

# Policy-Aware Virtual Relay Placement for Inter-Domain Path Diversity

Tao Guo, Ning Wang, Rahim Tafazolli, and Klaus Moessner  
Centre for Communication Systems Research, University of Surrey  
Guildford, GU2 7XH, United Kingdom  
Email: {t.guo, n.wang, r.tafazolli, k.moessner}@surrey.ac.uk

**Abstract**—Exploiting path diversity to enhance communication reliability is a key desired property in Internet. While the existing routing architecture is reluctant to adopt changes, overlay routing has been proposed to circumvent the constraints of native routing by employing intermediary relays. However, the selfish inter-domain relay placement may violate local routing policies at intermediary relays and thus affect their economic costs and performances. With the recent advance of the concept of network virtualization, it is envisioned that virtual networks should be provisioned in cooperation with infrastructure providers in a holistic view without compromising their profits. In this paper, the problem of policy-aware virtual relay placement is first studied to investigate the feasibility of provisioning policy-compliant multipath routing via virtual relays for inter-domain communication reliability. By evaluation on a real domain-level Internet topology, it is demonstrated that policy-compliant virtual relaying can achieve a similar protection gain against single link failures compared to its selfish counterpart. It is also shown that the presented heuristic placement strategies perform well to approach the optimal solution.

## I. INTRODUCTION

In today's Internet, mission-critical applications such as Voice over Internet Protocol (VoIP) require end users stay connected during end-to-end communication sessions. The *de facto* inter-domain routing protocol, Border Gateway Protocol (BGP) [1], advertises only the *best* path for a destination prefix to its neighbors. In many cases, this mechanism causes that only one single forwarding path can be learned between each source-destination pair, especially for the low-tier Autonomous Systems (ASes). Given a failure-prone Internet due to various disruption causes such as maintenance, fiber cut, policy change and misconfiguration, single-path routing may dramatically degrade the application performance when a route change occurs since the dynamics of BGP may cause several minutes of packet loss during convergence.

While BGP amendments see little hope to change the current architecture in the near future due to the Internet ossification problem [2], overlay routing [3][4] has been proposed as an alternative approach to route traffic around failures by applying relay function at end hosts. However, the benefits of overlay routing in previous work mostly come at the expense of violating the administrative policies of the transit ASes [5]. These policies reflect the commercial agreements with interconnected ASes which tightly link with the economic profit, and furthermore, they determine the traffic engineering objectives inside an operating network. The policy violation

by overlay routing may significantly affect the benefits and performance of the underlying network operators. With the recent advance of the concept of network virtualization [2], it is envisioned that provisioning virtual networks on top of the physical network should come up with a holistic view. The virtual network technology should become a part of future Internet architecture innovations rather than a disruptive patch trading underlying network policies for its own objectives.

To provide effective protection paths for the primary path, it is expected that the number of physical links shared between the paths should be minimized since the physically-disjoint paths are very unlikely to fail at the same time. The achievable path diversity largely depends on how to place the relays. [6][7] studied the overlay node placement problem to maximize the path diversity inside an AS under an optimization framework. [8] formulated application-specific objective functions and evaluated their approaches on AS-level topologies but did not consider the policy constraints. To the best of our knowledge, there is no work investigating the problem of placing virtual relays for enhancing inter-domain communication reliability under the policy constraints. In this paper, the problem of policy-aware virtual relay placement is first studied to minimize the average unprotected link ratio of the default paths via up to  $k$  relays. Two heuristics are presented for the construction of the relay set. Evaluation on a real AS-level Internet topology shows that for almost any single link failure on a default path, at least one policy-compliant backup path can be found to route traffic around the failure. The proposed scheme in this paper can be adopted by virtual network providers to provide reliable transmission services without harming the benefits of infrastructure providers.

The rest of this paper is organized as follows: In Section II, the issue of overlay policy violation is described. Section III formulates the problem of placing virtual relays under policy constraints and presents two heuristics. Section IV evaluates the performance of policy-aware placement strategies. Finally, this paper is concluded in Section V.

