International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.1, Issue1, pp-099-107 ISSN: 2249-6645

IMPLEMENTATION OF OBJECT ORIENTED APPROACH TO FAST IP RECOVERY BASED ON MULTIPLE ROUTING CONFIGURATIONS

HARSHA PUJARI
2/2 M.TECH CSE, DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT, TEKKALI
ANDHRA PRADESH, INDIA

U.D.PRASANNA
ASSOC.PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT, TEKKALI
ANDHRA PRADESH, INDIA

Abstract— **To assure fast recovery from link and node failures in IP networks, we present JAVA based recovery scheme called Multiple Routing Configurations (MRC). This proposed scheme guarantees recovery in all single failure scenarios, using a single mechanism to handle both link and node failures, and without knowing the root cause of the failure. MRC is based on keeping additional routing information in the routers, and allows packet forwarding to continue on an alternative output link immediately after the detection of a failure.**

***Index Terms*— MRC; Java; Node,**

I. INTRODUCTION

In recent years the Internet has been transformed from a special purpose network to a ubiquitous platform for a wide range of everyday communication services. The demands on Internet reliability and availability have increased accordingly. A disruption of a link in central parts of a network has the potential to affect hundreds of thousands of phone conversations or TCP connections, with obvious adverse effects. The ability to recover from failures has always been a central design goal in the Internet. This re-convergence assumes full distribution of the new link state to all routers in the network domain. When the new state information is distributed, each router individually calculates new valid routing tables.

This network-wide IP re-convergence is a time consuming process, and a link or node failure is typically followed by a period of routing instability. During this period, packets may be dropped due to invalid routes. This phenomenon has been studied in IGP [1] and has an adverse effect on real-time applications [2]. Events leading to a re-convergence have been shown to occur frequently [3]. Much effort has been devoted to optimizing the different steps of the convergence of IP routing, i.e., detection, dissemination of information and shortest path calculation, but the convergence time is still too large for applications with real time demands .A key problem is that since most network failures are short lived, too rapid triggering of the re-convergence process can cause route flapping and increased network instability.

The IGP convergence process is slow because it is reactive and global. It reacts to a failure after it has happened, and it involves all the routers in the domain. In this project I present a new scheme for handling link and node failures in IP networks. Multiple Routing Configurations (MRC) is proactive and local, which allows recovery in the range of milliseconds. MRC allows packet forwarding to continue over pre-configured alternative next-hops immediately after the detection of the failure. Using MRC as a first line of defense against network failures, the normal IP convergence process can be put on hold. This process is then initiated only as a consequence of non-transient failures. MRC guarantees recovery from any single link or node failure, which constitutes a large majority of the failures experienced in a network [5]. MRC makes no assumptions with respect to the root cause of failure, e.g., whether the packet forwarding is disrupted due to a failed link or a failed router.

The main idea of MRC is to use the network graph and the associated link weights to produce a small set of backup network configurations. The link weights in these backup configurations are manipulated so that for each link and node failure, and regardless of whether it is a link or node failure, the node that detects the failure can safely forward the incoming packets towards the destination on an alternate link. MRC assumes that the network uses shortest path routing and destination based hop-by-hop forwarding.

The shifting of traffic to links bypassing the failure can lead to congestion and packet loss in parts of the network [6]. This limits the time that the

proactive recovery scheme can be used to forward traffic before the global routing protocol is informed about the failure, and hence reduces the chance that a transient failure can be handled without a full global routing re-convergence. Ideally, a proactive recovery scheme should not only guarantee connectivity after a failure, but also do so in a manner that does not cause an unacceptable load distribution. This requirement has been noted as being one of the principal challenges for pre-calculated IP recovery schemes [7]. With MRC, the link weights are set individually in each backup configuration. This gives great flexibility with respect to how the recovered traffic is routed. The backup configuration used after a failure is selected based on the failure instance, and thus we can choose link weights in the backup configurations that are well suited for only a subset of failure instances.

It is important to stress that MRC does not affect the failure free original routing, i.e., when there is no failure, all packets are forwarded according to the original configuration, where all link weights are normal. Upon detection of a failure, only traffic reaching the failure will switch configuration. All other traffic is forwarded according to the original configuration as normal. If a failure lasts for more than a specified time interval, a normal re-convergence will be triggered. MRC does not interfere with this convergence process, or make it longer than normal. However, MRC gives continuous packet forwarding during the convergence, and hence makes it easier to use mechanisms that prevent micro-loops during convergence, at the cost of longer convergence times. If a failure is deemed permanent, new configurations must be generated based on the altered topology.

