. Each router has a priori knowledge about the networks attached to it directly. A routing protocol shares this information first among immediate neighbours, and then throughout the network. This way, routers gain knowledge topology of the network.

Consider a source node *s* sending data to destination node d .Suppose some link (i, j) on the shortest path from *s* to d fails. An IGP will find an alternate path from *s* to *d* that avoids (i,j). To overcome this we implementing Ring Topology network in which each node connected with to other nodes to provide continuous path between source and destination. If any one of a node is failure or break, the alternative route or multipath can be introduced.

The source-specific multicast is the simplest model for multicast where source node is fixed and the receivers will never send data to the multicast. Any failed link in the path will disrupt the service to some nodes. The number of nodes affected could be very large especially when the failure is at the proximity of the root node.

The standard solution is to reconstruct the multicast node after a link failure is detected. The fast reroute restores the multicast node without route re-convergence and therefore, shortened the disruption. This scheme pre installs another set of routes on each multicast routers.

2. BODY TEXT

2.1. Failure recovery

Techniques developed for fast recovery from single-link failures provide more than one forwarding edge to route a packet to a destination. Whenever the default forwarding edge fails or a packet is received from the node attached to the default forwarding edge for the destination, the packets are rerouted on the backup ports. In the authors present a framework for IP fast reroute detailing three candidate solutions for IP fast reroute that have all gained considerable attention. When a forwarding link on a tree fails, the packet may be switched to the other node.

Types of failure:

i) Link Failure

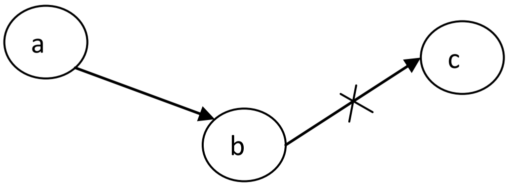


Figure 1: Link Failure

ii) Node Failure

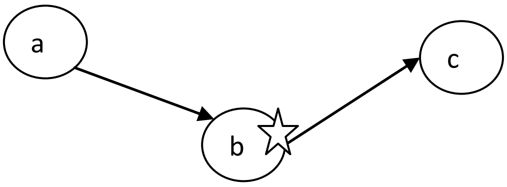


Figure 2: Node Failure

2.1. Fast re-route method

We now present the details of the method. Let $G = (N,A)$ be an undirected connected graph with node set N and arc set A. For $x \in N$, let $N(x)$ be the set of neighbors of x, where a neighbour of x is a node one arc away from x. We associate with each undirected arc $(i, j) \in A$ a cost $c(i, j)$, and require each $c(i, j)$ to be a positive integer. (The integer valued restriction can always be met by approximating, to the desired accuracy, each arc cost by an improper fraction, and then multiplying all the fractions by the least common multiple of the fraction denominators.) For $i, j \in N$, let $c\_(i, j)$ be the cost of the shortest path in G between i and j. When using $Route(s, d)$ for fast re-route in the event of an arc failure, which is the target application, $c\_(i, j)$ represents the shortest path cost *before* the IGP has re-converged in response to the link failure.

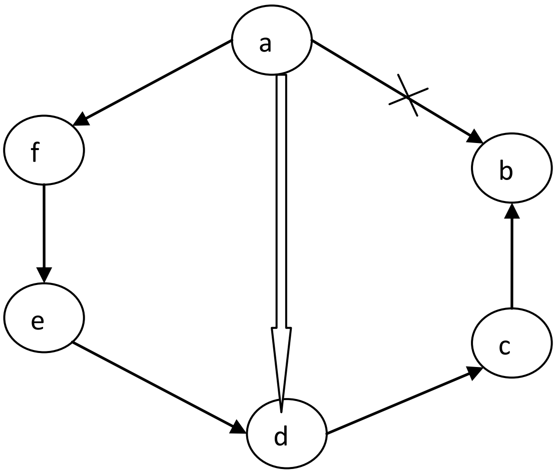


Figure 3: Fast Reroute

Source: a, Destination: b, Failed Path: a→b, Re Route Path: a→d.

