The Impact of Topology on Overlay Routing Service

Zhi Li and Prasant Mohapatra
{lizhi,prasant}@cs.ucdavis.edu
Department of Computer Science
University of California, Davis
Davis, CA 95616, USA

***Abstract*—**
A moderate amount of recent work has been dedicated to using overlay network to support value-added network service, such as overlay multicast, OverQoS, etc. As it does not require the underlying network support, a lot of new services can be easily deployed across Internet using overlay technique . Overlay service network is a generic service framework which is designed to provide a variety of services to overlay service customers.

To design an overlay service network, the first step is to choose an overlay topology connecting all the overlay service nodes. For example, RON [6] has used full mesh topology connecting all the nodes. When considering the overlay topology, several questions were left unconsidered: 1) Are there any other topologies which also can provide us satisfactory performance? 2) How the overlay topologies affect the overlay routing performance? 3) Do they have some direct connections with each other?

In this paper, we did a study on the impact of topology on the overlay routing service. We found that the overlay topology has significant impact on the overlay routing in terms of routing performance and routing overhead. For example, the full mesh topology does not always give us the best performance. Moreover, the physical topology information can benefit us a lot to construct an efficient overlay topology. In addition, some alternative topologies can provide us with better performance when considering both the routing performance and overhead.

***Index Terms*— Overlay Service Network, Overlay Routing, Overlay Topology**

I. Introduction

Overlay technique is an effective way to support new applications as well as protocols without any changes in the underlying network layer. For example, Qbone[4] and Mbone[13] utilize overlay technique to support QoS and multicast services, respectively, on top of the current Internet infrastructure.

An overlay network is formed by a subset of underlying physical nodes. The connections between the overlay nodes are provided by overlay links (IP-layer paths), each of which is usually composed of one or more physical links. As the overlay applications are usually built at the application layer, it can effectively use the Internet as a lower level infrastructure to provide higher level services to end users. Several recent applications have utilized overlay network to provide value-added Internet service, such as Peer-to-Peer file sharing[1], overlay multicasting[10], [9], [15], [29], etc.

This research was supported in part by the National Science Foundation through the grants CCR-0296070 and ANI-0296034.

Several different overlay service networks have been proposed, such as Resilient Overlay Network (RON)[6], Service Overlay Network (SON)[12], QoS-aware routing for Overlay Networks (QRON)[17], OverQoS[27]. An overlay service network (OSN) is usually composed of a set of fixed overlay nodes, which are strategically placed by a third party. The third party purchases access service for the overlay nodes from different Internet Service Providers (ISPs). These overlay nodes cooperate with each other to provide an overlay service platform, on top of which a variety of application-specific overlay can be constructed, such as multicast overlay, anycast overlay, end-to-end QoS overlay, etc.

As overlay links may share the same physical links with other overlay links or Internet applications, the overlay service nodes usually cannot directly control the underlying physical link resource. The up-to-date overlay link performance information can only be obtained through the measurement methods. As the number of overlay links increases, the probing overhead will increase dramatically.

Even though a moderate amount of research have done on overlay service network, no work has been dedicated to the overlay service network topology issues. As the overlay nodes are connected via IP-layer paths (overlay links), theoretically, there is an overlay link connecting each pair of overlay nodes. Different selections of overlay links (topologies) would affect the overlay service quality and cost.

To connect all the overlay nodes, we have many candidate topologies. For example, we can use minimum spanning tree to connect all the overlay nodes. Alternatively, we also can leverage the underlying topology information to construct the overlay topology. Or, we can just use full mesh to connect all the overlay nodes. How will these different topologies affect the service performance of overlay service network? Does the performance of overlay routing protocols differ very much in different overlay topologies? Is there any scalable overlay topology to substitute full mesh topology?

As we metioned above, overlay service network is a service platform, on top of which a lot of application-specific overlays can be built up. As an indispensable module of overlay service network, overlay routing can provide data delivery service to all the application-specific overlays. Resilient overlay routing service not only can be seen as an application-specific overlay built on top of overlay service network, but also an important module of overlay service network, on top of which a variety of application-specific overlays can be set up. In this paper, we use resilient overlay routing service as an example to evaluate the different overlay topologies' impact on overlay service network. Resilient overlay routing service is based on the same underlying theory as RON[6]. RON is designed to utilize the Internet path redundancy properties to provide resilient overlay path when normal Internet paths get outages. Thus, we can provide resilient overlay routing support to a variety of applications. To achieve this goal, it uses the full-mesh topology connecting all the overlay nodes. To get the up-to-date information of the overlay paths performance and achieve short failover time, each overlay node continuously monitors the overlay links connecting to itself and sends this information to all the other overlay nodes. Because the probing and routing message overhead, the authors have pointed out that the RON architecture is not scalable over 50 overlay nodes[6].

Resilient overlay routing service works as follows. By default, an overlay user always sends the traffic to destination via the default IP-layer path. When it realizes that a path is faulty (such as long end-to-end delay, lower bandwidth), it will forward the data to its nearest overlay node. Then, the overlay node will forward the data to destination overlay node via an overlay path. During the study, we vary the overlay protocols and overlay topologies to study the performance impact due to the overlay topologies.

To provide resilient overlay routing service, we consider several different overlay routing protocols, such as link-state based source routing[6], or feedback based routing[32].

In this paper, we first present several overlay service network topologies, such as K-spanning tree, mesh-tree, adjacent connection, typology-aware K-spanning tree. Then, we present a series of simulation study to evaluate the performance of the two routing protocols on top of different overlay topologies. We did the simulation on top of two physical topologies, one random topology generated by GT-ITM[2] and Sprint ISP topology published in [24]. We evaluate the performance in terms of failure recovery ratio, routing overhead, overlay path penalty, etc. The simulation results have shown that the topology has significant impact on the performance of the overlay routing services. A routing protocol's performance differs a lot when using different overlay topologies. A routing protocol may work well on a topology while another protocol may can not achieve good performance on the same topology. Thus, we need to consider both overlay routing protocol and overlay topology when designing an overlay service network.

The rest of the paper is organized as follows. In Section II, we introduce the resilient overlay service network architecture. The candidate overlay service network topologies are presented in Section III. In Section IV, we introduce two different overlay routing protocols. The simulation setup and evaluation performance metrics are discussed in Section V. We present and analyze the simulation results in section VI. The related works are discussed in Section VII. Finally, we draw the conclusions and present the future work in Section VIII.

II. Resilient Overlay Service Network

Overlay Service Network is composed of a number of specialized overlay nodes that are placed in the Internet by a third party (termed as Overlay Service Provider (OSP)) to provide generic overlay service support to a variety of applications. Overlay nodes can be placed either at the edge of a domain or in the core. The OSP subscribes high bandwidth connections for these overlay nodes from the Internet backbone. Fig. 1 shows an example of overlay service network topology, which is composed of 5 overlay nodes from 3 ASes. The overlay nodes are connected via overlay links, each of which is an IP-layer path connecting an overlay node pair. In this example, the overlay nodes are connected together based on a random topology. From the figure, we can see that some of the overlay links are overlapped at physical layer even though they are completely disjoint at overlay service layer. This is one of the special characteristics of overlay networks. In addition, we can see that each of the overlay links is usually composed of several physical links. When considering an overlay link, other non-overlay traffic or other overlay links may pass the same physical link. This means that the overlay links' capacity is not fixed and cannot be controlled by the overlay nodes. To provide satisfactory service to overlay users, the overlay nodes need to continuously probe each other to obtain the latest performance of the overlay links.