II. Related Work

There are several proposals for mitigating the impact of link failures on network performance. MRC is to use the network graph and the associated link weights to produce a small set of backup network configurations. The link weights in these backup configurations are manipulated so that for each link and node failure, and regardless of whether it is a link or node failure, the node that detects the failure can safely forward the incoming packets towards the destination. MRC assumes that the network uses shortest path routing and destination based hop-by-hop forwarding.

In this thesis, it is sometimes claimed that the node failure recovery implicitly addresses link failures too, as the adjacent links of the failed node can be avoided. This is true for intermediate nodes, but the destination node in a network path must be reachable if operative ("The last hop problem", [6]). MRC solves the last hop problem by strategic assignment of link weights between the backup configurations.

MRC has a range of attractive features:

- It gives almost continuous forwarding of packets in the case of a failure. The router that detects the failure initiates a local rerouting immediately, without communicating with the surrounding neighbors.
- MRC helps improve network availability through suppression of the re-convergence process. Delaying this process is useful to address transient failures, and pays off under many scenarios [4]. Suppression of the re-convergence process is further actualized by the evidence that a large proportion of network failures is short-lived, often lasting less than a minute [5].
- MRC uses a single mechanism to handle both link and node failures. Failures are handled locally by the detecting node, and MRC always finds a route to the destination (if operational).
- MRC makes no assumptions with respect to the root cause of failure, e.g., whether the packet forwarding is disrupted due to a failed link or a failed router. Regardless of this, MRC guarantees that there exists a valid, preconfigured next-hop to the destination.
- An MRC implementation can be made without major modifications to existing IGP routing standards. IETF recently initiated specifications of multi-topology routing for OSPF and IS-IS, and this approach seems well suited to implement our proposed backup configurations [7][8][9]. The concept of multiple routing configurations and its application to network recovery is not new. Our main inspiration has been a layer-based approach used to obtain deadlock-free and fault-tolerant routing in irregular cluster networks based on a routing strategy called Up*/Down*[10].

General packet networks are not hampered by deadlock considerations necessary in interconnection networks, and hence I generalized the concept in a technology independent manner and named it Resilient Routing Layers [11]. In the graph-theoretical context, RRL is based on calculating spanning sub topologies of the network, called layers. Each layer contains all nodes but only a subset of the links in the network.. The work described in this paper differs substantially from RRL in that, we do not alter topologies by removing links, but rather manipulate link weights to meet goals of handling both node and link failures without needing to know the root cause of the failure. In MRC, all links remain in the topology,

but in some configurations, some links will not be selected by shortest path routing mechanisms due to high weights.

III. Theoretical Background

Network topology is the study of the arrangement or mapping of the elements (links, nodes, etc.) of a network, especially the physical (real) and logical (virtual) interconnections between nodes.

Much work has lately been done to improve robustness against component failures in IP networks [7]. In this section, I focus on the most important contributions aimed at restoring connectivity without a global re-convergence. This indicates whether each mechanism guarantees one-fault tolerance in an arbitrary bi-connected network, for link and node failures, independent of the root cause of failure (failure agnostic). This also indicates whether MRC solve the "last hop problem".

Network layer recovery in the timescale of milliseconds has traditionally only been available for networks using MPLS with its fast reroute extensions [12]. In the discussion below, I focus mainly on solutions for connectionless destination-based IP routing. IETF has recently drafted a framework called IP fast reroute where they point at Loop-Free Alternates (LFAs) as a technique to partly solve IP fast reroute. From a node detecting a failure, a next hop is defined as an LFA if this next hop will not loop the packets back to the detecting node or to the failure. Since LFAs do not provide full coverage, IETF is also drafting a tunneling approach based on so called "Not-via" addresses to guarantee recovery from all single link and node failures [8]. Not-via is the connectionless version of MPLS fast reroute [12] where packets are detoured around the failure to the next-next hop.

To protect against the failure of a component P, a special not-via address is created for this component at each of P's neighbors. Forwarding tables are then calculated for these addresses without using the protected component. This way, all nodes get a path to each of P's neighbors, without passing through ("Not-via") P. The Not-via approach is similar to MRC in that loop-free backup next-hops are found by doing shortest path calculations on a subset of the network. It also covers against link and node failures using the same mechanism and is strictly pre-configured. However, the tunneling approach may give less optimal backup paths, and less flexibility with regards to post failure load balancing. Narvaez et al. [13] propose a method relying on multi-hop repair paths. They propose to do a local re-convergence upon detection of a failure, i.e., notify and send updates only to the nodes necessary to avoid loops.