2.3. Multipath routing

Multipath routing is a promising routing scheme to accommodate these requirements by using multiple pairs of routes between a source and a destination. Multipath routing is the routing technique of using multiple alternative paths through a network, which can yield a variety of benefits such as increased bandwidth, or improved security. The multiple paths computed might be overlapped, edge-disjointed or node-disjointed with each other. Extensive research has been done on multipath routing techniques.

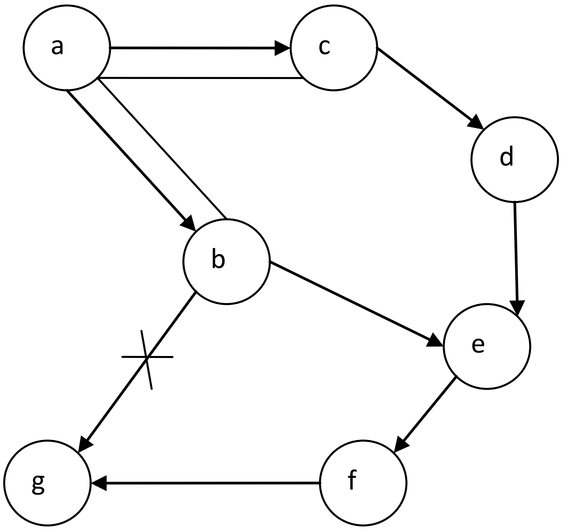


Figure 4: Multipath Routing

Source: a, Destination: g, Failed Path: b→g.
Multipath: 1st path is a-b-e-f-g
2nd path is a-c-d-e-f-g.

2.4. Method

Procedure *Route*(*s*, *d*)
1 **initialize:** $P = \emptyset$, $\Delta(n) = 0$ for $n \in N$, and $x = s$;

```
while $(x \_\neq d)$ {
    Let $Y = \{y \in N(x) \mid \Delta(y) = \min n \in N(x)\Delta(n)\}$;
    Pick any $y \in Y$ for which the sum
    $C (x, y) + c^*(y, d)$ is smallest;
    Set $\Delta(x) \leftarrow \Delta(x) + 1, P \leftarrow \{P, x\}$,
    and send the packet and $P$ from $x$ to $y$;
    Set $x \leftarrow y$;
}
```

Explanation: Let s and d be the source and destination which is connected to directed graph with node N.

If x $\in$ N then x is a node away from arc (i, j). The arc should be positive integer. If i, j $\in N$ then cost is c*(i, j) be the shortest path.

If the route s, d is used for fast re-route in arc failure the target application, c*(i, j) represents the shortest path cost before IGP.

If route (s, d) < p then order of list node have to visit p🡐{p,x} means that x is inserted after the rightmost element in *P*. Also, Δ (*n*) is *the multiplicity* of node *n*, indicating how many times *n* has been visited by the current packet.

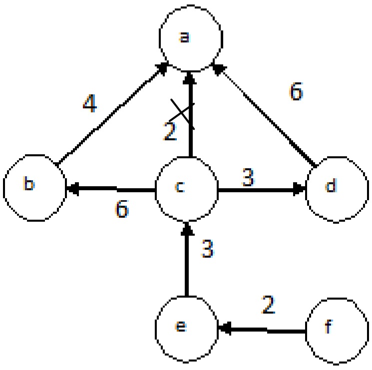


Figure 5: Finding re-route with low cost

Source: f, Destination: a.

Failure path: f-e-c-a.

Available path: 1st path: f-e-c-b-a=15.

2nd path: f-e-c-d-a=14.

The 2nd path is the shortest re-route path from source to destination with low cost.

2.5 link failure algorithm description

In our algorithm is based on sequential search in the primary link, which we call SS LINK. It contains the following steps.

1) Init: Set the backup port of each node to null, i.e., $bn = 0(n = 2; : : : ;N)$.