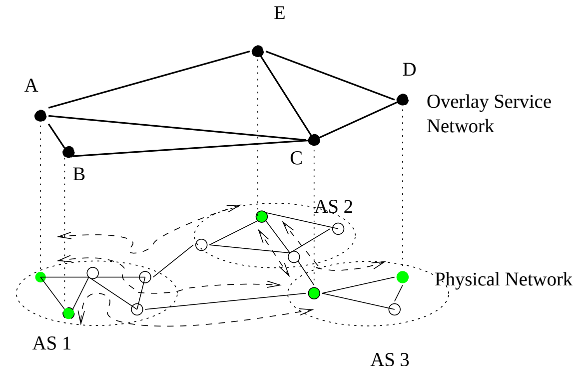

Fig. 1. Overlay Service Network

RON has proved that the overlay network can be used to provide users better overlay paths when the default Internet paths get fault. Based on the same inference, we can use resilient overlay service network to provide resilient routing service to large group of Internet users. The resilient routing service works as the following steps. 1)By default, the users always use the IP-layer routing service to send data traffic to destinations. 2)When a user realizes that there is a service outage (such as long end-to-end delay, or low throughput) in the default IP-layer path, it will send the subsequent data traffic to its nearest overlay node (source overlay node). 3)The source overlay node will forward the traffic to the destination overlay node via a fault-free overlay path. 4)The destination overlay node will forward the data to the traffic destination.

In this paper, we will not get into the detail architecture of resilient overlay service network. We use it as an example to evaluate the overlay routing protocol's performance on top of different overlay topologies.

III. Overlay Service Network Topologies

To connect all the overlay nodes to form an overlay service network, there are many possible topologies we can adopt. In this section, we propose and list several topologies that have appeared in literatures. Each of them could be a viable candidate topology for overlay service network.

A. Full-Mesh (FM)

As the overlay service network runs on top of IP layer, there is an IP-layer path connecting each pair of the overlay nodes. Thus, each pair of overlay nodes could be neighbors with each other at overlay layer. Based on this notion, all the overlay nodes can form a full-mesh topology. As we mentioned above, the overlay nodes cannot directly control and retrieve the overlay link resource information because the unexpected bypass non-overlay traffic may go through same physical links. To get the overlay link performance, such as latency, bandwidth, the overlay nodes need to continuously send the probing packets to neighboring overlay nodes. RON uses link-state based routing protocol, in which each node sends its local link state to every other overlay nodes. It was shown that for an overlay network composed of 50 nodes, each node will have around 33Kbps routing overhead[6]. Fig. 2 is an example of a full

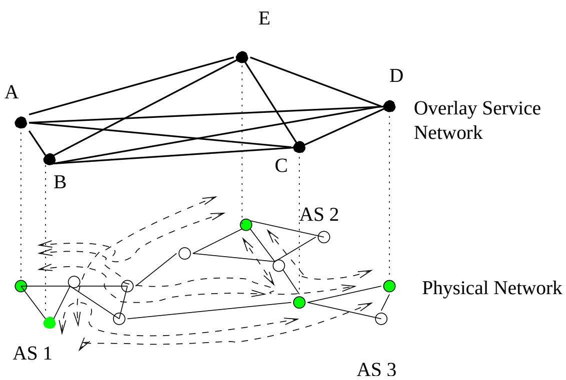

Fig. 2. Full Mesh Overlay topology.

mesh overlay topology. From the figure, we can observe that each pair of overlay nodes are neighbors with each other in the overlay topology. A lot of overlay links pass through the same common underlying physical links.

B. K-Minimum-Spanning-Tree (KMST)

A minimum spanning tree is a tree without any loops which connects all the nodes with the lowest cost among all the candidate trees. To minimize the state maintenace overhead, we propose an overlay service network topology which is composed of K minimal disjoint minimum spanning trees in the full mesh toplogies. The K trees have the minimal overlaps of overlay links and compose an overlay service network topology. Here, the cost of a overlay link is defined as the number of physical hops the overlay link passes through. We can take different value of K based on the different cost-performance tradeoff. Fig. 3 shows a 2-minimum spanning tree overlay topology, in which the dashed lines and solid lines belong to different spanning trees, which together compose the overlay topology.

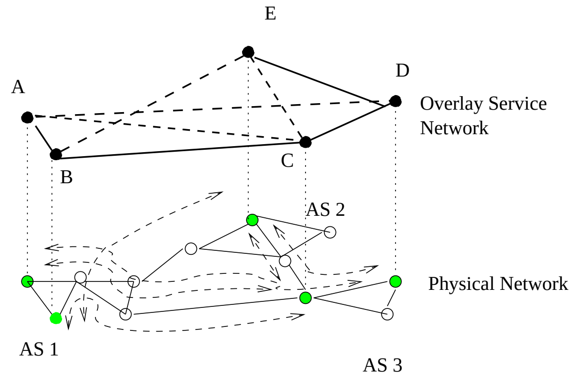

Fig. 3. 2-minimum Spanning Tree Overlay topology.

C. Mesh-Tree (MT)

We proposed a Mesh-Tree topology in HostCast[18] to enhance the resilience of the overlay multicast. The Mesh-Tree topology can also be modified as an overlay service network topology. A Mesh-Tree topology can be constructed as follows: 1) Set up a minimum spanning tree connecting all the overlay nodes; 2) If two overlay nodes have grandchild-grandparent or uncle-nephew relationship in the minimum spanning tree, there is also an overlay link connecting these two overlay nodes. Fig.

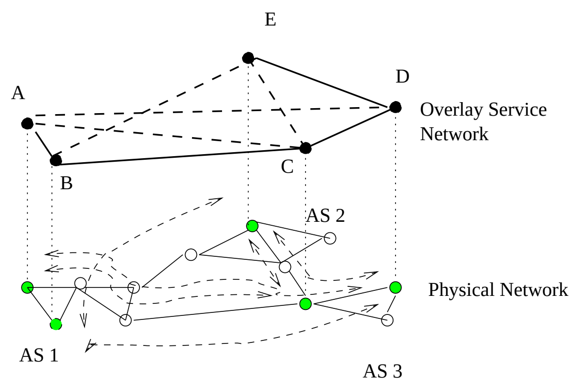

Fig. 4. Mesh-Tree Overlay topology.

4 shows an example of MT topology, in which the solid lines compose a spanning tree and the dash lines are the added mesh links.

D. Adjacent-Connection (AC)

Adjacent connection uses the knowledge of physical topology for the overlay topology construction. In this approach, we assume that we have the physical topology information connecting all the overlay nodes. By default, the Internet usually uses the shortest path based routing. Thus, the overlay topology construction method can be formalized as: if there is no other overlay node directly connected to one of the nodes on the IP-layer shorted path between two overlay nodes, there is an overlay link between these two overlay nodes. In [19] and [17], the authors use this method to construct overlay service topology.

Fig. 5 shows an example of the AC topology. In this example, no overlay node is on the physical path of any overlay links.

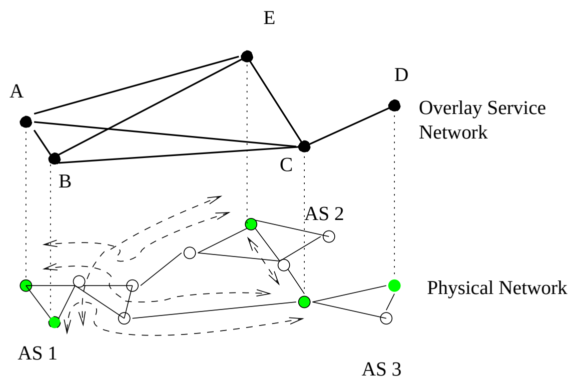

Fig. 5. Adjacent Connection Overlay topology.