A similar approach also considering dynamic traffic engineering is presented in [14]. I call these approaches local rerouting. They are designed only for link failures, and therefore avoid the problems of root cause of failure and the last hop. Their method does not guarantee one-fault-tolerance in arbitrary bi-connected networks. It is obviously connectionless. However, it is not strictly pre-configured, and can hence not recover traffic in the same short time-scale as a strictly pre-configured scheme, Nelakuditi et al. [4] propose using interface specific forwarding to provide loop-free backup next hops to recover from link failures.

Their approach is called failure insensitive routing (FIR). The idea behind FIR is to let a router infer link failures based on the interface packets are coming from. When a link fails, the attached nodes locally reroute packets to the affected destinations, while all other nodes forward packets according to their pre-computed interface specific forwarding tables without being explicitly aware of the failure. In another paper, they have also proposed a similar method, named Failure Inference based Fast Rerouting (FIFR), for handling node failures [15]. This method will also cover link failures, and hence it operates independent of the root cause of failure. However, their method will not guarantee this for the last hop, i.e., they do not solve the "last hop problem". FIFR guarantees one-fault-tolerance in any bi-connected network, it is connectionless, pre-configured and it does not affect the original failure-free routing. Our main inspiration for using multiple routing functions to achieve failure recovery has been a layer-based approach used to obtain deadlock-free and fault-tolerant routing in irregular cluster networks [16].

General packet networks are not hampered by deadlock considerations necessary in interconnection networks, and hence we generalized the concept in a technology independent manner and named it Resilient Routing Layers [17]. In the graph-theoretical context, RRL is based on calculating spanning sub topologies of the network, called layers. Each layer contains all nodes but only a subset of the links in the network. In this paper I refine these ideas and adapt them to an IP setting. None of the proactive recovery mechanisms discussed above takes any measures towards a good load distribution in the network in the period when traffic is routed on the recovery paths.

Existing work on load distribution in connectionless IGP networks has either focused on the failure free case [18] or on finding link weights that work well both in the normal case and when the

routing protocol has converged after a single link failure [19].Many of the approaches listed provide elegant and efficient solutions to fast network recovery, however MRC and Not-via tunneling seems to be the only two covering all evaluated requirements. However, MRC offers the same functionality with a simpler and more intuitive approach, and leaves more room for optimization with respect to load balancing.

IV. System Architecture

To send the packets from source node to destination node, first it checks the neighbor nodes of source node. The source node requests the destination node to generate the available paths. If don't select the destination node it asks to give destination node. To select the particular shortest path, and send the packets to the particular destination node. The failures are fairly common in the everyday operation of a network due to various causes such as maintenance, fault interfaces, and accidental fiber cuts.

MRC is based on using a small set of backup routing configurations, where each of them is resistant to failures of certain nodes and links. The original network topology, a configuration is defined as a set of associated link weights. In a configuration that is resistant to the failure of a particular node n, link weights are assigned so that traffic routed according to this configuration is never routed through node n. The failure of node n then only affects traffic that is sent from or destined to n. Similarly, in a configuration that is resistant to failure of a link l, traffic routed in this configuration is never routed over this link, hence no traffic routed in this configuration is lost if l fails. In MRC, node n and link l are called isolated in a configuration, when, as described above, no traffic routed according to this configuration is routed through n or l.