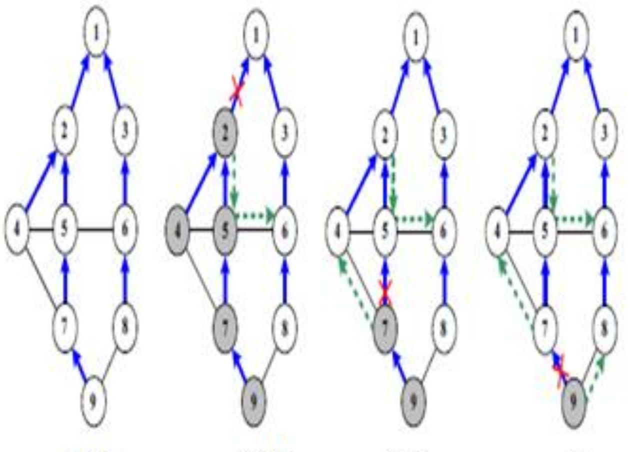
2) Explore the primary link T(1) using depth-first search. For each node *n* (*n* = 2*; : : : ;N*), assume its primary port *pn* fails (i.e., link *n ! pn* fails) and do the following:

a) If *bn* 6= 0, the backup port of node *n* is already found, go back to step 2 to process the next node; otherwise, continue to the next step.

b) The failure disconnects a sub-link T(*n*) from the primary link, where *n* is the root of the sub-link. Dye the nodes in T(*n*) black and all the other nodes in the topology white. The forwarding path from each white node is not affected by the failure.

c) In T(*n*), use breadth-first search to find the first node *i* that has a direct link to a white node *j*, set its backup port *bi* = *j*. We call this port i,j an *exit* of sub-tree T(*n*).

d) If i ≠ n, find the path from *n* to *i* in T (*n*). Suppose the path is *n* -> *m*1 ->*m*2 ...->*mL* -> *i*. Set the corresponding backup ports as *bn* = *m*1, *bm*1 = *m*2, . , *bmL* = *i*. Go back to step 2.



6(a) 6(b) 6(c) 6(d)

Figure: 6(a) Primary nodes , 6(b) Failure 2-1 6(c) failure 7–5, 6(d) failure 9–7.

Figure 6 shows the procedure of using LINK on the depth-first search path 2–5–7–9.

1) Failure 2–1 detaches sub-tree T(2) from the primary link. Using breadth-first search, an exit 5-> 6 is found and the rerouting path is 2->5->6. Thus, we set *b*2 = 5 and *b*5 = 6 (Figure 6(b)).

2) Failure 5–2 creates sub-tree T(5), the search is skipped since *b*5 ≠ 0.

3) Failure 7–5 dyes T(7) black, and the search immediately yields *b*7 = 4 (Figure 6(c)).

4) Failure 9–7 dyes T(9) black, the algorithm sets *b*7 = 4(Figure 6(d)).

2.6. Algorithm properties optimality

Theorem 1*:* LINK minimizes the number of switchovers in (1) if the primary tree is obtained using minimum hop routing.

Proof:

When the primary port of node *k* fails, the exit of T(*k*) is found using breadth first search. Therefore, the hop count from node *k* to the exit is minimized (since the primary tree is based on minimum hop routing). This minimizes the number of switch-overs because choosing any other exit requires more nodes to use backup ports. Since LINK minimizes the number of switch-overs under any possible failure, it achieves the optimality in (1).

Complexity:

The algorithm has low computation complexity. Although it contains two nested searches in the tree, the CPU cycles consumed by each step is very limited. In step 2a, a node is immediately skipped if its backup port is already found. In step 2c, the algorithm only checks if a node has a white neighbor ,thus requires very little computation. In step 2d, the path from *n* to *i* is exactly the reverse of the primary path from *i* to *n*, which does not require complicated route calculation. In particular, each router only runs a part of the algorithm when link is implemented in a distributed manner.

