Two Dimensional-IP Routing

Mingwei Xu
Tsinghua University

Shu Yang
Tsinghua University

Dan Wang
Hong Kong Polytechnic University

Jianping Wu
Tsinghua University

***Abstract*—Traditional IP networks use single-path routing, and make forwarding decisions based on destination address. Source address has always been ignored during routing. Lost of source address makes the traditional routing system inflexible and inefficient. The current network can not satisfy demands of both the network users and the ISP operators. Although many patch-like solutions have been proposed to bring the source address back to the routing system, the underlying problems of the traditional routing system can not be solved thoroughly.**

In this paper, we propose Two Dimension-IP Routing (TwoD-IP), which makes forwarding decisions based on both source and destination addresses. However, combining with source address, both the forwarding table and routing protocol have to be re-designed. To overcome the scalability problem, we devise a new forwarding table structure, which achieves wire-speeds and consumes less TCAM storage space. To satisfy demands of users and ISPs, we also design a simple TwoD-IP policy routing protocol. At last, we discuss the deployment problem of TwoD-IP.

I. Introduction

Internet has become one of the most successful communication networks world-wide, attracting billions of users and creating great number of applications. However, with more users, the Internet faces many challenges. For example:

- Traffic inside an ISP network is unevenly distributed;
- Complex network measurement and anomaly detection always annoy the network operators;
- Multi-path routing is hard to be used because of single-path routing in traditional networks;
- Flexible traffic management or policy routing is quite difficult, within destination-based single-path routing;

The current Internet makes forwarding decisions independently at each node according to the destination address of each packet. This simplicity, or dump core principle, of the traditional Internet pushes all complexities to the edges. However, for simplicity, traditional networks over-emphasize on their reachability to destinations, but do not pay much attention to other aspects related to sources. With the tremendous growing of the Internet, there are increasing demands for identifying the sources of traffic, e.g., ISPs usually desire to divert traffic from one customer network to an egress router, rather than the one selected by the best path selection algorithm of BGP [1]. The absence of source address identity in the routing system causes many problems. For example, it is difficult for malicious traffic from hackers to be filtered, and difficult for traffic for emergency service to take precedence.

To achieve better manageability and flexibility during routing, we are now deploying Two Dimensional-IP (TwoD-IP) routing. More specifically, the forwarding decisions of intermediate routers will be based on both the destination addresses and the source addresses. Packets from different sources towards the same destination may be delivered to different next hops in TwoD-IP routing, rather than the same one that is on the shortest path in traditional routing.

With TwoD-IP, the routing system will become more flexible, manageable and reliable. However, the new TwoD-IP routing architecture will cause additional overheads in both data and control planes, which can be seen as a trade-off between simplicity and flexibility. In data plane, storage cost may increase explosively with the addition of one more dimension in the forwarding table. In control plane, we need new routing protocols to control the routing paths from different sources.

We devise a new forwarding table structure for TwoD-IP. The new TwoD-IP forwarding table structure uses two separate TCAM tables to store source and destination prefixes, and a larger SRAM array to store the next hop information. When packets arrive, the router first lookups both source and destination addresses in the two TCAMs, and then use the output information to access the SRAM array and obtain the next hop information. Within the new structure, we can almost keep the same speed as the traditional destination-based forwarding table, and also realize a tolerable growth of storage.

We design a policy routing protocol based on extensions of OSPF. It can divert traffic from a customer network to another egress router rather than a default one. ISP operators can flexibly use the new protocol to carry out their policies.

We have developed prototypes of the TwoD-IP routers and new protocol on Bit-Engine 12004, and set up small scale tests under our testbed as well. The results show that TwoD-IP routers can achieve line speeds. The policy routing protocol is a simple example of TwoD-IP routing protocol; we can also design new protocols for other purposes.

II. Related Work

Due to the important semantic of source address, recent years see more research on giving sources control over routing.

IP (loose/strict) source routing [2], where the route is carried in the packet, is naturally combined with IP protocol, and allows the sender to take full control of the routing path. However, due to security reasons [3], source routing is disabled in most networks. In addition, source routing hands most control to the end users, which is unfavorable for ISP operators. MPLS [4] is often used to manage traffic per flow. However, due to the control and management overheads, MPLS raises concern about scaling when the number of label switching paths (LSPs) increases [5]. The more the LSPs, the heavier the system burden [6]. Overlay [7] can also be used. This however, is beyond the network layer. For an ISP, a light-weight, pure IP-based, and more network-controllable solution is desired.