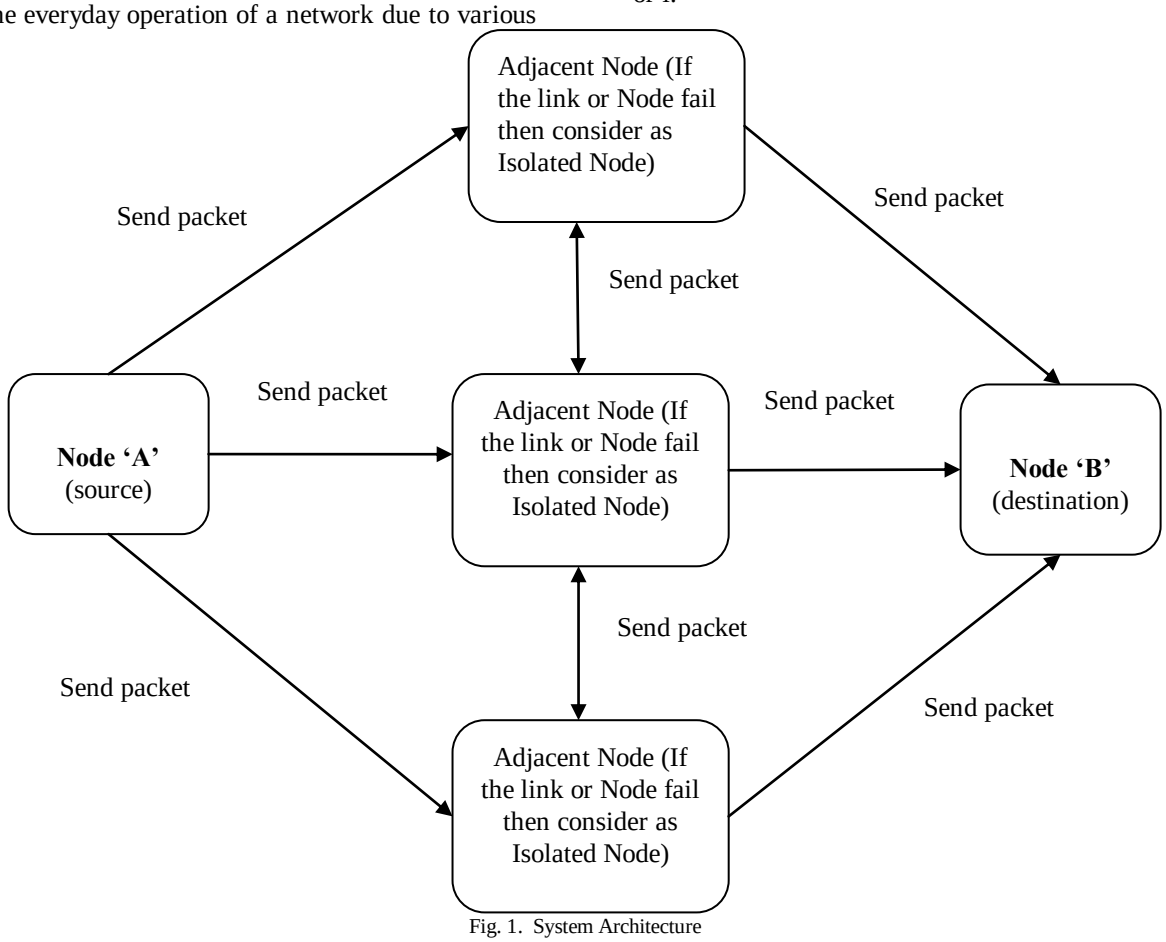

Fig. 1. System Architecture

MRC approach is threefold. First, create a set of backup configurations, so that every network component is isolated in one configuration. Second, for each configuration, a standard routing algorithm like OSPF is used to calculate configuration specific shortest path trees and create forwarding tables in each router, based on the configurations. The use of a standard routing algorithm guarantees loop free forwarding within one configuration. Forwarding process that takes advantage of the backup configurations to provide fast recovery from a component failure.

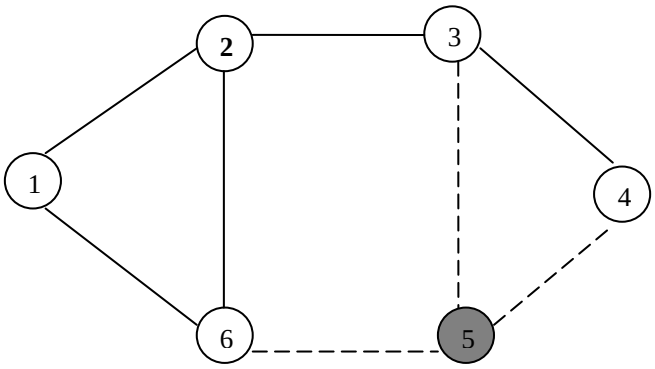

Fig. 1. Isolating a node.

Fig.2 illustrates a configuration where node 5 is isolated. In this configuration, the weight of the stapled links is set so high that only traffic sourced by or destined for node 5 will be routed over these links, which are restricted links. Node failures can be handled through blocking the node from transiting traffic. This node-blocking will normally also protect the attached links. But a link failure in the last hop of a path can obviously not be recovered by blocking the downstream node (ref. "the last hop problem"). Hence, I must make sure that, in one of the backup configurations, there exists a valid path to the last hop node, without using the failed link. A link is isolated by setting the weight to infinity, so that any other path would be selected before one including that link.

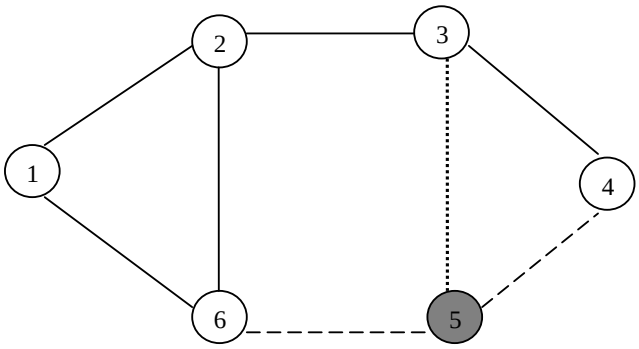

Fig. 3 Isolating links of a node.

