Load-balanced overlay recovery scheme with delay minimisation

Shengwen Tian* and Hongyong Yang

School of Information and Electrical Engineering,
Ludong University,
Yantai, 264025, China
Email: sw_tian@yeah.net
Email: hyyang@yeah.net
*Corresponding author

Abstract: Recovery from a link or node failure in the internet is often subjected to seconds or minutes of routing convergence, during which certain end-to-end connections may experience seconds or minutes of outage. According to this problem, existing approaches reroute the data traffic to a pre-defined backup path to detour the failed components. However, the maintenance of backup path increases the significant bandwidth expenditure. On the other hand, the diverted traffic may cause congestion on the backup path if it is not carefully split over multiple paths according to their available capacity. In this paper, we propose an efficient recovery scheme by using one-hop overlay multipath source routing, which is a post-failure recovery method. Once a failure happens, multiple one-hop overlay paths are constructed by selecting strategically multiple relay nodes, and the affected traffic is diverted to these paths in a well-balanced manner. We formulate the traffic allocation problem as a tractable linear programming (LP) optimisation problem, whose goal is to minimise the worse-case network congestion ratio. Simulations based on a real ISP network and a synthetic internet topology show that our scheme can effectively balance link utilisation dramatically and improve the reliability of network.

Keywords: failure recovery; load balance; overlay routing; linear programming.

Reference to this paper should be made as follows: Tian, S. and Yang, H. (2019) 'Load-balanced overlay recovery scheme with delay minimisation', *Int. J. High Performance Computing and Networking*, Vol. 13, No. 1, pp.119–128.

Biographical notes: Shengwen Tian works as a Lecturer in the Ludong University, China. He received his PhD in Computer Science and Technology from the Beijing University of Posts and Telecommunications in 2015. His research interests include overlay networks, P2P network, network virtualisation, cloud computing, data centre network and complex network.

Hongyong Yang works as a Professor in the Ludong University, China. He received his PhD in Control Theory and Control Engineering from the Southeast University in 2005. His research interests covers complex network, network communication, multi-agent systems, and intelligence control.

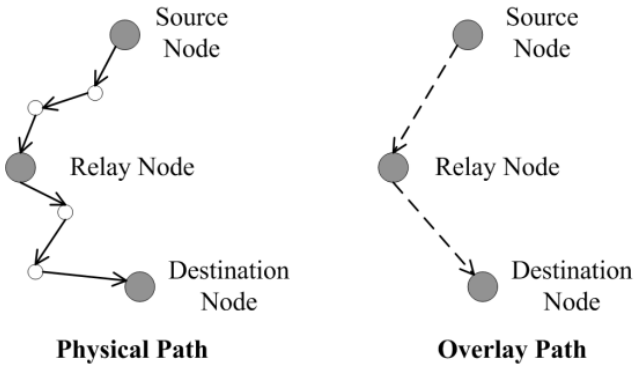
1 Introduction

Link and router failures are frequent in the internet (Markopoulou et al., 2004; Venkataraman and Chatterjee, 2012). The convergence time for routing protocols to route around these failures is often in the order of seconds or minutes (Griffin and Premore, 2001; Labovitz et al., 2001), during which certain end-to-end connections may experience seconds or minutes of outage (Boutremans et al., 2002). To mitigate the impact of failure, many IP fast recovery schemes have been proposed in the previous literatures (Hopps, 2000; Kini et al., 2010; Kvalbein et al., 2009), in which routers pre-compute and store backup paths for potential failures, and once a failure happens, the route will redirect the affected traffic to the backup paths. However, the maintenance of backup paths increases the significant bandwidth expenditure. On the other hand, the diverted traffic may cause congestion on the backup paths if their link available capacities are ignored.

In this paper, we propose a one-hop overlay source routing recovery scheme, by which the source can reroute the traffic to the destination relayed by an overlay node to detour the failed links when the given physical path suffers from the link failure or performance degradation. In one-hop overlay source routing, an overlay path consists of two overlay links, and each overlay link consists of one or multiple physical links, as shown in Figure 1.

With the rapid increase of new internet-based applications, such as voice-over-IP, peer-to-peer (Betts et al., 2014), and video-on-demand, large amounts of multimedia data need to be transmitted between source-destination node pairs. Under the situation, multipath routing is necessary for load balancing. In our scheme, multipath transmission is used to increase the throughput of the network. If multiple paths are not used in one-hop overlay source routing, the over-heavy traffic passing through the same relay node may cause frequent package loss and delay jitter, which can degrade the throughput and utilisation of network.

Figure 1 One-hop overlay path



Our proposed method is a post-failure recovery scheme, the recovery paths are provisioned only after the failure of primary path. If the default primary path works normally, the data transmission between the source-destination pair is fulfilled by routers in the physical network. When a path failure is detected, the source node selects $k(k \geq 2)$ overlay relay nodes to construct k one-hop overlay alternative paths, and then split its traffic into k sub-traffics, and reroute these sub-traffics through the constructed k one-hop overlay paths. Note that the traffic is rerouted from the source to the relay nodes and from the relay nodes to the destination along the shortest path in the physical network. Because our proposed one-hop overlay multipath routing is a post-failure recovery method, delay is an important consideration for the selection of k one-hop overlay paths. Spurred by the characteristics that a few nodes with high betweenness centrality can provide more optimal routes for a large number of node pairs in the internet (Cohen and Raz, 2014; Kawahara et al., 2009), we select a given number of overlay nodes whose betweenness centralities are higher than others as the candidate overlay relay nodes. $k(k \geq 2)$ overlay relay nodes are selected from the candidate relay nodes to construct k one-hop overlay paths, which can decrease the delay of network.

In addition, we also take into account the capacity of nodes and links for load balancing. The key to load balancing is to allocate reasonably the traffic over each one-hop overlay path, i.e., to determine an optimal split ratio. To solve this problem, a linear programming (LP) formulation is developed, whose goal is to minimise the worse-case network congestion ratio.