## II. OVERLAY POLICY VIOLATION

The Internet today consists of tens of thousands of ASes, each of which is an independently administered domain in turn consisting of a number of connected routing prefixes. Each domain has its own local routing policies to express its willingness on where and how to route traffic to the remote

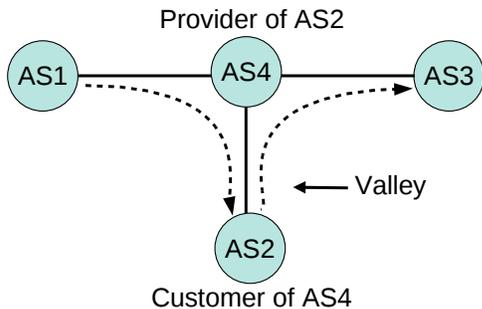


Fig. 1. An example of policy violation via overlay routing.

destination prefixes. These policies are mainly motivated by economic cost and performance gains, and reflect the business relationships with one's neighbors. Three common relationships observed in previous works are: Provider-Customer (P-C), Peer-Peer (P-P) and Sibling-Sibling (S-S) [9]. If an AS pays another higher-tier AS for any traffic sent or received through that AS, the payee and the payer form a provider-customer relationship. If two ASes of the same tier levels agree to freely exchange traffic for each other and their customers, but do not exchange traffic for their providers or other peers, they build a peer-peer relationship. The two belonging to the same administrative entity and thus freely exchanging any traffic between them via a dedicated link form a sibling-sibling relationship. Clearly, the basic rule governing the routing policies is that an AS would not provide transit services between two of its providers or peers since that AS is not receiving any economic reimbursement for this.

The transit policies are enforced by the BGP routing configurations to achieve a policy-compliant AS path between any source-destination prefix pair. However, in overlay routing, one or multiple relays are used to concatenate several native BGP path sections to form an end-to-end overlay path. Furthermore, as an application-layer solution, overlay functions are usually placed on end-hosts which are often located in a stub AS. It is very likely that an overlay path may violate the transit policies by forcing an customer to exchange traffic for their providers, or a peer to transit non-customer traffic to another peer. In this case, a valley will be formed as shown in Fig.1. In this example, a stub AS2 has to use its bandwidth to and from its upstream provider AS4 to act as a relay for the traffic from AS1 to AS3. It has been shown in [5] by a case study that about 70% overlay paths violate the transit policies. Even if overlay traffic is a small part of the overall traffic, policy violations may trigger routing instability. Due to the Internet ossification problem, the provision of virtual networks on top of the physical network has been considered as a key building component for the future Internet instead of just a modification to the current architecture. There is no doubt that the traffic imported by virtual network providers will experience significant growth in the future. The unexpected and spontaneous transit traffic may harm the economic profits of the transit ASes and degrade their traffic engineering objectives. Another problem of placing relay nodes at stub

ASes is the overlap of incoming and outgoing links for non-multihomed stubs, e.g. AS2 as shown in Fig.1. The two-way traffic may make the only link become a bottleneck while they are actually carrying the same information. The transit traffic may go through unnecessary delay in this case, which can be simply avoided by placing relays on their upstream providers.

### III. POLICY-AWARE VIRTUAL RELAY PLACEMENT

In this section, we first formulate the policy-aware relay placement problem and develop the objective function to characterize the feasibility of providing protection by placing virtual relays. To solve the problem, the policy-compliant relay candidate pools are built based a two-layer graph model. The approach of selecting  $k$  relays between each source-destination pair given a relay set is presented, based on which the objective function can be calculated. Due to the problem's *NP*-hardness, heuristic algorithms are presented to solve the problem.

#### A. Problem Formulation

To enhance the inter-domain communication reliability, it is expected that each inter-AS link on the default path should be provided with at least one backup alternative. If only one backup path is provided for the default path, the full physical disjointness is required to protect all the links on the default path, which is a rather rigid constraint. As multiple concurrent link failures rarely occur at AS level, one backup path can be used to protect part of the default path while the remaining can be protected by other backup paths. Each backup path can share different links with the default path. In this paper, we assume  $k$  detour paths can be applied to jointly protect the links of each default path. And single-hop overlay routing is adopted in each detour path, i.e. each detour path consists of a path section from a source to an intermediary and a path section from that intermediary to a destination. Whereas more hops can be used in a detour path to connect one intermediary to another intermediary, a common sense has been realized that no significant improvement is achieved with additional hops [4] [10]. Let  $L_d(s, d)$  denote the set of links on the default path  $p_d(s, d)$  connecting source  $s$  to destination  $d$ , and  $L_i(s, d)$  ( $1 \leq i \leq k$ ) denote the set of links on the  $i$ th detour path  $p_i(s, d)$  between the same two nodes. The Unprotected Link Ratio (ULR) for  $s$  and  $d$  using  $k$  detour paths is defined as:

$$ULR_{(s,d)}^k = \frac{\sum_{l \in L_d(s,d)} \begin{cases} 1, & l \in L_i(s,d) \text{ for all } 1 \leq i \leq k \\ 0, & \text{otherwise} \end{cases}}{|L_d(s,d)|} \quad (1)$$

where  $|L_d(s, d)|$  denotes the number of elements in the set  $L_d(s, d)$ . The intuition behind is that instead of utilizing a single path to protect all the default links, up to  $k$  paths can be selectively used, each of which can protect part of the links. Various approaches can be applied to detect the failed link and select the appropriate backup path. For example, a source can send packets through all  $k$  intermediaries in parallel and then route traffic through the one whose response packet is first returned [10].

As inter-domain path diversity is concerned in this paper, we ignore the network architecture inside a AS and treat each AS as a single entity. Consider a AS-level topology consisting of a set  $V$  of  $n$  ASes that are interconnected to neighbors by one of the relationships described in the previous section. Given a set of AS-level source-destination pairs  $S \subseteq V \times V$  and the allowed number of relays  $m$  to be placed, the problem of policy-aware virtual relay placement is formulated as follows:

$$\text{minimize : } \overline{ULR_{(s,d)}^k} = \frac{\sum_{(s,d) \in S} ULR_{(s,d)}^k}{|S|}$$

subject to :

$$R \subseteq V, \quad |R| = m$$

$$p_i(s, d) \text{ is a policy-compliant path, } 1 \leq i \leq k, \forall (s, d) \in S$$

While there are other factors contributing to the AS-level path selection, the business relationships with neighbors are dominant. In this paper, we assume that the AS-level paths are completely determined by the business relationships between ASes. Therefore, a policy-compliant path is referred to a valley-free path. The valley-free shortest path between two ASes is assumed to be the default path between them. We assume that all the ASes can act as potential relays. The number of relays in the relay set  $R$  is generally determined by the deployment cost. Clearly, without the constraint of the size of the relay set, the  $k$  best relays in  $V$  can be selected for each source-destination pair  $(s, d) \in S$ . However, it is uneconomic to deploy such many relays when  $|S|$  is large. The trade-off will be investigated in the evaluation section.

### B. Two-layer Graph Model

To facilitate path computation and relay placement, a relationship-embedded AS-level graph representation should be found. In [9], an annotated AS graph is proposed to represent AS relationships, in which only edges between providers and customers are directed. The concepts of downhill and uphill paths are introduced. A downhill path is a path  $(u_1, u_2, \dots, u_{n-1}, u_n)$  with  $u_{i+1}$  being the provider or sibling of  $u_i$ , and an uphill path is a path  $(u_1, u_2, \dots, u_{n-1}, u_n)$  with  $u_{i+1}$  being the customer or sibling of  $u_i$ , for all  $i < n$ . A valid valley-free path should be zero or one uphill path, followed by zero or one peer-peer edge, and then followed by zero or one downhill path. However, the annotated AS graph is only a partially directed graph which cannot be directly used for path computation. A two-layer directed graph model is proposed in [11] to compute edge-disjoint valley-free paths. A more simplified graph model is proposed in [12] to compute a valley-free shortest path between two ASes, which is adopted in this paper to form the basis of our path computation.