E. Topology-aware-K-Minimum-Spanning Tree (TKMST)

The construction of TKMST is also constrained by the underlying physical topology. This topology uses K minimal spanning trees connecting all the overlay nodes. However, when considering the disjoint property of two overlay links, we not only consider the overlay layer but also the IP layer. If two overlay links pass through a common physical link, we also deem it as overlapped. Thus, the resulting K spanning trees have the least overlap at the physical links. Using this method, the overlay network can provide each source-destination pair diverse physically disjoint overlay paths.

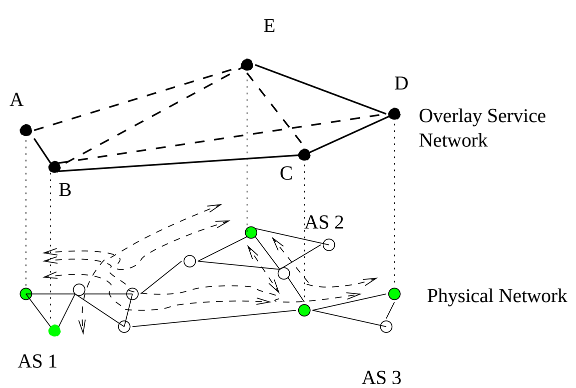

Fig. 6. Topology-aware 2-minimum Spanning Tree Overlay topology.

Fig. 6 shows an example of topology-aware 2-minimum spanning tree overlay topology, in which the dash lines and solid links belong to two least disjoint spanning trees.

F. Summary of the Overlay Topologies

	Degree	Link number	Resilience	Distortaion
FM	n-1	n(n-1)/2	M	Low
KMST	K	<=K(n-1)	<min(M,K)	High
MT	(>2)&(<n-1)	<3(n-1)	>1	High
TKMST	K	<=K(n-1)	<min(M,K)	High
AC	<n-1	<n(n-1)/2	M	Low

TABLE I

COMPARISION OF VARIOUS OVERLAY TOPOLOGIES

The characteristics of the overlay topologies are summarized in Table 1. Node degree determines the overlay topology neighbor states each node needs to maintain. M is the maximal disjoint paths the physical topology can provide for each pair of overlay nodes. Resilience is defined as the average number of physical disjoint paths the overlay topology can provide for each pair of ovelay nodes. The exact number is also determined by the underlying physical topology. The distortion is defined as the overlay path length penalty if the two nodes cannot be directly connected via the IP-layer path.

IV. ROUTING PROTOCOLS IN OVERLAY SERVICE NETWORK

The routing protocols in the overlay service network can provide alternative overlay paths to deliver users' traffic when faults occur in the default Internet paths. To provide satisfactory service, the overlay nodes should have the up-to-date overlay path performance information.

A. Link-state Based on-demand Approach

Link-state routing is one of the most popular routing protocols. It is used in RON. To provide the best overlay paths with less failover time to users, it assumes that each overlay node knows the global topology information as well as the up-to-date overlay link performance information. Thus, the overlay service network can quickly find an overlay path connecting to the destination overlay node when there is a fault in the default Internet path. The link-state based on-demand approach requires that each overlay node to continuously monitor the performance of the overlay links that connect it to its neighboring overlay nodes. In addition, each node also needs to send the measurement result to all the other overlay nodes. As a result, the routing overhead not only includes the measurement probing traffic but also sending and receiving the link-state message traffic. Suppose the overlay topology has n nodes and average node degree is d, the overall routing overhead is n*d*(number of probing messages) plus n*(n-1)*(number of link state messages).

B. Feedback Based Proactive approach

Feedback based routing was proposed in [32]. It was also used in [30] to provide resilient routing service. We generalize the feedback based proactive routing protocol as follows. In this approach, it assumes that the overlay topology is relatively fixed. Each overlay node maintains two backup overlay paths to every other overlay nodes, which are disjoint with each other in the overlay topology. Then, it continuously monitors the overlay paths' performance. When the default Internet path incurs a fault, it will pick one of the backup paths with the best performance to bypass the failure point. As the backup paths are disjoint with each other at overlay layer, the chance of failure in all the paths is low. Thus, the overlay network has a high probability of bypassing the path faults. Note that the generalized routing protocol is not physical topology-aware based. The two backup overlay paths still have the possibility of sharing some physical links with the default IP-layer path.

This method differs from the link-state based routing in the following aspects: 1) The performance monitoring is path-based instead of link-based measurement. As a result, the measurement results not only reflect the overlay links' performance but also the overlay nodes' performance, such as processing speed. 2) The overlay nodes only need to maintain the overlay path performance to each overlay node instead of all the overlay links' performance. As a result, the routing overhead only involves sending and receiving of probing messages. Suppose the overlay topology has n nodes and average path length is h, the overall overhead is n*(n-1)*h*(the number of probing messages)*(num of backup paths).

V. Network Model & Simulation Setup

The simulations are based on two underlying physical topologies: a real ISP topology (an intra-AS topology) and a random topology (which can represent some properties of inter-AS topology). The ISP topology we used is the Sprint POP level topology published by Rocketfuel[24], which is composed of 44 nodes and 212 pop level links The random topology is Waxman topology[28] generated by GT-ITM[2], which is composed of 100 nodes and connected by 354 links.

GT-ITM uses the following approach to generate a sample topology. Network nodes are randomly chosen in a square ($\alpha * \alpha$) grid. A link exists between the nodes u and v with the probability $P(u, v) = a * e^{-d(u,v)/(b*\alpha^2)}$, where d(u,v) is geometric distance between u and v, a and b are constants that are less than 1. In the simulation, we take $a = 0.03$ and $\alpha = 100$. All the physical link delays are uniformly set as 2ms.

For each of the overlay nodes, we randomly attach it to one of the physical nodes. During the simulation, we vary the number of overlay nodes and percentage of failure links to get simulation results. In real overlay service network, the link failure means that either some physical links are really broken or the links do not meet the overlay service users' QoS requirement, such as delay, bandwidth, etc.

For each simulation, the overlay nodes are connected via one of the following overlay topologies:

1) Full Mesh (FM).
2) Mesh-Tree (MT).
3) Minimum Spanning Tree (MST).
4) Two Minimum Spanning Tree (2-MST).
5) Adjacent Connection (AC).
6) Topology-aware Two Minimum Spanning Tree (T2MST).

During the simulation, we use PRIM algorithm [11] to construct the MST. The goal is to minimize the sum of physical hops connecting all the overlay nodes.

For the overlay routing algorithm, we simulated the two previously listed approaches: link-state on-demand and feedback based proactive approach. For the proactive routing approach, each overlay node maintains two overlay paths to each other overlay node. For IP-layer, we assume it always uses the shortest path based routing protocol: link-state routing protocol.