The rest of the paper is organised as follows. In Section 2, we introduce the related work. Section 3 presents the method of constructing one-hop overlay recovery paths. The post-failure load balance is described in Section 4. In Section 5, we discuss the deployment of our proposed scheme. In Section 6, we present and analyse the simulation results. Finally, the paper is concluded in Section 7.

2 Related work

Numerous IP failure recovery schemes have been proposed previously. The existing schemes can be classified into three categories (Tseng and Chung, 2012): loop-free alternate-based (LFA-based) scheme, backup routing table-based (BRT-based) scheme, and tunnelling-based (tunnel-based) scheme. In LFA-based scheme (Hopps, 2000), when a failure is detected, the routers adjacent to the failed link divert the affected traffic to the pre-determined alternate neighbouring nodes that ensure the loop-free property. However, the study (Raj and Ibe, 2007) shows that LFA-based scheme remains inadequate for protection coverage. BRT-based scheme achieves recovery in IP network by backup routing tables re-computed before any failure occurs (Kvalbein et al., 2009). The main design principle of LFA-based and BRT-based schemes is to compute and store backup paths for potential failures beforehand, which increases the significant maintenance expenditure. In tunnel-based scheme (Kini et al., 2010), when a router detects the adjacent link failure, it selects an intermediate router as temporary router to relay the failed traffic to the destination router. Although tunnel-based scheme is a post-failure recovery method, the adjacent link failure in a router is detected only when the router table of this router is updated periodically, during which the failed traffic cannot be diverted. Moreover, the encapsulation and decapsulation of packets introduce extra burden on routers in the network. Divakaran and Chinnagounder (2015) propose a restoration framework for survivable traffic grooming in optical networks based on the topological features of network.

Our proposed recovery scheme is a post-failure recovery method by using overlay multipath routing. There have been considerable researches on overlay routing to improve the reliability and performance of the internet. RON (Andersen et al., 2001) uses the overlay routing to quickly detect and recover path outages and degraded performance. Due to its full mesh architecture, RON requires that each node actively monitor all the other nodes and broadcast a full copy of its link state table. So, RON is lack of scalability, and it does not consider the load balancing. Cha et al. (2006) study the one-hop overlay routing problem for the robustness of network, which focused only on the placement of relay nodes in an intra-domain network. In SOSR, Gummadi et al. (2004) present the concept of one-hop source routing and study this problem by the experiment data on the PlanetLab. The results in SOSR show that one-hop source routing with four relay nodes selected randomly from the network can recover from 56% of network failures.

Cohen and Raz (2014) and Roy et al. (2009) study the cost associated with the intermediate node placement for overlay routing. Liao et al. (2016) propose an interest overlay network model to solve recommendation problem in social network service under distributed environment. Recently, overlay routing technology is also applied to solve the multi-controller synchronisation problem in software-defined networks (SDN) (Benamrane et al., 2016).

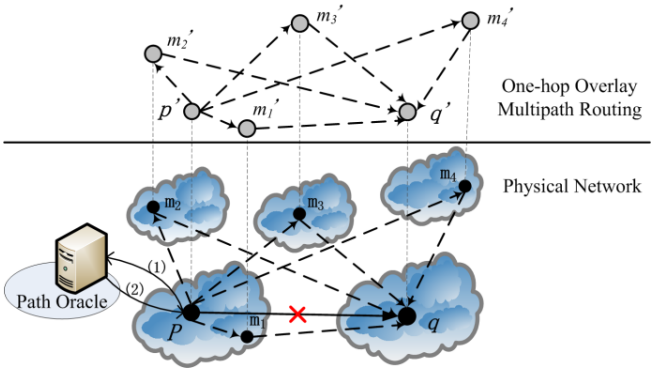
Many researches on load-balanced routing have been conducted (Hussein et al., 2015; Liu et al., 2016). Hussein et al. (2015) propose a weighted throttled load balancing algorithm by assigning a weight to each virtual machine (VM) in cloud environment. Liu et al. (2016) use Bayesian probabilistic inference to forecast the load information in multi-tenant service environment and proposed a service migration method for load balancing. Chen et al. (2016) propose a novel classification method of peer-to-peer network traffic identification based on machine learning for balancing the link load and improving QoS. Multipath routing schemes with load balancing has been widely applied in IP network, which can be classified into traditional IP-based and multiprotocol label switching (MPLS)-based. The IP-based multipath routing needs to extend the existing routing algorithms (RIP, OSPF, or BGP) for multipath support, which cannot take full advantage of multiple paths that exist frequently in internet service provider network (Lee and Choi, 2002). Although the MPLS-based multipath routing (Singh et al., 2012; Yoshida and Kawarasaki, 2012) is proposed as a powerful technology supporting load balancing recently, the sophisticated operations are performed by the MPLS traffic-engineering (TE) technology, which focuses on the IP-layer network. However, legacy networks mainly employ shortest-path-based routing protocols such as open shortest path first (OSPF) and intermediate system to intermediate system (IS-IS). This means that the IP routers deployed in the legacy networks need to be transformed into label switching routers (LSR) for supporting label distribution protocol (LDP), which will significantly increase the capital expenditures (Oki and Iwaki, 2010). In addition, TE needs to change the routing path frequently to adapt the dynamic traffic demand, which may cause the network instability. Different from the previous reports, our proposal is deployed at the application layer without any changes in the internet infrastructure.

3 One-hop overlay recovery paths setup

In our proposed one-hop overlay recovery scheme, when a path failure is detected, the source node first selects $k(k \geq 2)$ overlay relay nodes to construct k one-hop overlay alternative paths, and splits its traffic into k sub-traffics, and then re-routes these sub-traffics through the constructed k different one-hop overlay paths. During the rerouting, each sub-traffic is transferred between a source-destination node pair in two stages. First, k sub-traffics are directed to k different overlay relay nodes, respectively. Next, every relay node forwards the received sub-traffic to the final destination. The traffic is first routed from the source to the relay nodes and then from the relay nodes to the destination according to the shortest-path-based protocol in the physical network. For example, in Figure 2, when the source p suffers from a path failure to the destination q, its traffic is split into four sub-traffics (i.e., $k = 4$) and rerouted simultaneously through relay nodes m_1, m_2, m_3, m_4.