Given an annotated AS graph  $G(V, E)$ , where  $V$  denotes the set of AS vertices and  $E$  denotes the set of edges, the objective is to convert the partially directed graph  $G$  to a fully directed graph  $G'(V', E')$  such that all the relationships denoted by edges in  $G$  are completely mapped to directed edges in  $G'$ . The two-layer graph generation algorithm is described in Algorithm 1, which has two phases: vertex generation and

---

### Algorithm 1 Two-Layer Graph Construction

---

**Input:** Annotated graph  $G(V, E)$

**Output:** Fully directed graph  $G'(V', E')$

#### Phase 1: Vertex Generation

**for all** vertex  $u$  in  $G$  **do**

make two copies  $u_u$  and  $u_d$  in  $G'$

**end for**

#### Phase 2: Edge Generation

**for all** directed edge  $(u, v)$  in  $G$  **do**

add directed  $(v_u, u_u), (u_d, v_d), (u_u, v_d)$  in  $G'$

**end for**

**for all** undirected edge  $(u, v)$  in  $G$  **do**

**if**  $relationship(u, v) = \text{P-P}$  **then**

add directed  $(v_u, u_d), (u_u, v_d)$  in  $G'$

**else if**  $relationship(u, v) = \text{S-S}$  **then**

add directed  $(v_u, u_u), (u_u, v_u), (v_d, u_d), (u_d, v_d)$  in  $G'$

**end if**

**end for**

---

edge generation. As each vertex in  $G$  may have different roles in relationships with different neighbors, it can belong to either a downhill path or an uphill path in a valid valley-free path [9]. In the first phase, each vertex  $u$  in  $G$  is converted to two vertices  $u_u$  and  $u_d$  in  $G'$ , which denotes its possible role in an uphill path and in a downhill path, respectively. Thus, two layers are formed: uphill layer consisting of  $u_u$  and downhill layer consisting of  $u_d$ , for each  $u$  in  $G$ . In the second phase, edges are added in  $G'$  to represent the feasible transitions between each vertex pair. The principle is to make sure at most one inter-layer transition can be allowed in a path. For each directed edge  $(u, v)$  in  $G$ , the edge  $(v_u, u_u)$  denotes a valid link of an uphill path in uphill layer and similarly,  $(u_d, v_d)$  denotes a valid link of a downhill path in downhill layer. The edge  $(u_u, v_d)$  denotes an inter-layer transition, i.e. an uphill path is followed by a downhill path. For each edge  $(u, v)$  with P-P relationship in  $G$ , its occurrence in a path indicates an inter-layer transition since at most one such link can be allowed in a valley-free path. Two edges  $(v_u, u_d)$  and  $(u_u, v_d)$  are added to reflect the fact that peering ASes can carry traffic for each other. For each edge  $(u, v)$  with S-S relationship in  $G$ , it can appear in either an uphill path or a downhill path but does not indicate a transition between them. And therefore, a bidirectional connection is added in each layer. An example of converting an annotated AS graph to a two-layer fully directed graph is shown in Fig.2. It is stated in [12] that after the graph mapping,  $G'$  describes all the routing possibilities under the valley-free constraint and any conventional shortest path algorithm can be directly used on  $G'$  to obtain valley-free shortest paths.

### C. Build Relay Candidate Pools

Given a set of source-destination pairs  $S \subseteq V \times V$ , based on the two-layer graph model, a relay candidate pool  $CP(s, d)$

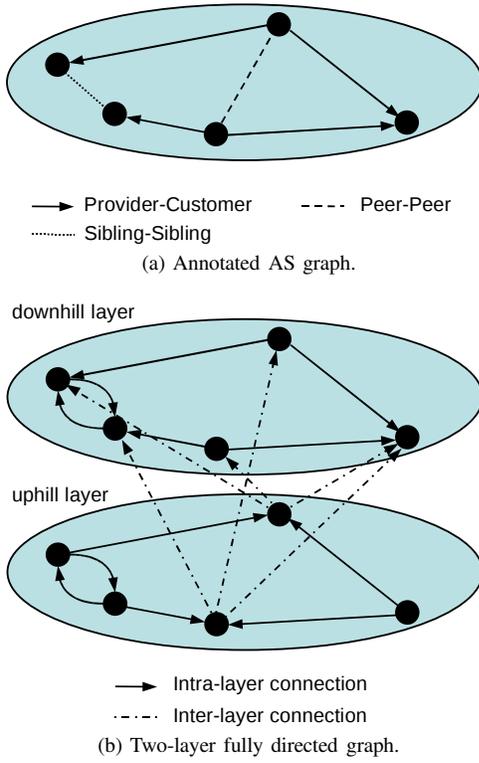


Fig. 2. An example of two-layer graph model construction.