Fig.3 shows the same configuration as before, except now link 3-5 has been isolated (dotted). No traffic is routed over the isolated link in this configuration; traffic to and from node 5 can only use the restricted links. In Fig.4, shows how several nodes and links can be isolated in the same configuration. In a backup configuration like this, packets will never be routed over the isolated (dotted) links, and only in the first or the last hop be routed over the restricted (dashed) links.

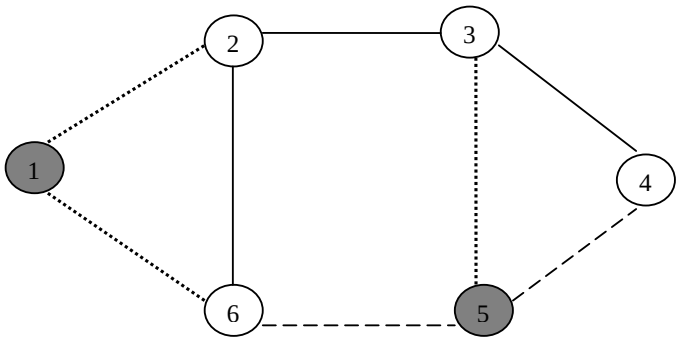

Fig.4 isolating more than one node

Some important properties of a backup configuration are worth pointing out. First, all non-isolated nodes are internally connected by a sub-graph that does not contain any isolated or restricted links. We denote this sub-graph as the backbone of the configuration. In the backup configuration shown in Fig.4, nodes 6, 2 and 3 with their connecting links constitute this backbone. Second, all links attached to an isolated node are either isolated or restricted, but an isolated node is

always directly connected to the backbone with at least one restricted link. Using a standard shortest path calculation, each router creates a set of configuration-specific forwarding tables. For simplicity, we say that a packet is forwarded according to a configuration, meaning that it is forwarded using the forwarding table calculated based on that configuration.

When a router detects that a neighbor can no longer be reached through one of its interfaces, it does not immediately inform the rest of the network about the connectivity failure. Instead, packets that would normally be forwarded over the failed interface are marked as belonging to a backup configuration, and forwarded on an alternative interface towards its destination. The packets must be marked with a configuration identifier, so the routers along the path know which configuration to use. Packet marking is most easily done by using the DSCP field in the IP header.

If this is not possible, other packet marking strategies like IPv6 extension headers or using a private address space and tunneling can be imagined. It is important to stress that MRC does not affect the failure free original routing, i.e. when there is no failure, all packets are forwarded according to the original configuration, where all link weights are normal. Upon detection of a failure, only traffic reaching the failure will switch configuration. All other traffic is forwarded according to the original configuration as normal.

A. Generating Backup Configurations

The algorithm will typically be run once at the initial startup of the network, and each time a node or link is permanently added or removed to give on the back up configurations used in MRC.

Configuration Constraints

To guarantee single-failure tolerance and consistent routing, the backup configurations used in MRC must adhere to the following requirements:

1. A node must not carry any transit traffic in the configuration where it is isolated. Still, traffic must be able to depart from and reach an isolated node.
2. A link must not carry any traffic at all in the configuration where it is isolated.
3. In each configuration, all node pairs must be connected by a path that does not pass through an isolated node or an isolated link.

Every node and every link must be isolated in at least one backup configuration.

The first requirement decides what weights must be put on the restricted links attached to an isolated node. To guarantee that no path will go through an isolated node, it suffices that the restricted links have a weight W of at least the sum of all link weights w in the original configuration

$W > \sum_{e_{i,j} \in \varepsilon} w_{i,j}$

It guarantees that any other path between two nodes in the network will be chosen by a shortest path algorithm before one passing through the isolated node. Only packets sourced by or destined for the isolated node itself will traverse a restricted link with weight W, as they have no shorter path. An algorithm, restricted and isolated links are given the same weight in both directions in the backup configurations, i.e., we treat them as undirected links. However, it does not prevent the use of independent link weights in each direction in the default configuration. The second requirement implies that the weight of an isolated link must be set so that traffic will never be routed over it. Such links are given infinite weight.

V. IMPLEMENTATION

In this paper, we consider four modules.

- Network construction
- Find Link Failure
- Calculate Load Balancing
- Find Isolated Node

A. Network construction

MRC configurations are defined by the network topology, which is the same in all configurations, and the associated link weights, which differ among configurations. We formally represent the network topology as a graph, with a set of nodes and a set of unidirectional links In order to guarantee single-fault tolerance, the topology graph must be bi-connected. A configuration is defined by this topology graph and the associated link weight function.