We elaborate our proposed scheme in two steps. In the first step, we select k suitable relay nodes to construct k one-hop overlay routing paths. In the second step, we address how to divert the failed traffic onto the recovery paths for minimising the congestion. The key of constructing k one-hop overlay routing paths is to select reasonably k relay nodes, in which we first define a set of candidate relay nodes to reduce the search space, and then select strategically k relay nodes from the set of candidate nodes.

Figure 2 One-hop overlay recovery routing with multiple paths (see online version for colours)



3.1 Selection of candidate relay nodes

In this paper, the physical network is represented as a directed graph $G(V, E)$, where V is the set of nodes and E is the set of links. The overlay nodes are given as a subset $Q \subseteq V$ where each node can be a source or destination. Let $|Q| = N$. We define d_{pq} as the traffic demand from the source node $p \in Q$ to the destination node $q \in Q$.

Because we need to select simultaneously k relay nodes for one-hop overlay multipath routing, the selection process of these k relay nodes is not independent. The time complexity of this process is equivalent to the combination number C_N^k. With the increase of the number of overlay nodes N, it becomes harder to select k relay nodes within a practical time. Therefore, we first define a set of candidate relay nodes $I \subseteq Q$ for the selection of relay nodes to reduce the search space, as shown in Figure 3.

Our ideas come from the characteristics that only a few nodes with high betweenness centrality are repeatedly present in many routing paths (Cohen and Raz, 2014). In other words, a small number of relay nodes can provide optimal routes to a large portion of end-to-end pairs. Betweenness centrality (Brand, 2008) of a node v is the sum of the fraction of all-pairs shortest paths that pass through v, which is denoted as follows:

$$BC = \sum_{s,t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1}$$

where V is the set of nodes, σ_{st} denotes the number of shortest paths from s to t, and for any $v \in V$, $\sigma_{st}(v)$ is the number of shortest paths from s to t that go through v.

Figure 3 Selection of candidate relay nodes (see online version for colours)

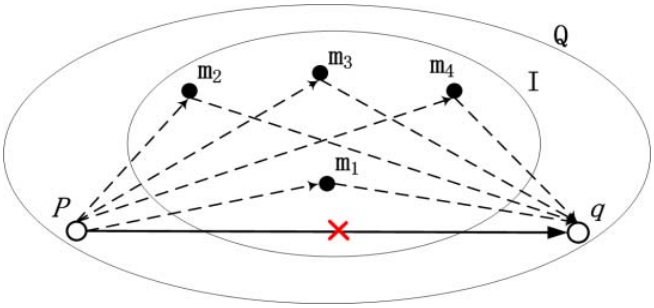
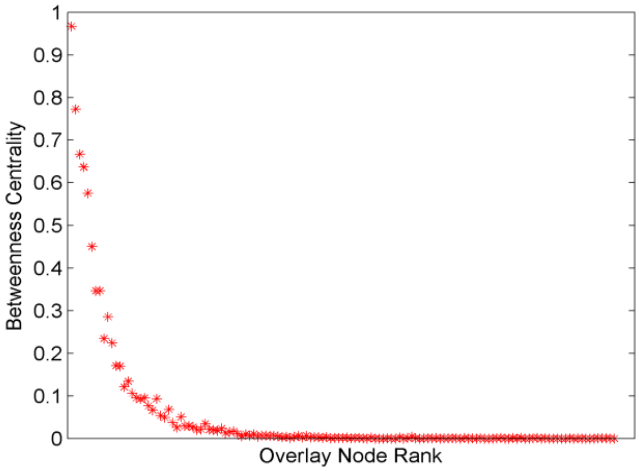


Figure 4 Betweenness centralities of nodes in the physical network (see online version for colours)



In order to validate these characteristics, we use the data of a real internet topology CN070 (Zhang, 2006) (depicted in detail in Section 6) and plot the betweenness centralities of all nodes in the network, as shown in Figure 4, where in x-axis node IDs are sorted by their betweenness centralities in a decreasing order. In CN070, the link bandwidth (available bandwidth) is assigned according to a uniform distribution in the range [40, 120] Mb/s. Assigning different weights to the links can generate different network topologies. In Figure 4, the betweenness centrality of each node is the average value after assigning the link weight for 2,000 times. From this figure, we can obtain that only a few nodes have extremely high betweenness centralities.

We select the nodes with higher betweenness centrality as the candidate relay nodes, which can reduce the hops of routing path between each source-destination node pair. To some extent, smaller routing hops means shorter delay. Therefore, we compute the betweenness centrality of each overlay node, and select M nodes with highest betweenness centralities in the overlay network as the candidate relay nodes and form the set I. The size of M depends on the size of physical network; the experiment data (in Section 6) show that about 10% of the network size can achieve a good effect. When the arrival or departure of some overlay nodes causes the changes of the set Q, we recalculate the betweenness centrality of each overlay node in Q to update the set I.

3.2 Selection of k relay nodes

Because our proposed one-hop overlay multipath routing is a post-failure recovery method, delay is an important consideration for constructing k one-hop overlay paths. The key to the constructing process is to select strategically k relay nodes. In this paper, we use congestion delay as a performance metric to select k best relay nodes from the candidate relay nodes set I.

According to the M/M/1 queuing model (Medhi, 2002), for a physical link e with capacity C_e, if its link load is L_e, the mean delay experienced by a single packet is $1 / (C_e - L_e)$. The total delay on this link is $L_e / (C_e - L_e)$. Therefore, the delay of network can be represented as follows:

$$\sum_{e \in E} \frac{\sum_{(p,q)} \xi^{e}_{pq(m)} d_{pq}}{C_e - \sum_{(p,q)} \xi^{e}_{pq(m)} d_{pq}} \tag{2}$$

where $\xi^{e}_{pq(m)}$ refers to a link indicator. $\xi^{e}_{pq(m)} = 1$ if the one-hop overlay path from p to q relayed by the intermediate overlay node m includes the link e; $\xi^{e}_{pq(m)} = 0$ otherwise. $\sum_{(p,q)} \xi^{e}_{pq(m)} d_{pq}$ is the total load of link e. Equation (2) is subjected to the constraints $\sum_{(p,q)} \xi^{e}_{pq(m)} d_{pq} < C_e$ for each link $e \in E$.