can be built for each  $(s, d) \in S$  as presented in Algorithm 2. For each source  $s$ , single source shortest path algorithm is used to calculate the valley-free shortest path tree from  $s$  to all  $i \in V$  with the help of graph  $G'(E', V')$ . In particular, the uphill vertex  $s_u$  and downhill vertex  $s_d$  in  $G'$  are combined to a single vertex  $s$  by keeping all their outgoing edges while removing their incoming edges. Dijkstra's algorithm is run to obtain the shortest path from  $s$  to  $i_u$  and  $i_d$  in  $G'$ , where  $i_u$  and  $i_d$  are the uphill vertex and downhill vertex of  $i$  respectively. Then the shorter one is determined as the shortest path from  $s$  to  $i$ . Next, all the edges in  $G'$  are reversed to obtain a new graph  $G''$ . By applying the same single source shortest path algorithm with  $d$  as the source in  $G''$ , the shortest path tree from  $d$  to all  $i \in V$  is found. By reversing the shortest path tree, we obtain the shortest path from each  $i$  to  $d$ . Now we have calculated the valley-free shortest path from  $s$  to  $i$  and from  $i$  to  $d$  for all  $i \in V$ . In the following, the policy-compliant path concatenation will be found. Since S-S link can appear anywhere in a valid path, we first get the nearest non-sibling neighbor to  $i$  in both directions, which are denoted by  $u$  and  $v$  respectively. For representation simplicity, we use  $i$  to denote not only the relay candidate itself but also all its siblings. Clearly, the detour path via  $i$  will be policy-compliant if the path section  $(u, i, v)$  falls into one of the following cases: 1) an uphill path section; 2) a downhill path section; 3) an uphill link followed by a P-P link; 4) a P-P link followed by a downhill link. All the valid relays that are not on the default path  $p_d(s, d)$  are added in  $CP(s, d)$ . And we also obtain both the default path and the detour path database for each  $(s, d)$ .