In the simulation, we focus on the following performance metrics:

1) **Failure Recovery Ratio**
For resilient routing service overlay, it should be able to forward the data traffic via the overlay path to the destinations when the default IP-layer path fails or gets service degradation. The failure recovery ratio is an important metric to evalute the overlay network's service performance. During simulation, when a physical link fails, if the IP-layer path connecting the two overlay nodes fails, the source overlay node will try to use the overlay to find an overlay path connecting to the destination overlay node. How does the source overlay node locate the destination overlay node is out of the scope of this paper. The Failure Recovery Ratio is used to evaluate the OSN's performance in failure recovery. It is defined as follows.
Failure Recovery Ratio =
$$\frac{Num.\ of\ overlay\ paths\ of\ recovered}{Num.\ of\ IP\!-\!layer\ faulty\ paths}$$
2) **Recovery Path Hop Penalty**
We assume that the IP-layer always takes the shortest paths connecting the source and destination pairs, which means that the recovered overlay paths usually will take longer physical paths. The longer physical path could mean that the path has longer latency or consume more network resources. In reality, the IP-layer inter-AS path is determined by each AS's routing policies, which may result in non-shortest path. In this case, the overlay recovery path may result in shorter path than IP-layer path. We use recovery path hop penalty to quantify the overlay paths' physical distance compared to original IP-layer path.
Recovery Path Hop Penalty =
$$\frac{Length\ of\ the\ recovered\ overlay\ path\ (num\ of\ physical\ hops)}{Length\ of\ the\ original\ IP\!-\!layer\ path\ (num\ of\ physical\ hops)}$$
3) **Routing Overhead**
As the overlay network cannot control and reserve the underlying network resource, to infer up-to-date overlay link performance information, the overlay nodes need to continuously send probing messages to neighboring overlay nodes. For the link-state based routing, each node also needs to broadcast the up-to-date link state information to all the other overlay nodes. During the simulation, we use the same probing interval and routing interval as in [6], 12 seconds and 14 seconds, which can achieve average fault detection time of 19 seconds. We evaluate the routing overhead by comparing the average probing and routing traffic overhead each overlay node receives per second.
4) **Average Node Degree**
In overlay network, the node degree determines the number of probing traffic the node will receive from neighboring overlay nodes as well as the number of possible overlay paths connecting this node to other overlay nodes. After comparing the previous performance metrics, we will compare each topology's average node degree and try to infer the relationship between the performance and average node degree.

VI. Simulation Result

During the simulation, we vary the following variables: num of overlay nodes, physical link failure ratio, overlay routing

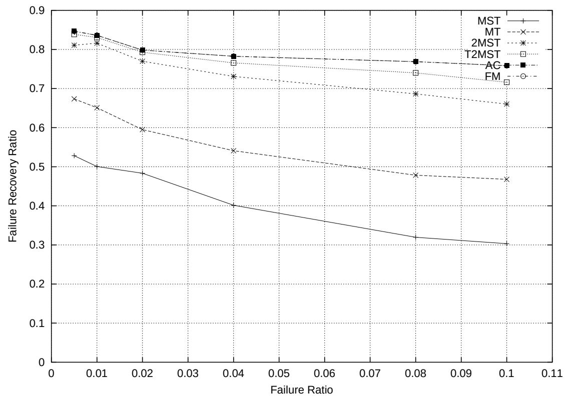

Fig. 7. Failure Recovery Ratio vs. Failure Ratio (Sprint topology, Link-state Routing, 30 overlay nodes).

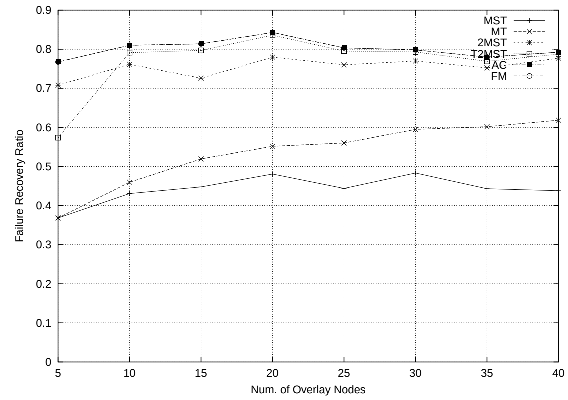

Fig. 9. Failure Recovery Ratio vs. Num. of Overlay Nodes (Sprint topology, Link-state Routing, failure ratio=0.02).

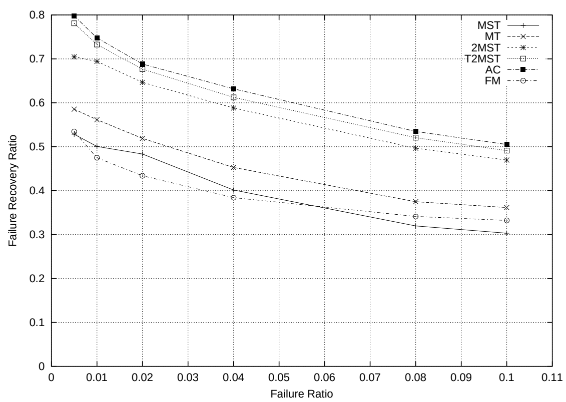

Fig. 8. Failure Recovery Ratio vs. Failure Ratio (Sprint topology, Feedback based Routing, 30 overlay nodes).

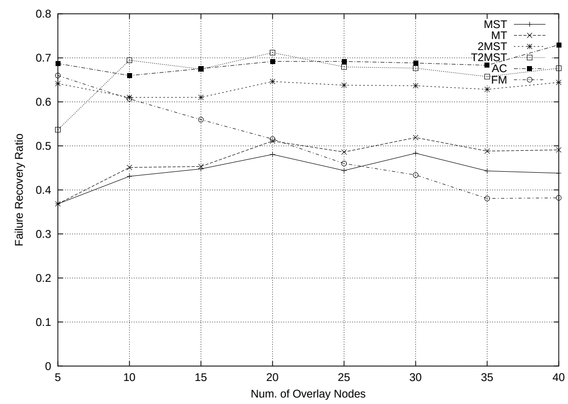

Fig. 10. Failure Recovery Ratio vs. Num. of Overlay Nodes (Sprinttopology, Feedback based Routing, failure ratio=0.02).

protocol, overlay topology. The physical link failure ratio is defined as the number of concurrent failure links. The link failure ratio is selected between 0 and 0.1. For the sprint topology, the number of overlay nodes is set as: 5, 10, 15, 20, 25, 30, 35 and 40, respectively. For the random topology, the number of overlay nodes is selected from the following: 10, 20, 30, 40, 50, 60, 70, 80 and 90. For each different simulation setup, we the run the simulator for 2000 times and get the average value for each performance metric. The simulation results for the Sprint topology are shown as follows. From the simulation results, we have got, for almost all the performance metrics, each overlay topology gives us the same performance trend on the two physical topologies. The results for the random topology are shown in the appendix.

A. Failure Recovery Ratio

Fig. 7 shows the failure recovery ratio of a 30 nodes overlay network using link-state on-demand routing protocol on top of the Sprint topology. From the simulation results, we can see that the failure recovery ratio decreases as the failure ratio increases. Among the topologies, the minimum spanning tree's recovery ratio drops about 20% when the failure ratio increases from 0.005 to 0.1 while the full mesh and adjacent connection topology more or less keep stable.

When comparing the topologies, the full mesh and adjacent connection topologies have the same recovery performance, which means that some of the overlay links in the full mesh can be pruned without scarifying the performance. Both full-mesh and Adjacent connection topologies can achieve better performance than topology-aware 2 minimum spanning tree, 2 spanning tree, mesh tree and minimum spanning tree. Another result we can observe from the figure is that topology-aware 2 spanning tree can achieve almost similar performance as full mesh and adjacent connection topology.

Fig. 8 shows the failure recovery ratio of a 30 nodes overlay network using feedback based proactive routing protocol on top of Sprint topology. In contrast to the link-state based routing, we can see that the full-mesh topology's performance is similar to minimum spanning tree, which is the worst among all the candidate overlay topologies. This is because the two shortest disjoint overlay paths at overlay layer may not be disjoint at physical layer as well as with default IP-layer path in full mesh topology. Thus, they cannot provide the desirable resilient overlay service. The comparisons among the rest of the topologies are the same as in Fig. 7.