We select k intermediate nodes corresponding to the minimum k total delay of network as the relay nodes to construct k one-hop overlay recovery paths. The algorithm of the selection of relay nodes can be described as follows:

Algorithm 1 Selection of relay nodes

Input: $G(V, E)$, Q, source-destination pair (p, q) and d_{pq}.

Output: relay nodes set R.

1: compute the betweenness centralities BCs of all overlay nodes in $Q - \{p, q\}$ based on equation (1).

2: select M nodes from $Q - \{p, q\}$ as the candidate relay nodes according to the descending order of BCs, and obtain the candidate relay set I.

3: For $i = 1 \cdots k$

$m \in I$;

$$v = \arg \min_{m} \sum_{e \in E} \frac{\sum_{(p,q)} \xi^{e}_{pq(m)} d_{pq}}{C_e - \sum_{(p,q)} \xi^{e}_{pq(m)} d_{pq}};$$

$R = R \cup v$;

$I = I \setminus v$;

End for

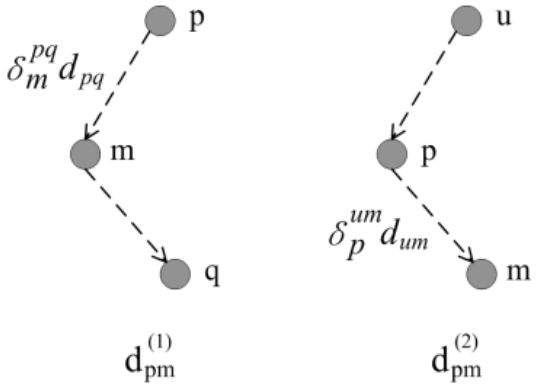
The number of one-hop overlay recovery paths depends on the value of k, which is critical. It should be not too small; otherwise, it is not good for load balancing. And it should not be too large because it is impractical and inefficient to detour data through such a large number of alternative paths. A suitable choice for the value of k is 4, as shown in Gummadi et al. (2004) based on internet experiments.

4 Post-failure load balancing

In this section, we address how to divert the failed traffic onto the recovery paths for minimising the network congestion. After the k one-hop overlay recovery paths are constructed, we need to allocate the failed traffic over each one-hop overlay path, i.e., to determine an optional split ratio for load balancing in the network.

For each traffic demand d_{pq} routed from source node $p \in Q$ to destination node $\in Q$, we define δ_m^{pq} as the fraction of traffic from p to q relayed by the relay node $m \in Q$ in the one-hop overlay routing, i.e., δ_m^{pq} is a split ratio.

Figure 5 Traffic distribution in one-hop overlay recovery scheme



We assume that d_{pm} refers to the traffic between node p and node m, which consists of two components, as shown in Figure 5. The first one is the traffic generated by node p and relayed by node m, which is defined as $d_{pm}^{(1)}$. The second one is the traffic for m relayed by node p, which is defined as $d_{pm}^{(2)}$. In other words, node p is the source and node m is the relay node in $d_{pm}^{(1)}$, while in $d_{pm}^{(2)}$ node p is the relay node and node m is the destination. It is easy to see that $d_{pm}^{(1)}$ and $d_{pm}^{(2)}$ hold:

$$d_{pm}^{(1)} = \sum_{q \in Q} \delta_m^{pq} d_{pq} \tag{3}$$

$$d_{pm}^{(2)} = \sum_{u \in Q} \delta_p^{um} d_{um} \tag{4}$$

Therefore, d_{pm} is given by:

$$\begin{aligned} d_{pm} &= d_{pm}^{(1)} + d_{pm}^{(2)} \\ &= \sum_{q \in Q} \delta_m^{pq} d_{pq} + \sum_{u \in Q} \delta_p^{um} d_{um} \end{aligned} \tag{5}$$

Likewise, d_{mq} is represented as follows:

$$\begin{aligned} d_{mq} &= d_{mq}^{(1)} + d_{mq}^{(2)} \\ &= \sum_{q \in Q} \delta_m^{pq} d_{pq} + \sum_{v \in Q} \delta_p^{mv} d_{mv} \end{aligned} \tag{6}$$

In addition, let $(i, j) \in E$ represent a directed link in the network from node $i \in V$ to node $j \in V$. To simplify the notation, we also refer to a link by e instead of (i, j). C_{ij} and L_{ij} is the capacity of link (i, j) and the load of link (i, j), respectively. The sets of incoming and outgoing edges at node i are denoted by $E^-(i)$ and $E^+(i)$, respectively. For each $i \in Q$, we denote the upper bounds on the total amount of traffic entering and leaving node i by $b^-(i)$ and $b^+(i)$ respectively, which can avoid overload on the node i. Let $\psi_{pm}^{ij} = 1$ if the shortest path from node p to node m traverses through the link (i, j), and $\psi_{pm}^{ij} = 0$ otherwise. In our proposal, the objective of post-failure load balancing is to minimise the network congestion ratio μ, which refers to the maximum value of all link utilisation rates in the physical network, and can be defined as follows:

$$\mu = \max_{(i,j) \in E} \left\{ \frac{L_{ij}}{C_{ij}} \right\} \tag{7}$$

Different source-destination node pairs have different relay nodes set R for one-hop overlay recovery routing. After determining the relay nodes set R, it is necessary to determine an optional split ratio for load balancing. We formulate the problem as a LP, which can be stated as follows:

$$\text{minimise:} \quad \mu \tag{8}$$

subject to:

$$\sum_{m \in R} \delta_m^{pq} = 1 \tag{9}$$

$$\sum_{m \in R} \left(\psi_{pm}^{ij} + \psi_{mq}^{ij} \right) \delta_m^{pq} d_{pq} + \beta_{ij} < \mu C_{ij}, \; p \neq q \tag{10}$$

$$\begin{aligned} &\sum_{(u,j) \in E^+(u)} \psi_{pq}^{uj} - \sum_{(i,u) \in E^-(u)} \psi_{pq}^{iu} \\ &= \begin{cases} +1, & \text{if } u = p \\ -1, & \text{if } u = q \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{11}$$