B. Find Link Failure

Send Packets through constructed network towards Destination Node. If sending node receive acknowledgement from Destination, means the link that forward packet is good. If not the link is considered as failure. This failure link is also called as isolated link.

C. Calculate Load Balancing

MRC offers functionality with a simpler and more intuitive approach, and leaves more room for optimization with respect to load balancing.

The backup configurations are constructed in a way that gives better load balancing and avoids congestion after a failure. We propose a procedure to do this by constructing a complete set of valid configurations in three phases. First, the link

weights in the normal configuration are optimized for the given demand matrix while only taking the failure free situation into account. Second, we take advantage of the load distribution in the failure free case to construct the MRC backup configurations in an intelligent manner. Finally, we optimize the link weights in the backbones of the backup configurations to get a good load distribution after any link failure.

D. Find Isolated Node

A node must not carry any transit traffic in the configuration where it is isolated. Still, traffic must be able to depart from and reach an isolated node. With MRC, restricted links are always attached to isolated nodes.

A restricted link connects the node before failure link and find alternate path, by searching this isolated node. Isolated node is the node which does not carry any traffic. With our algorithm, all nodes and links in the network are isolated in exactly one configuration.

The third property above results in the following two invariants for our algorithm, which must be evaluated each time a new node and its connected links are isolated in a configuration.

1) A configuration must contain a backbone
2) All isolated nodes in a configuration must be directly connected to the backbone through at least one restricted link.

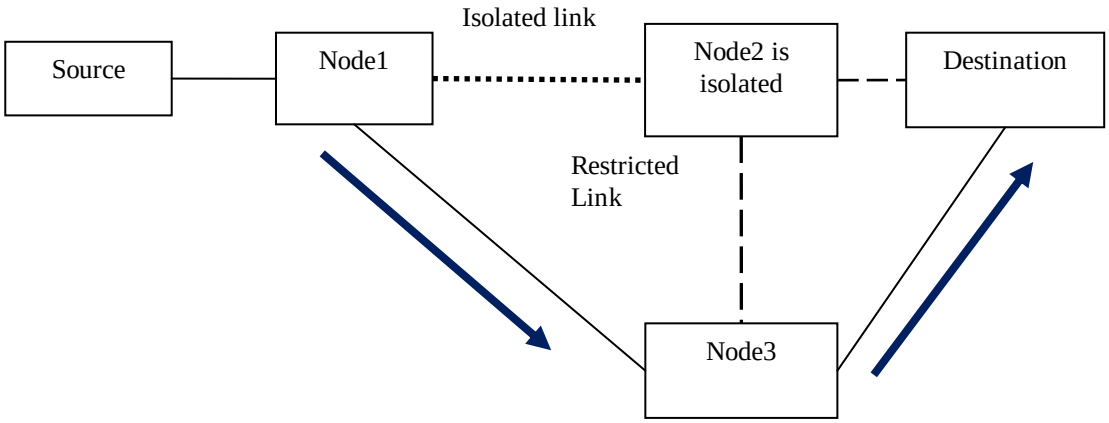

Fig.5 Finding isolate node

E. Performance Requirements

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

VI. RESULTS

MRC requires the routers to store additional routing configurations. The amount of state required in the routers is related to the number of such backup configurations. Since routing in a backup configuration is restricted, MRC will potentially give backup paths that are longer than the optimal paths. Longer backup paths will affect the total network load and also the end-to-end delay. It must be noted that MRC yields the shown performance immediately after a failure. The complexity of the proposed algorithm is determined by worst case O(|N|+|A|).Consider the following graph Fig.6