Fig. 9 and Fig. 10 show the failure recovery ratio of the two routing protocols on all the candidate overlay topologies with fixed failure 0.02 and various sizes of overlay network. From the figures, we can see that the performance of all the candidate topologies maintains steady except full-mesh topology in feedback based proactive routing case, in which, the failure recovery ratio drops with the increase in the number of overlay

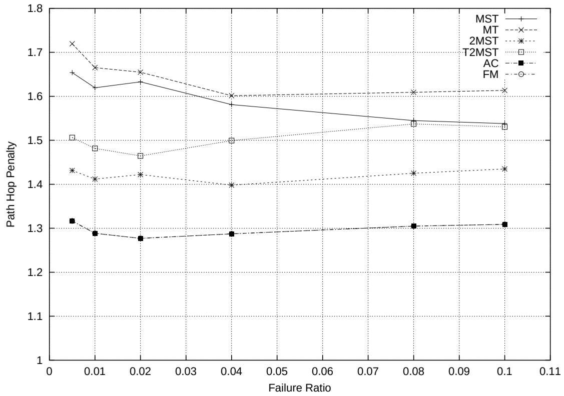

Fig. 11. Recovery Path Hop Penalty vs. Failure Ratio (Sprint topology, Link-state Routing, 30 overlay nodes).

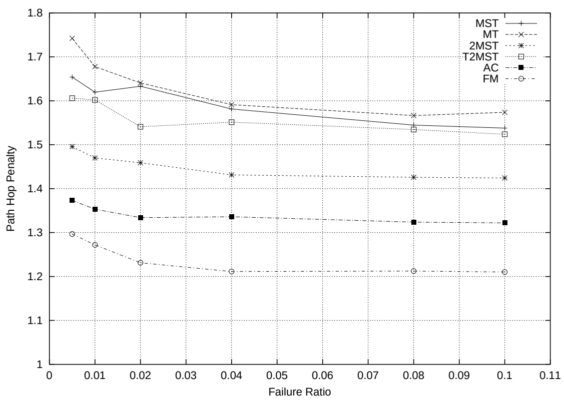

Fig. 12. Recovery Path Hop Penalty vs. Failure Ratio (Sprint topology, Feed-back based Routing, 30 overlay nodes).

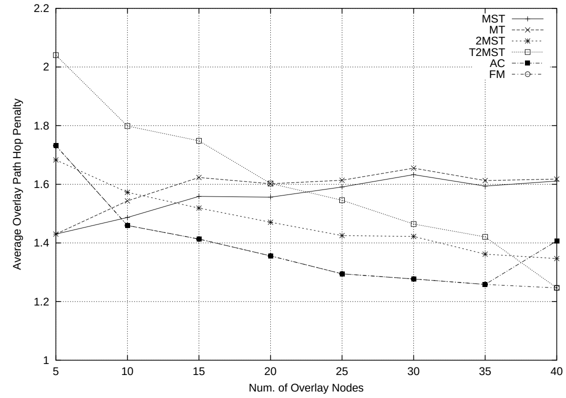

Fig. 13. Recovery Path Hop Penalty vs. Num. of Overlay Nodes (Sprint topology, Link-state Routing, failure ratio=0.02).

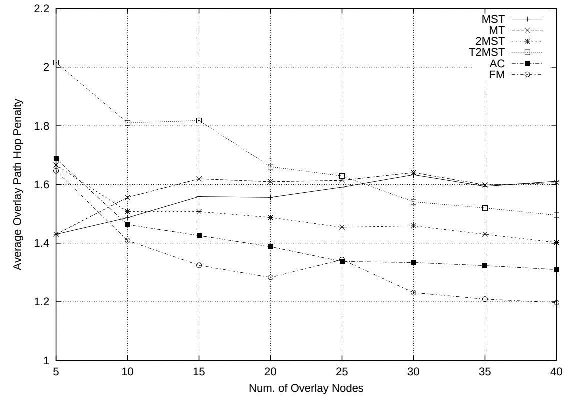

Fig. 14. Recovery Path Hop Penalty vs. Num. of Overlay Nodes (Sprint topology, Feedback based Routing, failure ratio=0.02).

nodes. Because with the number of nodes increase, the more IP-layer shortest paths will be affected failure. For full-mesh topology, even though it has lots of overlay links, it always tries to choose the two shortest overlay paths without considering underlying physical topology. As a result, the backup overlay paths will have higher chance to overlap with the original IP-layer shortest paths at physical links, which result in average lower falure recovery ratio. When considering other topologies, the increase in overlay network size also cannot increase failure recovery ratio very much. However, as we know, larger size overlay network can benefit more overlay service customers.

B. *Overlay Path Hop Penalty*

Fig. 11 shows the overlay path hop penalty of the link-state based on-demand routing on the different candidate overlay topologies. From the result, we can observe that the full-mesh topology and the adjacent connection topology have the least overlay path hop penalty, which is around 1.3. Following these two, the topology-aware 2 minimum spanning tree topology has lower penalty than all the other topologies. This is because link-based on-demand routing is source-based approach. The more links the overlay topology has, the higher chance that it will use shorter overlay paths to pass around the failure point. Another trend we can observe from the result is that the path hop penalty drops as the physical link failure ratio increases in 2 minimum spanning tree and mesh tree topologies, while the path hop penalty increases in the other topologies.

Fig. 12 shows the overlay path hop penalty of the feed-back based proactive routing approach on top of the overlay topologies. Similarly, the full-mesh topology also gives us the least hop penalty. This is because full-mesh topology can provide us with the maximal number of candidate backup overlay paths connecting a pair of overlay nodes and always selects the shortest two. As a result, the physical hop distances of the two backup overlay paths must be equal or less than the two backup overlay paths in any other topologies. The comparison among the other topologies is same as in the link-state routing case. The penalty difference between the largest and smallest is around 0.4.

When fixing the link failure ratio as 0.02 and varying the size of the overlay networks, the simulation results in overlay path hop penalty are depicted in Fig. 13 and Fig. 14. The variation in the size of overlay network doesnot affect the performance comparison among the candidate overlay topologies. From the results in these two figures, we can see that except the minimum spanning tree and mesh-tree, the overlay path hop penalty decreases with the increase in the size of the overlay network for all the other topologies. However, minimum spanning tree and mesh-tree give us opposite results.

C. *Routing Overhead*

Routing overhead is a major concern of overlay service network. When using link state based on-demand routing protocol,

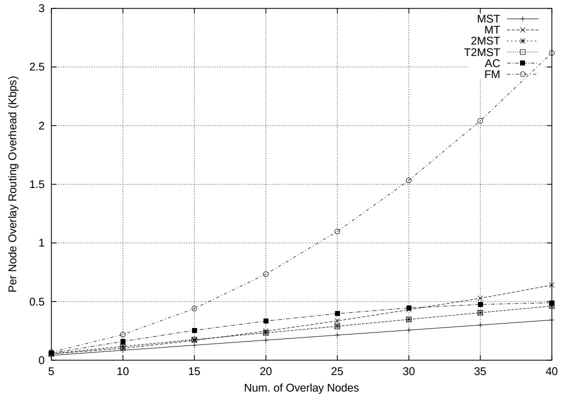

Fig. 15. Routing Overhead vs. Num. of Overlay Nodes. (Sprint topology, Link-state Routing).