$$\sum_{q \in Q} d_{mq} \leq b^+(m), \quad m \in R \tag{12}$$

$$\sum_{p \in Q} d_{pm} \leq b^-(m), \quad m \in R \tag{13}$$

In LP (8) to (13), p, q and m denote the source node, the destination node and the relay node. The objective function in equation (8) minimises the network congestion ratio, i.e., maximises the throughput of the network. Constraint (9) states that the sum of δ_m^{pq} through k relay nodes m for each source-destination node pair in the one-hop overlay routing is equal to 1. Constraint (10) requires that the utilisation of each physical link on one-hop overlay path can not exceed the congestion ratio μ. $\psi_{pm}^{ij} \in \{0,1\}$ and $\psi_{mq}^{ij} \in \{0,1\}$. When $\psi_{pm}^{ij} = 1$ and $\psi_{mq}^{ij} = 1$, the physical link (i, j) simultaneously belongs to the overlay link (p, m) and (m, q). In constraint (10), β_{ij} is the background traffic of the link (i, j), which can be obtained from the traffic matrix. The values d_{pq} and C_{ij} in constraint (10) are constants, and hence this constraint is linear. Constraint (11) is the flow conservation constraint, ensuring that the variable ψ_{pq}^{ij} represents a flow of value 1 from p to q. Constraints (12) and (13) are the limitation of out- and in-traffic of the relay nodes, in which d_{pm} and d_{mq} depend on equations (5) and (6), respectively. We compute the split coefficient δ_m^{pq} based on LP (8) to 13), which can be solved optimally with a standard LP solver.

5 Deployment of one-hop overlay recovery paths

For the deployment of one-hop overlay recovery routing, it is essential to obtain some information about the physical network, such as the network topology and the traffic matrix. So, we need to deploy an entity (Path Oracle) in the physical network, as shown in Figure 2. The implementation of Path Oracle can refer to the previous literatures (Xie et al., 2008; Tutschku et al., 2009). The Path Oracle acts as an abstract routing underlay to the overlay network, which is a service offered by the ISPs. The oracle service can be realised as a set of replicated servers within each ISP, that is, we might deploy a server in each AS to collect some information about the AS topology and the network performance. So, the Path Oracle is implemented in a distributed and asynchronous manner.

When the source p detects a path failure to the destination q, the source first sends the request to the Path Oracle with the parameters, including the destination node and the traffic demand, and requests the Path Oracle to provide it with the addresses of k relay nodes and the corresponding split coefficient δ_m^{pq}, cf. Step 1 in Figure 2. Next, Path Oracle obtains the results calculated by our proposed algorithm, and then returns them to the requester, cf. Step 2 in Figure 2. Finally, the source p uses the received results to forward the traffic to the destination q via k relay nodes.

6 Performance evaluation

6.1 Simulation settings

To evaluate the performance of our proposed algorithm, we compare our algorithm with the method of selecting randomly k relay nodes. For convenience, we call it 'random method'. Random method is designed just for the selection of k relay nodes from the overlay nodes set Q. The corresponding congestion ratio μ_{rand} and the split coefficient δ_m^{pq} are computed based on LP (8) to (13). Random method selects k relay nodes randomly from the set Q. In addition, we also compute the non-split one-hop overlay routing and obtain its congestion ratio μ_{non}, in which the number of relay node is 1 and the relay node is selected randomly from the overlay nodes. Since the optimal (minimum) congestion ratio μ implies the maximum admissible network traffic, we define $S = 1 / \mu$, that is, for our proposed algorithm, $S_{our} = 1 / \mu_{our}$, while for random method and the non-split one-hop overlay routing, $S_{rand} = 1 / \mu_{rand}$ and $S_{non} = 1 / \mu_{non}$.

We carry out the simulation on top of two IP-layer topologies: a real AS-level topology CN070 (Zhang, 2006) with 135 nodes and 338 links, and a random topology GT180 generated by GT-ITM (GT-ITM: Modeling Topology of Large Internetworks, http://www.cc.gatech.edu/projects/GT/) with 200 nodes and 502 links. CN070 records the interconnection situation of most routers in China in 2006. GT180 is based on the Waxman (1988) probability.

In CN070 and GT180, link capacities are generated randomly with uniform distribution in the range of [80, 120]. d_{pq} is also generated randomly with uniform distribution in the range of [0,100]. $b^+(m)$ and $b^-(m)$, which are the capacities of overlay nodes, are also randomly generated in the range of [100, 200]. We set $b^+(m) = b^-(m)$ for each overlay relay node. The link weights used for shortest path computation and betweenness centralities computation are set to be $1 / (C_{ij} - L_{ij})$. We set the number of relay nodes $k = 4$, and select randomly a certain amount of nodes from the physical network CN070 and GT180 as the overlay nodes set Q, respectively. In each simulation, we randomly choose a pair of source and destination from the set Q. We assume that the IP-layer always takes the shortest path protocol based on the link-state information as its routing protocol.

We have implemented our proposed algorithm by MATLAB and CPLEX (ILOG Inc., 2006). For each simulation scenario, we run the simulation 2,000 times and obtain the average value for each performance metric.

6.2 Performance metrics

During the simulation, we use two performance metrics to evaluate the performance of our proposed algorithm. The first metric is the performance gain for our algorithm:

$$GAIN = \frac{S_{our}}{S_{non}} \tag{14}$$

For random method, $GAIN = S_{rand} / S_{non}$. Larger value of GAIN means smaller congestion ratio and greater network throughput.