There are many other routing schemes that have been combined with source address lookup, such as policy-based routing (PBR) [8], customer-specific routing [9], user-directed

routing [10], multi-topology routing [11], where traffic flows on user-specific topology [12]. In our paper, we try to design a routing architecture which is well combined with source address lookup, and scales in both control and data planes.

At edge routers, CERNET2 (China Education and Research Network 2) has deployed SAVI (Source Address Validation Improvement) [13], that guarantees that each packet will hold an authenticated source IP address. Currently, confirmed SAVI users are more than 900,000. CERNET2 then decides to further deploy source IP functionalities in its network.

III. Adding Source Address to the Routing System

In the current Internet routing, only destination address is used for forwarding decision making. This fundamentally limits the diversity of the functions and services that the Internet routing system can provide. Facing the demands from the users and applications, many proposals [2][14][4][7] provides additional functions by including source address, explicitly or implicitly, in their decision making; however, with their own syntax. It is widely accepted that the routing system today is less expressive and provides less basic primitive functions.

In this paper, we propose to add source address in the Internet routing system so that routers can make forwarding decisions based on both the source and the destination addresses. This greatly enriches the semantics the routing system can provide. Some services are illustrated as the following.

Example 1, Policy routing: An ISP wants the traffic from source address A to destination address B passes by router C. *With TwoD-IP routing*, routers in network make forwarding decisions based on destination and source addresses, thus they can recognize packets from A to B, and divert them to C.

Example 2, Traffic engineering with Load-Balancing: Assume an ISP has four routers with the topology shown in Fig. 1. Assume there are 50 hosts attached to the ingress router a, and each host sends traffic to the server attached to the egress router d at 1Mbps. The total traffic demand is 50Mbps. Using current destination-based single-path routing, traffic towards the same destination should take the same route. To achieve Min-max link utilization, all traffic will take the route through b and the maximum link utilization is 83.3%. *With TwoD-IP routing*, router a could differ according different sources. The optimal distribution is to let traffic of 30 hosts take the route through b, and traffic of the other 20 hosts take the route through c; the maximum link utilization is 50.0%.

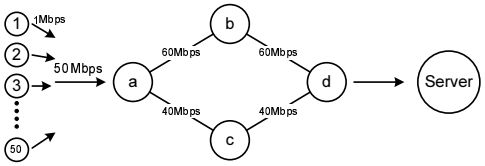

Fig. 1: TwoD-IP routing for better traffic distribution

Example 3, Diagnosis: In Fig. 2, assume an ISP has four routers. To monitor router b, c and d, the ISP sets up a monitor at router a. With destination-based routing, a has to send two probe packets. one to the destination of c and the other to the destination d. *With TwoD-IP routing*, by identifying the source address from a, b will recognize that these packets are for probing and can forward them by path (b, d), (d, c). So that only one probe packet is needed.

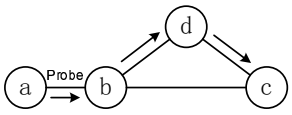

Fig. 2: TwoD-IP routing for network monitoring

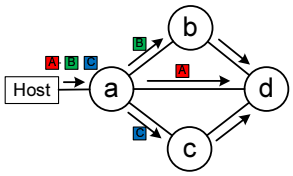

Fig. 3: TwoD-IP routing for Multi-path

Example 4, Multi-path Routing: The Internet is over-provisioned with links and bandwidth; it is well-known that the Internet routing can be more efficient with multi-path routing. However, it is not straightforward for an ISP to support flexible multi-path in a traditional routing system. ISPs have to go over through MPLS or overlay network, both of which bring overheads and complication. It is much simpler given TwoD-IP routing. See the example in Fig. 3, where the network has four routers, a host connected to router a and sends packets to d. *With TwoD-IP routing*, we can provide multiple paths towards the same destination at the same time. To achieve this, we only need to let the host own multiple source addresses, e.g., A, B and C. Router a can make forwarding decisions based on these source addresses (together with the destination address). For example, a can forward the packets with source address A directly to d, the packets with source address B to b, and the packets with source address C to c.

The benefits from adding source addresses to the routing system is not limited to the above examples. Intrinsically, we enrich the semantics of the entire Internet routing system.

IV. Overview of TwoD-IP

Fig. 4 shows the architecture of our TwoD-IP routing. Similar to the traditional architecture, it is separated into data plane and control plane.