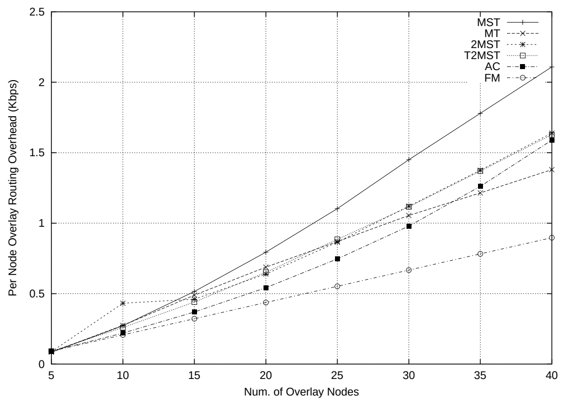

Fig. 16. Routing Overhead vs. Num. of Overlay Nodes. (Sprint topology, Feedback based Routing).

the overhead not only includes the overlay link performance probing overhead but also the overhead of exchanging link state information. To provide up-to-date performance information to all the overlay nodes, the link state information also needs to be updated frequently. In feedback based proactive routing, each overlay node needs to continuously monitor the backup paths performance. The overhead of this approach is only comprised of probing traffic overhead.

Fig. 15 shows the routing overhead of link-state based routing on different overlay topologies. From the figure, we can observe that the routing overhead is relatively steady with the increase in the number of overlay nodes in most of the overlay topologies excepting the full-mesh topology. Thus, the link-state routing protocol scales to large size overlay network when using these overlay topologies. The routing overhead in full-mesh topology increases dramatically with the increase in the size of overlay network. As shown in IV.A, the overhead complexity for full mesh topology is O(n^3). When comparing the overlay topologies, we can see that the minimum spanning tree has the least routing overhead.

However, from Fig. 16, we can see that minimum spanning tree has the largest routing overhead which confirms to the results in Fig. 12. In which, the overlay paths in this overlay topology are longer than all the other topologies. In addition, full mesh topology gives us the least overlay routing overhead when using feedback based proactive routing approach. As we

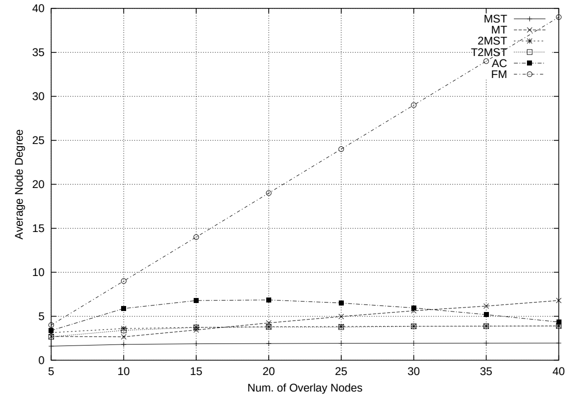

Fig. 17. Average Node Degree.

know, if we use feed-back based routing, the overhead is determined by the length of backup paths. We can always find the shorter overlay backup paths if we use full-mesh topology instead of other candidates. Comparing to Fig. 15, we can see that feedback based proactive approach incurs more routing overhead than link-state based routing in most of the overlay topologies. However, in feedback based proactive approach, the probing based path measurement gives us the end-to-end overlay path performance information while the link-state based routing only gives us the overlay link performance information.

D. Node Degree

Fig. 17 shows the different overlay topologies' per node degree distribution. From the figure, we can observe that except in full mesh overlay topology, the average overlay node degree maintains steady in other overlay topologies with the size of overlay network increases. Most of the average node degrees are around 5. The average node degree has the maximal value in adjacent connection overlay topology when the size of the overlay network is around half size of the physical network. When comparing Fig. 17 and Fig. 15, another fact we observe that the node degree is directly related to the routing overhead when using link-state based routing protocol.

E. Summary of Results

From the above simulation results, we can draw the following conclusions.

1) **The performance of overlay routing service is affected by the overlay topology.** For example, when using the full mesh topology, the link-state based routing can provide us with the best failure recovery ratio. However, when using feedback based proactive routing approach, it gives us the least failure recovery ratio. Especially, as the size of the overlay network increases, the performance of feedback proactive routing approach decreases significantly in full mesh topology. Thus, we need consider both the overlay routing protocol and the overlay topology at the same time when designing an overlay service network.
2) **Physical topology aware based overlay topology can achieve good performance in providing resilient routing service.** Among the simulated topologies, adjacent

connection and K-topology-aware spanning tree are two physical connectivity-aware overlay topologies. In these topologies, we consider the physical topology when constructing the overlay topology. The two overlay topology can always give us good failure recovery ratio no matter what overlay routing protocols we use. At the same time, they can only incur moderate path recovery penalty and routing overhead.

3) **Mesh tree is not a good choice as an overlay service topology.** Even though the mesh-tree topology can provide us good resilience service when considering overlay multicast protocol. However, its performance in overlay service network is not desirable. For example, it can provide similar performance as minimum spanning tree while larger routing overhead even though it has more overlay links than minimum spanning tree.
4) **Link-state based on-demand routing is scalable in most of the overlay topologies except full mesh.** The link-state based routing overhead at each node remains stable in most cases with the increase in the size of the overlay network except for full mesh topology. When considering both failure recovery ratio and routing overhead, the adjacent connection topology is a better choice than others. However, the feedback based proactive routing approach is not as scalable as link-state based routing protocol.

VII. Related Work

There has been a moderate amount of work in the area of overlay network. The effort on application-specific overlay networks has targeted on widely usable applications such as multicasting[10], [29], [15], [9], [7], [23] and peer-to-peer file sharing[25][22][1]. Several other work has been dedicated to proposing a general overlay service networks that can be used to provide value-added service for a variety of application-layer services, such as QRON[17], SON[12], Opus[8], YOID[14], OverQoS[27]. Another research effort is the Planet-lab[3] whose goal is to build a global testbed for developing and accessing new network services. It not only utilizes overlay technique but also provides overlay network a desirable test platform. X-Bone[5] is a system for the automated deployment of overlay network. It operates at the IP layer and based on IP tunnel technique. The main focus is to manage and allocate overlay link and router resource to different overlays and avoid resource contention among the overlays.

Resilient Overlay Network (RON)[6] is closely related to this paper. It is proposed to quickly detect and recover path outages and degraded performance. RON can better cope with the problem than BGP, which usually takes longer time to converge to a new valid route. A similar work was proposed in [30]. This work is based on Tapestry[31], which is a prefix based routing approach. It provides the resilient overlay routing by dynamically switching traffic to precomputed alternate routes. In addition, the messages can be duplicated and multicasted around the network congestion and failure hotpots with rapid reconvergence to drop duplicates.

In paper [16], the authors examines the role inter-domain topology and routing policy play in the process of delayed Internet routing process. In [21], the authors charcterize the real and generated physical topologies. Their focus is the difference between real topology and real Internet topology. The physical topology impact on four different multicast design issues were also studied in this paper. In [20], the authors use a game-theoretic approach to investigate the performance of selfish routing (overlay routing or source based routing) in Internet-like environments.

VIII. Conclusion & Future Work

In this paper, we studied the effect of overlay topology on the performance of overlay routing service. We evaluate the performance based on the following metrics: failure recovery ratio, overlay path hop penalty, overlay routing overhead, etc. Based on the simulation results, we can draw the following conclusions: 1) The overlay topology has significant impact on the overlay routing performance. The routing performance differs when the overlay service network takes different topology. 2) The underlying physical network information can benefit us a lot in constructing efficient overlay toplogies. 3) Adjacent connection topology and K-topology-aware spanning tree topologies can provide better routing performance than others. Even though the evaluation is based on resilient overlay routing service, we believe that it does reflect the overlay topology impact on overlay service network, which is an important aspect of designing an overlay service network.