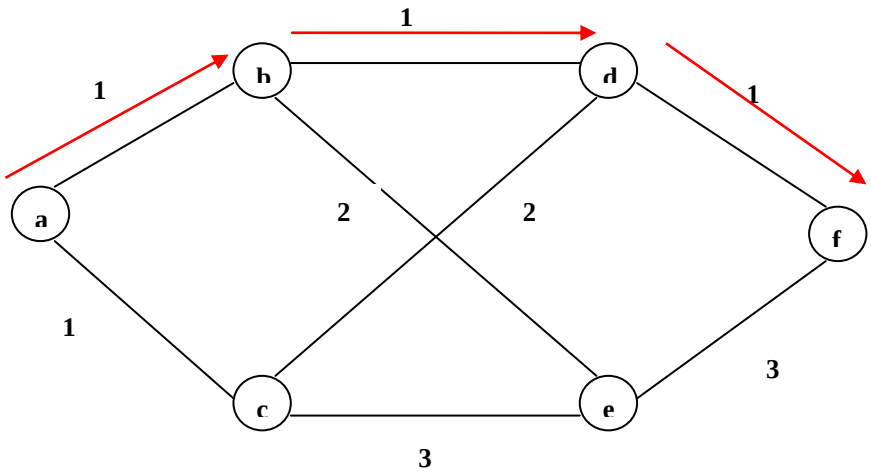

Fig.6.Configuration1

If data is send from source 'a' to destination 'f'. it choose the optimal path(a→b→d→f).Now I assume that the node 'd' is damaged, so isolate the node 'd' by isolating almost all its links and restricting at least one link, among the links to its immediate neighbor present in the alternate path to destination 'f' , After isolating the node 'd', the backup configuration obtained 'C_1' is shown below Fig.7,.

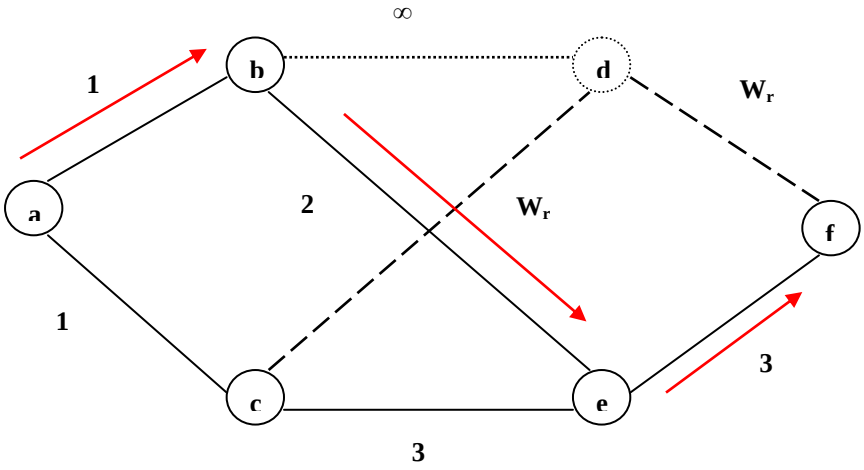

Fig.8.Configuration2

So the given complexity O(|N|+|A|) has been proved to be true when a single node isolation is considered (d) which as per the above graph is O(|1|+|3|). Now the data is send from source node 'a' to node 'f'. it choose alternate path a→b→e→f. After isolating the node 'b', the backup configuration obtained 'C2'shown below Fig.9, Now the alternate path a→c→e→f.

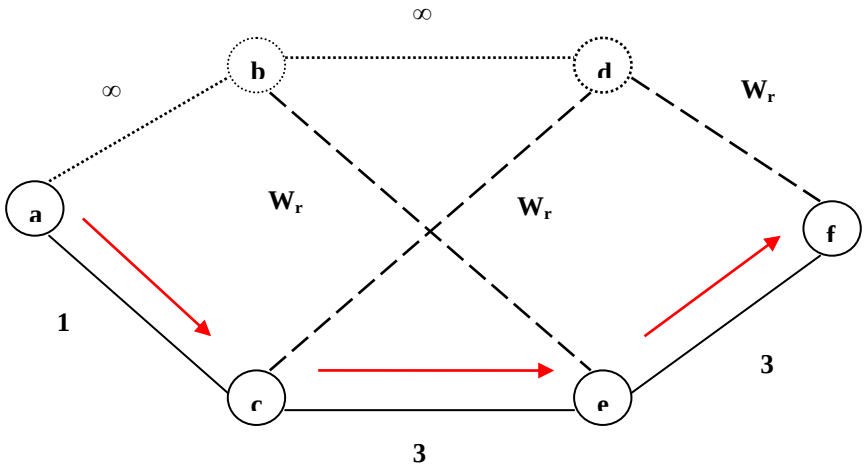

Fig.9.Configuration3

The complexity is also proved in the case of N=2(nodes isolated are ‘d’ and ’b’). Which as per the above graph is O(|2|+|5|). So the given computational complexity O(|N|+|A|) can be verified easily, when we consider all the nodes in the graph for isolation.

VII. CONCLUSIONS AND FUTURE WORK

Multiple Routing Configurations as an approach to achieve fast recovery in IP networks. MRC guarantees recovery from any single node or link failure in an arbitrary bi-connected network. By calculating backup configurations in advance, and operating based on locally available information only, MRC can act promptly after failure discovery. MRC operates without knowing the root cause of failure, i.e., whether the forwarding disruption is caused by a node or link failure. This is achieved by using careful link weight assignment according to the rules we have described. The link weight assignment rules also provide basis for specification of a forwarding procedure.