To evaluate the recovery performance of our proposed algorithm about the link failure, we define the second metric as recovery path hop penalty (RPHP). We assume that the IP-layer always takes the shortest path protocol to connect the source and destination pairs. This means that the recovered overlay path may have higher hops comparing with the default IP-layer path. To some extent, longer IP-layer path means longer latency. In practice, data packets transmission between inter-autonomous system (AS) may not be along the shortest path (Cohen and Raz, 2014), this is because each AS is an independent business entity and the BGP routing policy reflects the inter-AS commercial relationships. In this case, the recovered overlay path may be shorter than the default IP-layer path. We use the following RPHP to quantify the overlay path's physical distance compared with original IP-layer path:

$$RPHP = \frac{\sum_{i}^{k} Num.\ of\ hops\ in\ i^{th}\ recovered\ path\ via\ overlay}{k \times Num.\ of\ hops\ in\ the\ corresponding\ failed\ IP-layer\ path} \tag{15}$$

6.3 Simulation analysis

In this section, we analyse the performance of our proposed algorithm under the network topology CN070 and GT180. We set $k = 4$, $M = 15$ in CN070 and $M = 20$ in GT180. Note that k is the number of one-hop overlay paths, and M is the size of candidate relay node set, i.e., $M = |I|$.

Figure 6 and Figure 7 show the effect of overlay network size on GAIN under CN070 and GT180, respectively. From Figure 6 and Figure 7, we can obtain that the value of GAIN obtained by our proposed algorithm is significantly greater than that obtained by random method. This indicates that the congestion ratio obtained by one-hop overlay recovery routing is smaller than that by random method.

In addition, the change trend of GAIN obtained by one-hop overlay recovery routing is similar to that obtained by random method. GAIN increases as the overlay network size increases. This is because more good nodes are selected as the relay nodes with the increase of overlay network size. Especially for one-hop overlay recovery routing, with the increase of overlay network size, GAIN under both CN070 and GT180 increases rapidly at the beginning, and then shows a slow increased tendency. This is because the overlay network size can affect the selection of relay nodes. And when the number of overlay nodes is small, a few nodes with higher betweenness centrality are selected frequently as the relay nodes, which results in the link overlap among k one-hop overlay paths, thus increasing the congestion ratio of the network.

Figure 6 Overlay network size vs. GAIN under CN070 (see online version for colours)

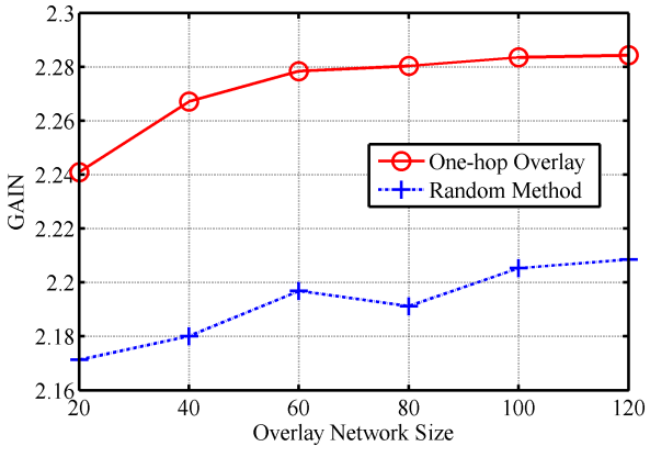


Figure 7 Overlay network size vs. GAIN under GT180 (see online version for colours)

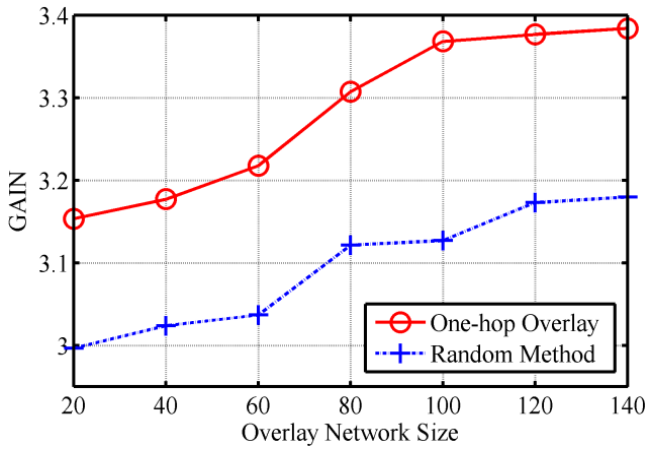


Figure 8 Overlay network size vs. RPHP under CN070 (see online version for colours)

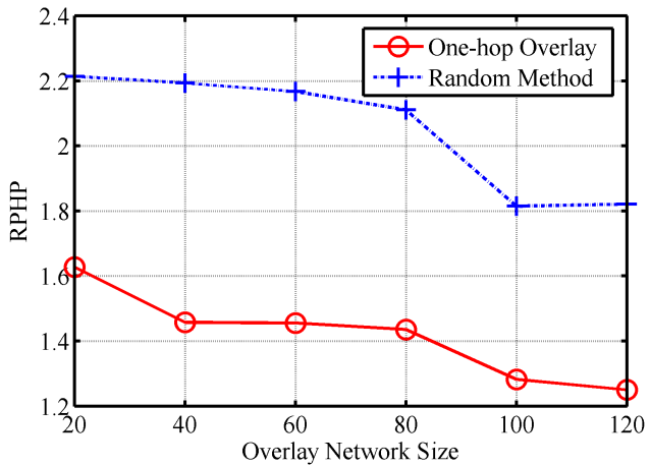


Figure 8 and Figure 9 show the recovery performance of the proposed algorithm under different overlay network sizes under CN070 and GT180, respectively. An IP-layer link failure means all IP-layer routing paths passing through this link fail. Because our proposed algorithm is a one-hop multipath source routing method and can always find the relay nodes to detour the failed links, we only use RPHP to

evaluate the performance of our proposed algorithm. From Figure 8 and Figure 9, we observe that the RPHP of our proposed algorithm is far superior to that by random method regardless of the overlay network size. This is because we select the nodes with higher betweenness centrality as the relay nodes in our proposal. Note that in our proposed algorithm the traffic is rerouted from the source to the relay nodes and from the relay nodes to the destination along the shortest path. These relay nodes with higher betweenness centrality might be on the shortest paths with high probability, which may reduce the routing path hops and lead to lower RPHP. On the other hand, from Figure 8 and Figure 9, we can obtain that the RPHP decreases with the increase of overlay network size under both CN070 and GT180. The reason is that with the increase of overlay network size, better-behaved nodes can be selected as relay nodes for one-hop overlay routing, which decreases the number of routing hops. To some extent, smaller routing hops means shorter delay. Therefore, our proposed algorithm can achieve better routing service.