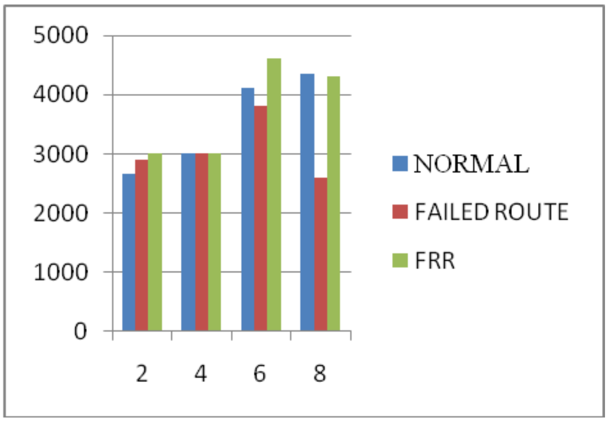
For node *n*, it finds its backup port *bn* and stops immediately. Denote the primary path from node *n* to node 1 as n-> yL->yL->1-> …-> y1->1, the computation is simplified by repeating step 2a to 2d from *y*1,.. *yL; x*. Further complexity reduction can be achieved by first, do not record other nodes' backup ports; second, jump along the search path.

For example, when node 7 in Figure 6 calculates its backup port, it only searches along 2–5–7. When node 2 finds an exit through node 5, the search jumps to the next node on the search path, which is node 7. Meanwhile, it is not necessary to record the backup ports of node 2 and 5.

3. RESULT AND DISCUSSION

These are critical performance metrics of fast rerouting because they have significant impact on router-to-router delay, congestion, and network efficiency. We compare our schemes with shortest path route recalculation to see the difference of the performance metrics. The topologies adopted in our evaluation include several practical networks and randomly generated ones. The results show that our schemes have consistent performance in various networks.
When compare with the other concepts and solution we have four advantages.

(1) The multicast tree can be arbitrary.
(2) There is only a minimal disruption and there would be no packet loss.
(3) Failure protection is available to a wider set of network topologies.
(4) Only a small number of multicast nodes are involved and the number of spare links used is also small.
(5) The schemes have low complexity and can be easily applied to practical networks to substantially shorten service disruption caused by failures



4. CONCLUSION

The results show that our schemes provide almost the same efficiency as route recalculation regardless of the node degree. In addition, we test our schemes in ring topology and find that they generate much higher overall traffic volume compared to route recalculation. Finally, our scheme is designed for link-state routing protocols, it is interesting to study the extension of the schemes for re routing so as to enhance the survivability of inter-domain routing.

5. REFERENCE

[1] Atlas, Ed., “U-turn alternates for IP/LDP fast-reroute,” IETF draft atlas -ip local-protect-uturn-03, Feb. 2006.
[2] Atlas and A. Zinin, Eds., “Basic specification for IP fast reroute: loopfree alternative,” IETF RFC 5286, Sept. 2008.
[3] P. Domschitz and M. Siegel, “Dual ring usage in FDDI,” Computer Communications, vol. 15, no. 7, pp. 447–457, 1992.
[4] A Fast Re-Route Method Eric Rosenberg and James Uttaro, Member, IEEE transactions on networking year 2013.
[5] S. Rai, B. Mukherjee, and O. Deshpande, “IP resilience within an autonomous system: current approaches, challenges, and future direction.
[6] G. Shen and W. D. Grover, “Extending the p-cycle on concept to path segment protection for span and node failure recovery,” J. Sel. Areas Comm., vol. 21, no. 8, 2003.

ACKNOWLEDGEMENT

This research paper is made possible through the help and support from everyone, including: parents, teachers, family, friends, and in essence, all sentient beings. Especially, we would like to dedicate our acknowledgment of gratitude toward the following significant advisors and contributors:

First and foremost, we would like to thank, Mrs M.Jothilakshmi, M.Sc., M.Phil., For her most support and encouragement. She kindly read my paper and offered invaluable detailed advices on grammar, organization, and the theme of the paper, and we would like to thank, Mrs.C.Theebendra, M.sc., and M.Phil. To read my thesis and to provide Valuable advices.

Second we would like to thank, Dr.S.Dhanalakshmi, MCA. M.Phil., Ph.D., M.E., to report of the paper, as well the other professors.

Finally, I sincerely thank my parents, family, and friends, who provide the advice and financial support. The product of this research paper would not be possible without all of them.