---

### Algorithm 2 Build Relay Candidate Pools

---

**Input:**  $G'(E', V')$

**Output:**  $CP(s, d), \forall (s, d) \in S$

**for all**  $(s, d) \in S$  **do**

$CP(s, d) \leftarrow \emptyset$

$p(s, i) \leftarrow \text{single-source-shortest-path}(G', s), \forall i \in V$

**begin:**

-find corresponding vertex  $s_u$  and  $s_d$  in  $G'$

-combine all outgoing edges of  $s_u$  and  $s_d$  to a single vertex  $s$

-obtain path  $p(s, i') \leftarrow \text{Dijkstra}(G', s), \forall i' \in V'$

- $p(s, i) \leftarrow \text{shorter}(p(s, i_u), p(s, i_d)), i_u, i_d \in V'$

**end**

convert  $G'$  to  $G''$  by reversing all edges

$p(d, i) \leftarrow \text{single-source-shortest-path}(G'', d), \forall i \in V$

$p(i, d) \leftarrow \text{reverse}(p(d, i))$

**for all**  $i \in V, i \notin p_d(s, d)$  **do**

$u \leftarrow \text{nearest-non-sibling}(p(s, i))$

$v \leftarrow \text{nearest-non-sibling}(p(i, d))$

**if**  $\text{relationship}(i, v) = \text{P-C}$  **then**

**if**  $\text{relationship}(u, i) = \text{P-C}$  or  $\text{P-P}$  **then**

$CP(s, d) \leftarrow CP(s, d) \cup \{i\}$

**end if**

**else if**  $\text{relationship}(v, i) = \text{P-C}$  **then**

**if**  $\text{relationship}(i, u) = \text{P-C}$  or  $\text{P-P}$  **then**

$CP(s, d) \leftarrow CP(s, d) \cup \{i\}$

**end if**

**end if**

**end for**

**end for**

---

#### D. $k$ Relay Selection

In this section, we present how to select  $k$  relays for each  $(s, d) \in S$  based on the relay candidate pool  $CP(s, d)$  and thus to facilitate the objective function calculation. Given a set of relays  $R$ , the  $k$  relays are selected for each  $(s, d)$  using an iterative approach. The relay  $r \in R$  is first selected if  $r \in CP(s, d)$  and it can protect the maximum number of links for the default path, i.e. the link overlap between the detour path via  $r$  and the default path is minimized. If multiple relays can protect the same maximum number of the default links, the one incurring the least hops is selected. If all the links of the default path have been protected, no further relay will be selected. Otherwise, The second relay will be selected from  $(R \setminus \{r\}) \cap CP(s, d)$  to provide the maximum protection to the remaining unprotected default links. The process continues until all the default links have been protected or  $\min(k, m)$  relays have been selected. Note that if the relay set size  $m < k$ , all the relays in  $R$  will be selected. The objective function  $ULR_{(s,d)}^k$  can be calculated by selecting  $k$  relays and calculating the number of the unprotected links for each  $(s, d)$ , and averaging over all  $(s, d) \in S$ .

### E. Placement Strategies

The problem defined in Section III-A can be easily proved to be a *NP*-hard problem through a similar derivation in [8]. Therefore, two heuristics are presented in this section to effectively solve the problem.

- **Greedy Heuristic:** The classic greedy heuristic can be used to iteratively add a relay node to  $R$  until  $m$  nodes have been added. We denote the total relay candidate pool as  $CP(S) = \bigcup_{(s,d) \in S} CP(s,d)$ . Starting with an empty set  $R$ , a node  $r \in CP(S)$  that has the minimum value of the objective function  $ULR_{(s,d)}^k$  is first added to  $R$ . Then  $r$  is removed from  $CP(S)$ . In the second iteration, a node from the remaining total relay candidate pool  $CP(S) \setminus \{r\}$  that can incur the maximum decrease of the objective function is added. The process continues until  $m$  nodes have been added. The complexity of iteratively selecting  $k$  relays from a given relay set is about  $O(km|S|)$  and thus, the complexity of this greedy heuristic can be estimated as  $O(km|S| \cdot m|CP(S)|)$ .
- **Ranking-Based Heuristic:** For each candidate  $r \in CP(S)$ , we calculate its capability to independently protect all the default paths:

$$C_r = \sum_{(s,d) \in S} \begin{cases} 1 - ULR_{(s,d)}^1, & r \in CP(s,d) \\ 0, & r \notin CP(s,d) \end{cases} \quad (2)$$

where  $ULR_{(s,d)}^1$  denotes the unprotected link ratio with only the detour path via  $r$ . The nodes in  $CP(S)$  are sorted in decreasing order according to  $C_r$ . The first  $m$  nodes are added to the relay set  $R$ . The complexity of this heuristic can be estimated as  $O(|S||CP(S)| + |CP(S)| \log |CP(S)|)$ , where the first part of the sum denotes the complexity of calculating  $C_r$  for each candidate  $r \in CP(S)$  and the second part of the sum denotes the complexity of the sorting algorithm.

## IV. PERFORMANCE EVALUATION

In this section, we evaluate the policy-aware virtual relay placement strategies on a real Internet AS topology with embedded relationships from CAIDA dataset of January, 2005 [13]. To better show the achievable path diversity, any AS in the dataset with only a single physical link is pruned off and we obtain an annotated AS graph with 12,145 vertices. All vertices can act as potential relays. 10 groups of 100 source-destination pairs are randomly selected and the average results over all the groups are presented. We incrementally increase the size of the relay set.

For comparison, the random placement strategy is considered, in which a relay set is randomly selected from all vertices without considering policy constraints while only policy-compliant detour paths will be employed, referred as *Random*. Since the optimal solution of the presented problem is hard to obtain, a lower bound of the optimal solution is used to show the near-optimality of the heuristics. It is calculated by relaxing the constraint of the relay set size and selecting the best  $k$  relays from all vertices for each source-destination pair, referred as *LB(Policy)*. To evaluate the effect of the policy constraints, we also consider a lower bound of the objective

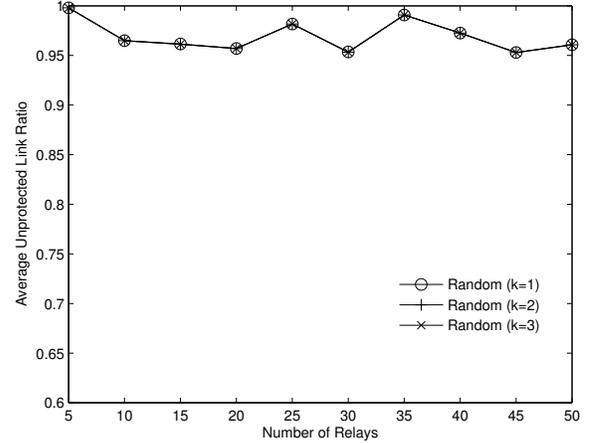


Fig. 3. Average unprotected link ratio of random placement.