Figure 9 Overlay network size vs. RPHP under GT180 (see online version for colours)

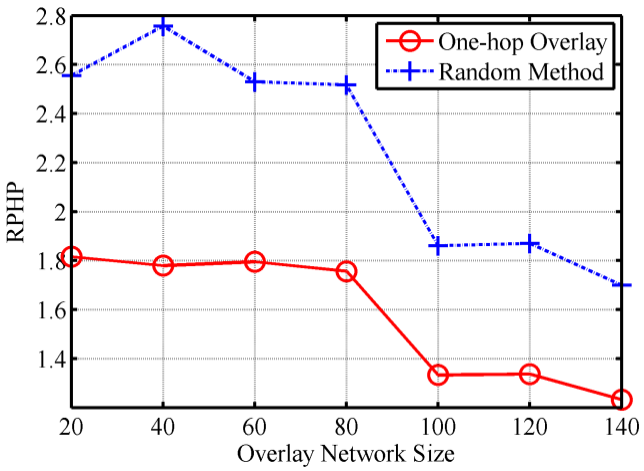


Figure 10 Num. of candidate relay nodes vs. congestion ratio (see online version for colours)

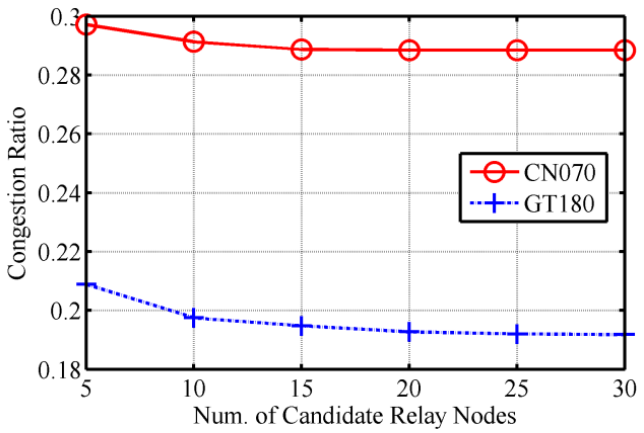
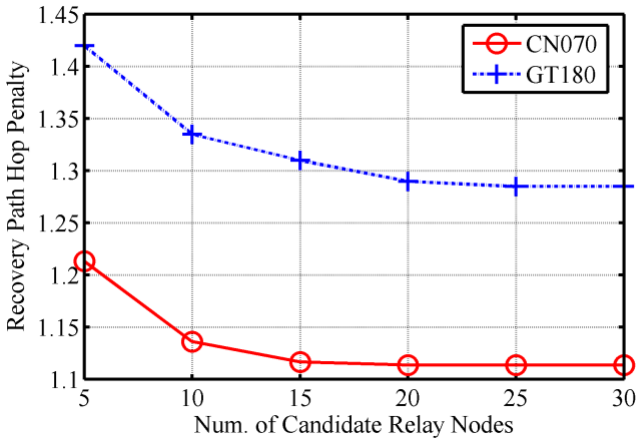


Figure 10 and Figure 11 show the effect of the number of candidate relay nodes on the performance, including congestion ratio and RPHP. We select randomly 50 nodes as overlay nodes from CN070 and GT180 respectively and vary the number of candidate relay nodes from 5 to 30. The simulation results are shown in Figure 10 and Figure 11. From these figures, we observe that with the increase of the number of candidate relay nodes, the congestion ratio and the RPHP decrease sharply at the beginning, and then decrease slowly, even almost retain the same value when the number of candidate relay nodes changes from 15 to 30. We also obtain that the number of candidate relay nodes '15' in CN070 is an inflection point for both congestion ratio and RPHP. By the same token, '20' is the inflection point in GT180. From these results, we can conclude that only a few candidate relay nodes are able to improve the performance of load balancing and decrease the routing hops for one-hop overlay recovery routing. In a word, our proposed algorithm is feasible and effective.

Figure 11 Num. of candidate relay nodes vs. RPHP (see online version for colours)



7 Conclusions

In this paper, a one-hop overlay recovery scheme is addressed by taking into account load balancing and minimum delay. In our proposed scheme, when a path failure is detected, the source selects strategically multiple overlay nodes to construct multiple one-hop overlay routing paths, and then splits the traffic and reroutes it concurrently through the selected one-hop overlay recovery paths. We take congestion delay as a performance metric to construct the one-hop overlay recovery paths for minimising the delay of network. Moreover, our proposed algorithm provides load balancing at the application layer instead of IP layer, which decreases the network overhead and improves the network utilisation. The simulation results show that the proposed algorithm can reduce the congestion ratio and improve the reliability of the network.

Acknowledgements

This work was jointly supported by National Natural Science Foundation of China (Nos. 61273152, 61673200), Foundation of Ludong University in China (No. LB 2016019, No. LB2016017) and Foundation of Department of Science and Technology of Shandong Province in China (No. 2013GGB01231).

References

Andersen, D., Balakrishnan, H., Kaashoek, F. and Morris, R. (2001) 'Resilient overlay network', *ACM Symposium on Operating Systems Principles (SOSP)*, pp.131–145.

Benamrane, F., Ros, F. and Mamoun, M-B. (2016) 'Synchronisation cost of multi-controller deployments in software-defined networks', *International Journal of High Performance Computing and Networking*, Vol. 9, No. 4, pp.291–298.

Betts, A., Liu, L., Li, Z. and Antonopoulos, N. (2014) 'A critical comparative evaluation on DHT-based peer-to-peer search algorithms', *International Journal of Embedded Systems*, Vol. 6, Nos. 2/3, pp.250–256.

Boutremans, C., Iannaccone, G. and Diot, C. (2002) 'Impact of link failures on VoIP performance', *Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, pp.12–14.

Brand, U. (2008) 'On variants of shortest-path betweenness centrality and their genetic computation', *Social Networks*, Vol. 30, No. 2, pp.136–145.