There are several questions that need to be answered in our future work. 1) How to model the toplogy contruction problem so as to find an optimal soluation? 2) How to dynamicaly construct overlay topology so that it can adapt to the dynamic physial network performance? 3) We are planning to evaluate the overlay topologies based on the real Internet inter-AS topology as published in [26].

References

[1] Gnutella[online]. http://www.Gnutella.com.
[2] "GT-ITM: Modeling Topology of Large Internetworks,"[online]. http://www.cc.gatech.edu/projects/gtitm/.
[3] Planet Lab[online]. http://www.planet-lab.org.
[4] Qbone[online]. http://qbone.internet2.edu/.
[5] Xbone[online]. http://www.isi.edu/xbone.
[6] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, R. Morris, "Resilient Overlay Network," In Proc. ACM SOSP'01, pp.131-185, Oct. 2001.
[7] S. Banerjee, B.Bhattacharjee, C.Kommareddy,in Proc. of ACM Sigcomm 2002, Pittsburgh, Pennsylvania, August 2002.
[8] R. Braynard, D. Kostic, A. Rodriguez, J. Chase, and A. Vahdat, "Opus: An Overlay Peer Utility Service," In Proc. IEEE OpenArch'02, June 2002.
[9] Y. Chawathe, S. Mccanne, E. A. Brewer, "RMX: Reliable Multicast for Heterogeneous Networks," In Proc. Infocom'00, pp.795-804, March 2000.
[10] Y. Chu, S. G. Rao, H. Zhang, "A Case for End System Multicast," In Proc. ACM SIGMETRICS 2000, pp.1-12.
[11] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms," McGraw-Hill Book Company 1990.
[12] Z. Duan, Z. Zhang, and Y. T. Hou, " Bandwidth Provisioning for Service Overlay Networks", In Proc. SPIE ITCOM Scalability and Traffi c Control in IP Networks (II) '02, July 2002.
[13] H. Eriksson, "MBone: The Multicast BackBone," Communications of ACM, vol.37, no. 8, pp.54-60, Aug. 1994.
[14] P. Francis, "Yoid: Extending the Internet Multicast Architecture," http://www.aciri.org/yoid/docs/index.htm.
[15] J. Jannotti, D. K. Gifford, K. L. Johnson, M. F. Kaashoek, and J. W. O'Toole, Jr. "Overcast: Reliable Multicasting with an Overlay Network, " In Proc. 4th USENIX OSDI, Oct. 2000.

[16] C. Labovitz and A. Ahuja, " The Impact of Internet Policy and Topology on Delayed Routing Convergence,", In Proc. IEEE INFOCOM 2001.
[17] Z. Li and P. Mohapatra, "QRON: QoS-aware Routing in Overlay Networks," to appear in the special issue on Service Overlay Networks in the IEEE Journal on Selected Areas in Communications.
[18] Z. Li and P. Mohapatra, "HostCast: A New Overlay Multicasting Protocol," In Proc. IEEE Int. Communications Conference (ICC) 2003.
[19] A. Nakao, L. Peterson, A. Bavier, "A Routing Underlay for Overlay Networks," to appear in Proc. ACM SIGCOMM 2003.
[20] L.Qiu, R.Y.Yang, Y.Zhang and S. Shenker, " On Selfish Routing in Internet-Like Environments," in Proc. ACM SIGCOMM 2003.
[21] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, D. Estrin, "On characterizing network topologies and analyzing their impact on protocol design", USC-CS-TR-00-731, March 2000. http://citeseer.nj.nec.com/radoslavov00characterizing.html
[22] S. Ratnasamy, P. Francis, M. Handley, R. Karp and Scott Shenker", "A Scalable Content Addressable Network," In Proc. of ACM SIGCOMM 2001, Aug.2001.
[23] S.Y.Shi, and J.S.Turner,“Routing in Overlay Multicast Networks,” In Proc. IEEE Infocom’02, June 2002.
[24] N. Spring, R. Mahajan, and D. Wetherall, " Measuring ISP topologies with Rocketfule," In Proc. ACM SIGCOMM’02, Aug. 2002.
[25] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications,” In Proc. ACM SIGCOMM 2001, August 2001, pp. 149-160.
[26] L. Subramanian, S. Agarwal, J. Rexford, and R.H.Katz, “Characterizing the Internet Hierarchy from Multiple Vantage Points,” IEEE INFOCOM 2002.
[27] L. Subramanian, I. Stoica, H. Balakrishnan and R.H.Katz, “OverQoS: Offering Internet QoS using Overlays", In Proc. HotNet-I Workshop, October 2002.
[28] B. M. Waxman, "Routing of Multipoint Connections","IEEE Jornal on Selected Areas in Communications", Dec. 1988.
[29] B. Zhang, S. Jamin, L. Zhang,“Host Multicast: A Framework for delivering Multicast to End Users,” In Proc. Infocom’02, June 2002.
[30] B. Y. Zhao, L .Huang, A. D. Joseph,and J. D. Kubiaotowicz, "Exploiting Routing Redundancy Using a wide-area overlay," submitted for publication.
[31] B. Y. Zhao, L .Huang, A. D. Joseph,and J. D. Kubiaotowicz, "Tapestry: A Resilient Global-scale Overlay for Service Deployment," IEEE Journal on Selected Areas in Communications.
[32] D. Zhu, M. Gritter, D. R. Cheriton, "Feedback Based Routing",In Proc. ACM HotNet-I, 2002,

APPENDIX

Fig. 18 to Fig.28 show the simulation results for waxman random topology with 100 overlay nodes, which are correponding to the Sprint topology results from Fig. 7 to Fig. 17 respectively. For all the metrics we considered, these two set of results have the similar simulations results.

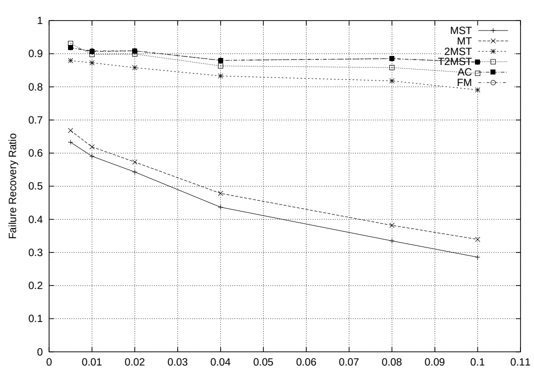

Fig. 18. Failure Recovery Ratio vs. Failure Ratio (Random topology, Link-state Routing, 50 overlay nodes).

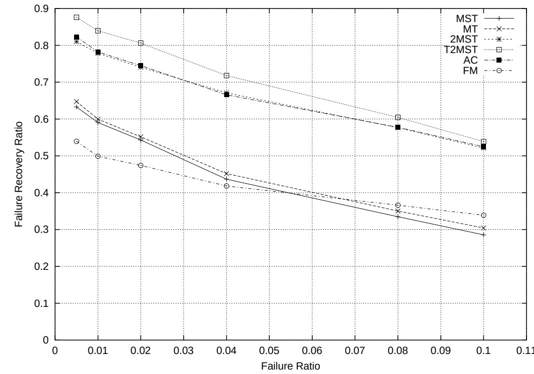

Fig. 19. Failure Recovery Ratio vs. Failure Ratio (Random topology, Proactive Routing, 50 overlay nodes).