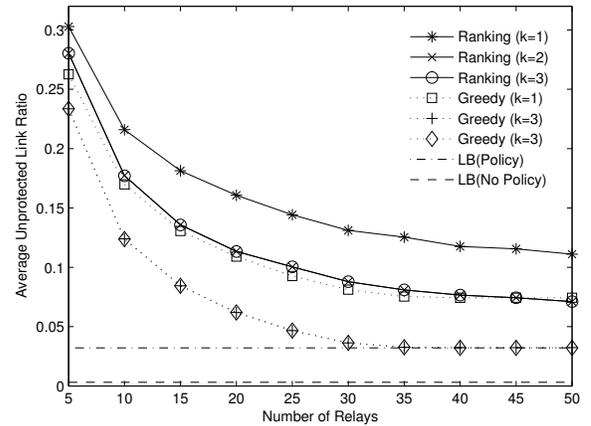


Fig. 4. Average unprotected link ratio of heuristics and LBs.

function when no policy constraints are enforced for the detour paths, referred as *LB(No Policy)*. In addition to the average unprotected link ratio, we also consider the average stretch ratio of all selected detour paths, defined as the ratio of the hop counts of a detour path to the one of its corresponding default path averaged over all valid detour paths. The stretch ratio of a detour path is tightly related to its capability to replace the default path.

The average unprotected link ratio of the random placement is shown in Fig.3. Clearly, the random placement does not consider the policy constraints when placing the relays and thus, very few policy-compliant detour paths can be found for the purpose of protection. The  $k$  value has little effect on the performance since there is no further relay that can be employed. In contrast, the policy-aware heuristic placement strategies take the policy constraints into account at the stage of placement, which significantly facilitates the following stage of path selection, as shown in Fig.4. In particular, greedy heuristic outperforms ranking based heuristic at the

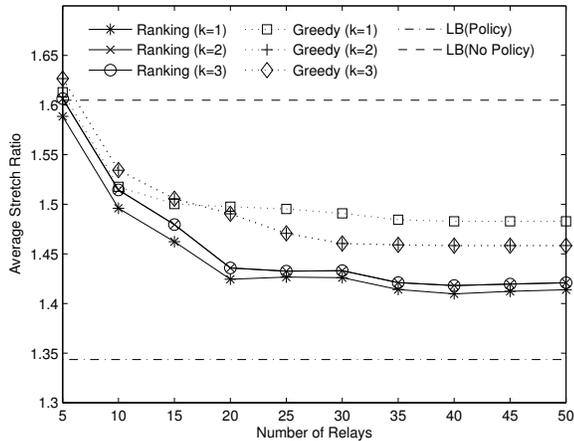


Fig. 5. Average stretch ratio of heuristics and LBs.

cost of increased calculation complexity. While ranking based heuristic can include some essential places into the relay set, its performance improvement is limited as more relays are included. The reason is that the following selected relays may cover the similar default paths that have been protected by the previously selected relays while leaving the remaining unprotected. It is worth noting that a hybrid heuristic by first selecting the essential places according to their ranks and then iteratively adding the remaining relays may achieve similar performance with reduced complexity.

In terms of the effect of the  $k$  values, it is shown that selecting two relays for each source-destination pair from the relay set can already achieve the maximum protection that can be provided by the relay set. In particular, the performance of the greedy heuristic with  $k = 2$  approaches the LB(Policy) when the relay set allows 30 places. If no policy constraints are enforced, a further reduction is possible but it is trivial. This shows that the policy constraints do not compromise the capability for at least single link failure recovery. In other words, for any single link failure on a default path, at least one policy-compliant backup path can be found to route traffic around the failure. Since multiple concurrent failures rarely occur at AS-level, we can say that effective link protection can be achieved without necessarily violating the routing policies.