Cha, M., Moon, S., Park, C-D. and Shaikh, A. (2006) 'Placing relay nodes for intra-domain path diversity', *IEEE International Conference on Computer Communications (INFOCOM)*.

Chen, Y., Ji, H., Liu, H. and Sun, L. (2016) 'A traffic identification based on PSO-RBF neural network in peer-to-peer network', *International Journal of Computational Science and Engineering*, Vol. 13, No. 2, pp.158–164.

Cohen, R. and Raz, D. (2014) 'Cost effective resource allocation of overlay routing relay nodes', *IEEE/ACM Transactions on Networking (TON)*, Vol. 22, No. 2, pp.636–646.

Divakaran, S. and Chinnagounder, C. (2015) 'A framework for topology-based traffic grooming with restoration in optical networks', *International Journal of High Performance Computing and Networking*, Vol. 8, No. 4, pp.358–369.

Griffin, T-G. and Premore, B-J. (2001) 'An experimental analysis of BGP convergence time', *International Conference on Network Protocols (ICNP)*, pp.53–61.

GT-ITM: Modeling Topology of Large Internetworks [online] http://www.cc.gatech.edu/projects/gtitm/ (accessed 21 September 2016).

Gummadi, K-P., Madhyastha, H., Gribble, S-D., Levy, H-M. and Wetherall, D-J. (2004) 'Improving the reliability of internet paths with one-hop source routing', *Symposium on Operating Systems Design and Implementation (OSDI)*.

Hopps, C. (2000) *Analysis of an Equal-cost Multi-path Algorithm*, IETF RFC 2992.

Hussein, W., Peng, T. and Wang, G. (2015) 'A weighted throttled load balancing approach for virtual machines in cloud environment', *International Journal of Computational Science and Engineering*, Vol. 11, No. 4, pp.402–408.

ILOG Inc. (2006) 'ILOG CPLEX: high-performance software for mathematical programming and optimization' [online] http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html (accessed 21 September 2016).

Kawahara, R., Kamer, S., Kamiyama, N., Hasegawa, H. et al. (2009) 'A method of constructing QoS overlay network and its evaluation', *IEEE Global Telecommunications Conference (GLOBECOM)*.

Kini, S., Ramasubramanian, S., Kvalbein, A. and Hansen, A-F. (2010) 'Fast recovery from dual-link or single-node failures in IP networks using tunneling', *IEEE/ACM Transactions on Networking (TON)*, Vol. 18, No. 6, pp.1988–1999.

Kvalbein, A., Hansen, A-F., Cicic, T., Gjessing, S. and Lysne, O. (2009) 'Multiple routing configurations for fast IP network recovery', *IEEE/ACM Transactions on Networking (TON)*, Vol. 17, No. 2, pp.473–486.

Labovitz, C., Ahuja, A., Bose, A. and Jahanian, F. (2001) 'Delayed Internet routing convergence', *IEEE/ACM Transactions on Networking (TON)*, Vol. 9, No. 3, pp.293–306.

Lee, G. and Choi, J. (2002) *A Survey of Multipath Routing for Traffic Engineering*, Technical Report, Information and Communication University (ICU).

Liao, Z., Li, J. and Zhang, C. (2016) 'Interest overlay network model on distributed social network service', *International Journal of Embedded Systems*, Vol. 8, Nos. 2/3, pp.228–236.

Liu, S., Di, Z., Wu, L., Pan, L. and Shi, Y. (2016) 'Probabilistic-based workload forecasting and service redeployment for multi-tenant services', *International Journal of High Performance Computing and Networking*, Vol. 9, Nos. 1–2, pp.134–149.

Markopoulou, A., Iannaccone, G., Bhattacharyya, S., Chuah, C-N. and Diot, C. (2004) 'Characterization of failures in an IP backbone', *Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pp.2307–2317.

Medhi, J. (2002) *Stochastic Models in Queueing Theory*, 2nd ed., Academic Press, London.

Oki, E. and Iwaki, A. (2010) 'Load-balanced IP routing scheme based on shortest paths in hose model', *IEEE Transactions on Communications (TOC)*, Vol. 58, No. 7, pp.2088–2096.

Raj, A. and Ibe, O-C. (2007) 'A survey of IP and multiprotocol label switching fast reroute schemes', *Computer Networks*, Vol. 51, No. 8, pp.1882–1907.

Roy, S., Pucha, H., Zhang, Z., Hu, Y-C. and Qiu, L. (2009) 'On the placement of infrastructure overlay nodes', *IEEE/ACM Transaction on Networking (TON)*, Vol. 17, No. 4, pp.1298–1311.

Singh, R-K., Chaudhari, N-S. and Saxena, K. (2012) 'Load balancing in IP/MPLS networks: a survey', *Communications and Networks*, Vol. 4, No. 2, pp.151–156.

Tseng, P. and Chung, W. (2012) 'Joint coverage and link utilization for fast IP local protection', *Computer Networks*, Vol. 56, No. 15, pp.3385–3400.

Tutschku, K., Zinner, T., Nakao, A. and Tran-Gia, P. (2009) 'Network virtualization: implementation steps towards the future internet', *Workshop on Overlay and Network Virtualization at KiVS.*

Venkataraman, M. and Chatterjee, M. (2012) 'Quantifying video-QoE degradations of internet links', *IEEE/ACM Transaction on Networking (TON)*, Vol. 20, No. 2, pp.396–407.

Waxman, B-M. (1988) 'Routing of multipoint connections', *IEEE Journal on Selected Areas in Communication (JSAC)*, Vol. 6, No. 9, pp.1617–1622.

Xie, H., Yang, Y-R., Krishnamurthy, A., Liu, Y. and Silberschatz, A. (2008) 'P4P: provider portal for applications', *ACM SIGCOMM*.

Yoshida, Y. and Kawarasaki, M. (2012) 'Relay-node based proactive load balancing method in MPLS network with service differentiation', *IEEE International Conference on Communications (ICC)*, pp.7050–7054.

Zhang, G. (2006) 'An algorithm for internet AS graph betweenness centrality based on backtrack', *Journal of Computer Research and Development*, Vol. 40, No. 10, pp.1790–1796.