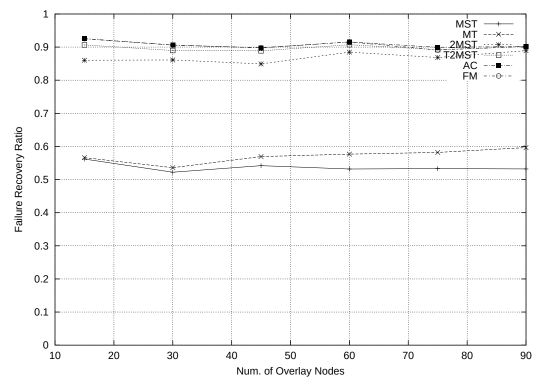

Fig. 20. Failure Recovery Ratio vs. Num. of Overlay Nodes (Randomtopology, Link-state Routing, failure ratio=0.02).

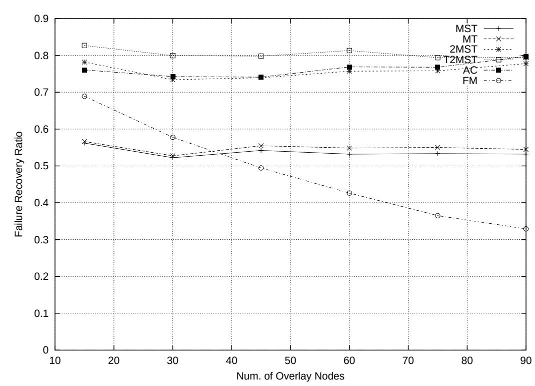

Fig. 21. Failure Recovery Ratio vs. Num. of Overlay Nodes (Randomtopology, Feedback based Routing, failure ratio=0.02).

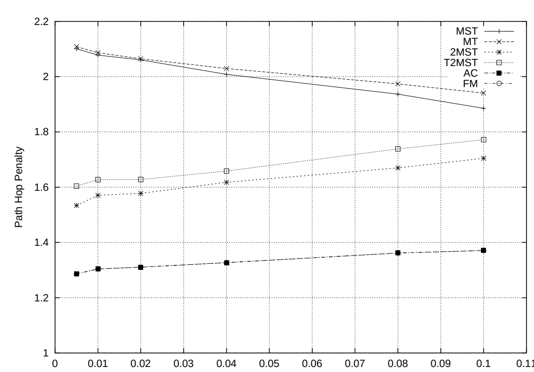

Fig. 22. Recovery Path Hop Penalty vs. Failure Ratio (Random topology, Link-state Routing, 50 overlay nodes).

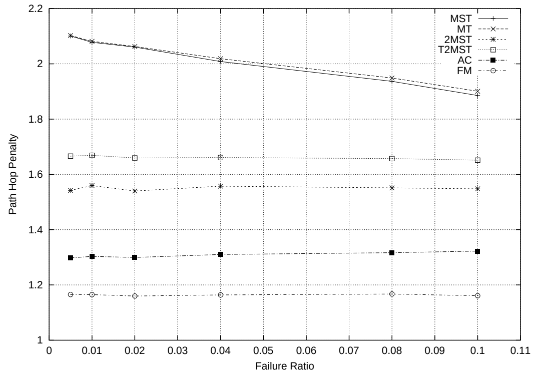

Fig. 23. Recovery Path Hop Penalty vs. Failure Ratio (Random topology, Feedback based Routing, 50 overlay nodes).

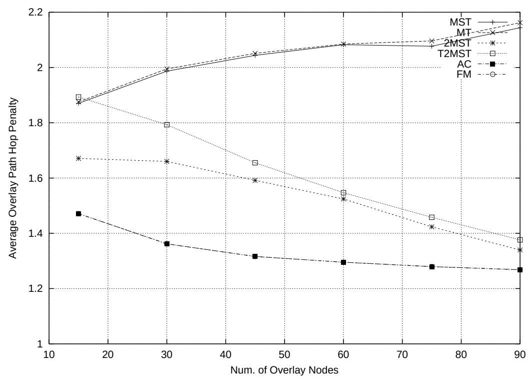

Fig. 24. Recovery Path Hop Penalty vs. Num. of Overlay Nodes (Random topology, Link-state Routing, failure ratio=0.02).

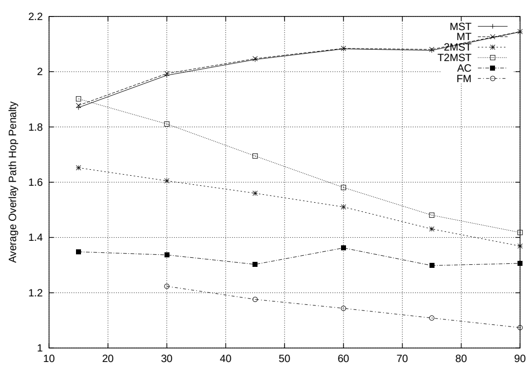

Fig. 25. Recovery Path Hop Penalty vs. Num. of Overlay Nodes (Random topology, Feedback based Routing, failure ratio=0.02).

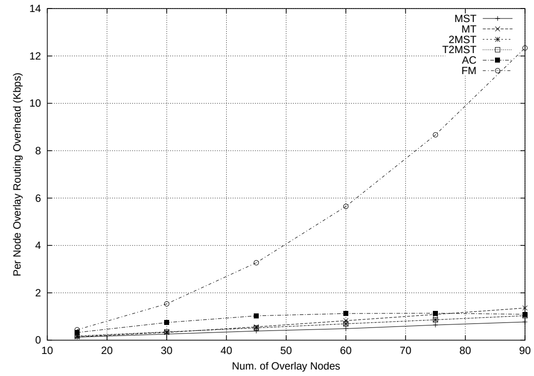

Fig. 26. Routing Overhead vs. Num. of Overlay Nodes. (Random topology, Link-state Routing).

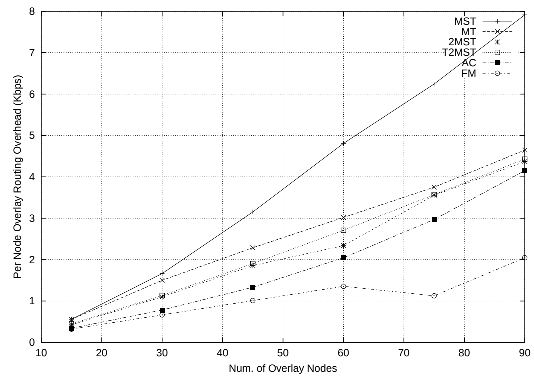

Fig. 27. Routing Overhead vs. Num. of Overlay Nodes. (Random topology, Feedback based Routing).

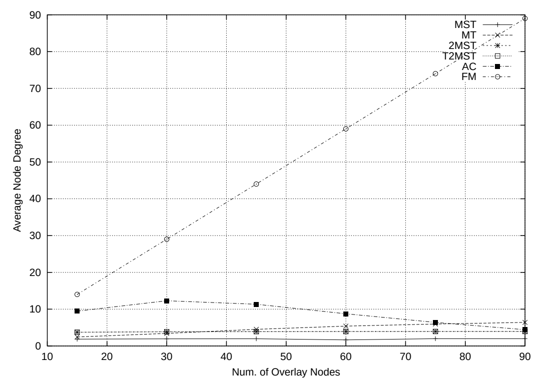

Fig. 28. Average Node Degree vs. Num. of Overlay Nodes. (Random topology, Feedback based Routing).