In this project, I focused how the network can be used to improve the distribution of the recovered traffic, and thus reduce the chances of congestion when MRC is used.

Future Work:

From the viewpoint of networking, there are still lots of open problems. Among them, the following questions will be studied in future:

- To reduce the risk of congestion after a failure by doing traffic engineering through intelligent link weight assignment in each configuration.
- To maintaining a separate multicast tree for each configuration to achieve very fast recovery from both link and node failures.

REFERENCES

[1] A. Basu and J. G. Riecke, “Stability Issues in OSPF Routing,” in Proceedings of SIGCOMM 2001, pp. 225–236, August 2001

[2] C. Boutremans, G. Iannaccone, and C. Diot, “Impact of link failures on VoIP performance,” in Proceedings of International Workshop on Network and Operating System Support for Digital Audio and Video, 2002.

[3] D. Watson, F. Jahanian, and C. Labovitz, “Experiences with monitoring OSPF on a regional service provider network,” in ICDCS ’03: Proceedings of the 23rd International Conference on Distributed Computing Systems. IEEE Computer Society, pp. 204–213, 2003.

[4] S. Lee, Y. Yu, S. Nelakuditi, Z.-L. Zhang, and C.-N. Chuah, “Proactive vs. reactive approaches to failure resilient routing,” in *Proceedings IEEE INFOCOM’04*, Mar. 2004.

[5] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, and C. Diot, “Characterization of failures in an IP backbone network,” in Proceedings of INFOCOM 2004, Mar. 2004.

[6] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, “An approach to alleviate link overload as observed on an IP backbone,” in Proceedings of INFOCOM’03, pp. 406–416, Mar. 2003.

[7] P. Psenak, S. Mirtorabi, A. Roy, L. Nguen, and P. Pillay-Esnault, “MTOSPF:Multi topology (MT) routing in OSPF,” IETF Internet Draft, Apr. 2005.

[8] T. Przygienda, N. Shen, and N. Sheth, “M-ISIS: Multi topology (MT) routing in IS-IS,” Internet Draft , May 2005.

[9] M. Menth and R. Martin, “Network resilience through multi-topology routing,” University of Wurzburg, Institute of Computer Science, Tech. Rep. 335, May 2004.

[10] I. Theiss and O. Lysne, “FROOTS - fault handling in up*/down* routed networks with multiple roots,” in Proceedings of the International Conference on High Performance Computing, 2003

[11] A. Kvalbein, A. F. Hansen, T. Cicic, S. Gjessing, and O. Lysne, “Fast recovery from link failures using resilient routing layers,” in Proceeding 10th IEEE Symposium on Computers and Communications (ISCC), June 2005.

[12] P. Pan, G. Swallow, and A. Atlas, “Fast reroute extensions to RSVP-TE for LSP tunnels,” RFC 4090, May 2005.

[13] P. Narvaez, K.-Y. Siu, and H.-Y. Tzeng, “Local restoration algorithms for link-state routing protocols,” in Proc. IEEE Int. Conf. ComputerCommunications and Networks (ICCCN’99), pp. 352–357, Oct. 1999.

[14] R. Rabbat and K.-Y. Siu, “Restoration methods for traffic engineered networks for loop-free routing guarantees,” in Proc. IEEE Int. Conf.Communications (ICC’01), Helsinki, Finland, vol. 5, pp. 1566–1570, Jun. 2001.

[15] Z. Zhong, S. Nelakuditi, Y. Yu, S. Lee, J. Wang, and C.-N. Chuah, “Failure inferencing based fast rerouting for handling transient link and node failures,” in Proc. IEEE INFOCOM, vol. 4, pp. 2859–2863, Mar. 2005.

[16] I. Theiss and O. Lysne, “FRoots, a fault tolerant and topology agnostic routing technique,” IEEE Trans. Parallel Distrib. *Syst.*, vol. 17,pp. 1136–1150, Oct. 2006.

[17] A. F. Hansen, T. Ci cic, S. Gjessing, A. Kvalbein, and O. Lysne, “Resilient routing layers for recovery in packet networks,” in Proc. Int.Conf. Dependable Systems and Networks (DSN 2005), pp. 238–247, Jun. 2005.

[18] B. Fortz and M. Thorup, “Internet traffic engineering by optimizing OSPF weights,” in Proc. IEEE INFOCOM, pp. 519–528, 2000.

[19] A. Sridharan and R. Guerin, “Making IGP routing robust to link failures,”in Proc. Networking, Waterloo, Canada, 2005.