To estimate the quality of the selected detour paths, the average stretch ratio is evaluated in Fig.5. While the minimum unprotected link ratio can be achieved by LB(No Policy), it significantly increases the stretch ratio since the most disjoint detour path may go through the edge of the Internet. In contrast, the policy constraints can enforce the detour paths going through the core of the Internet resulting in less path stretch. Compared to the ranking based heuristic, the greedy heuristic may incur a bit more path stretch. That is the cost of the reduced unprotected link ratio due to the limited relay set. Note that as the number of relays increases, the stretch ratios of both heuristics are reduced. That is because that as more options are available, they have a better chance to select

a path with both disjoint links and short hop counts.

## V. CONCLUSION

Traditional overlay routing may trade the local routing policies of the underlying providers for performance. The lack of a holistic view makes it unlikely to become part of future Internet. In this paper, we formulate the problem of policy-aware virtual relay placement for inter-domain path diversity and investigate the feasibility of providing policy-compliant backup paths via relays. By evaluation on a real AS-level Internet topology, we show that policy-compliant virtual relaying can provide an almost full link protection against single link failures. Furthermore, the presented placement heuristics approach the optimal solution with a small relay set. The proposed scheme can be adopted by virtual network providers to provide underlay-friendly reliable transmission services. For future work, it is of great interest to develop distributed placement strategies under policy constraints based on partial topologies and adopting to dynamic traffic demand.

## ACKNOWLEDGMENT

The work in this paper has formed part of the Flexible Networks area of the Core 5 Research Programme of the Virtual Centre of Excellence in Mobile & Personal Communications (Mobile VCE), and has been jointly funded by Mobile VCE's industrial partners and the UK Engineering and Physical Sciences Research Council (EPSRC).

## REFERENCES

- [1] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, Jan. 2006.
- [2] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the Internet impasse through virtualization," *Computer*, vol. 38, no. 4, pp. 34–41, Apr. 2005.
- [3] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: a case for informed internet routing and transport?," *IEEE Micro*, vol. 19, pp. 50–59, Jan. 1999.
- [4] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient overlay networks," in *Proc. ACM SOSP'01*, Banff, Alberta, Canada, Oct. 2001.
- [5] S. Seetharaman and M. Ammar, "Characterizing and mitigating inter-domain policy violations in overlay routes," in *Proc. IEEE ICNP'06*, Santa Barbara, California, USA, Nov. 2006.
- [6] M. Cha, S. Moon, C.-D. Park, and A. Shaikh, "Placing relay nodes for intra-domain path diversity," in *Proc. IEEE INFOCOM'06*, Barcelona, Spain, Apr. 2006.
- [7] V. Bui, W. Zhu, and L. T. Bui, "Optimal relay placement for maximizing path diversity in multipath overlay networks," in *Proc. IEEE GLOBECOM'08*, New Orleans, LA, USA, Dec. 2008.
- [8] S. Roy, H. Pucha, Z. Zhang, Y. Hu, and L. Qiu, "On the placement of infrastructure overlay nodes," *IEEE/ACM Trans. Netw.*, vol. 17, no. 4, pp. 1298–1311, Aug. 2009.
- [9] L. Gao, "On inferring autonomous system relationships in the Internet," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 733–745, Dec. 2001.
- [10] K. P. Gummadi, H. V. Madhyastha, S. D. Gribble, H. M. Levy, and D. Wetherall, "Improving the reliability of Internet paths with one-hop source routing," in *Proc. OSDI'04*, San Francisco, CA, USA, Dec. 2004.
- [11] T. Erlebach, A. Hall, A. Panconesi, and D. Vukadinovic, "Cuts and disjoint paths in the valley-free path model of Internet BGP routing," in *Proc. CAAAN'04*, Banff, Alberta, Canada, Dec. 2004.
- [12] D. N. Bauer, D. Dechouniotis, C.-X. Dimitropoulos, and A. Kind, "Valley-free shortest path method," U.S. Patent Application, 20090141637, Jun. 4, 2009.
- [13] The CAIDA AS relationships dataset, (January, 2005). [Online]. Available: <http://www.caida.org/data/active/as-relationships/>