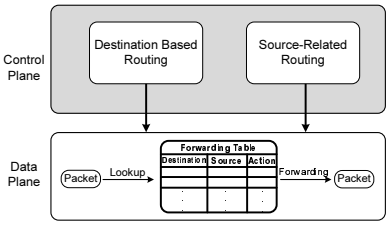

Fig. 4: TwoD-IP Routing Framework

A. Data Plane

Each entry of the TwoD-IP forwarding table is a 3-tuple, i.e., {source address, destination address, action}. When a packet arrives at a router, the router checks both destination address and source address, and then outputs a corresponding action (e.g., the next hop to be forwarded).

Compared with traditional forwarding table, the forwarding table in TwoD-IP routing can be much larger. If M is the size of source address space, a straightforward implementation will result in an increase of an order of M. We will discuss a novel architecture to address this problem in Section V-A.

B. Control Plane

Traditional routing protocol only exchanges network status information (e.g., network topology). TwoD-IP routing can meet more demands of the network users and ISPs. Therefore, the control plane can be more flexible. The key component is the routing protocols with updates according to both topology changes and policy changes. There are two components of the control plane of our TwoD-IP framework: destination-based routing protocols and source-related routing protocols.

- **Destination-Based Routing Protocol:** Traditional destination-based protocols, e.g., IS-IS and OSPF protocols that can run directly within the new architecture. The objective of these protocols is to provide connectivity services for users to reach the destinations. To provide better connectivity services, destination-based routing protocol should respond instantly to the changes of network topology.
- **Source-Related Routing Protocol:** Based on the combination of network status and user demands, we can make better decisions on routing for either users or ISPs. We will present one in Section V-A. Different routing protocols can coexist, although they need to be consistent. Source-related routing protocol should respond to the changes of user demands. Depending on the specific user demands, some source-related routing protocols need real-time updates, while others do not.

C. Key Challenges

Many opportunities can be explored, given that the TwoD-IP routing is deployed. To establish TwoD-IP routing, we consider the following main technical challenges.

1) Forwarding Table Design: The immediate change that TwoD-IP routing brings to the picture is the routing table size. More specifically, the Forwarding Information Base (FIB) will tremendously increase. Note that a first thought might think that the routing table only doubles. But this is not true, as for each destination address, it may correspond to different source addresses. A straightforward implementation means the FIB table should change from $\{\text{destination}\} \rightarrow \{\text{action}\}$ to $\{(\text{source address, destination address})\} \rightarrow \{\text{action}\}$.

This increases the FIB size for an order. The practical consequences can be calculated as follows. TCAM storage is 1 million. The current destination address space is 400,000. If TwoD-IP is used, and even if we only need to store 100 source prefixes, the total required storage is 40,000,000. This is far beyond a practical level. We solve this problem by proposing a novel FIST storage framework (see Section V).

2) New Source-Related Protocol: If all routers are equipped with source address checking functionality, we can design many source-related routing protocols for different purposes. Besides working correctly, the new protocols should be:

- Consistent: The protocols must be consistent with destination-based protocol and other source-related protocols. There must be no loops, and no policy conflicts.
- Efficiency: The protocol overheads should be low, e.g., maintaining minimum states on routers and bringing minimum exchanged messages between routers.

To illustrate the source-related protocol, we develop a simple policy protocol in Section V-B.

3) Incremental Deployment: Deployment is always a difficult problem for Internet routing systems. For TwoD-IP routing, it can be changed within an AS. Nevertheless, an incremental deployment is still greatly needed. The goals can be grouped into three levels: 1) backward compatiblity, 2) visible gain if only partial routers are deployed, 3) an upgrade sequence that can maximize the gain in each step. We believe 1) and 2) are a must and 3) needs to be greatly favored. We discuss incremental deployment in Section V-C.

V. Design

The TwoD-IP design has three main components: forwarding table, routing protocol, and deployment scheme. We describe each design component in turn.

A. FIST: Forwarding Table Design

We propose a novel forwarding table structure FIST (**FI**B **S**tructure for **T**woD-IP) for our TwoD-IP forwarding table. It achieves fast lookup and small memory space. The key of our design is a novel separation of TCAM and SRAM. TCAM contributes to fast lookup and SRAM contributes to a larger memory. Overall, our TwoD-IP forwarding table consumes $O(N+M)$ TCAM storage space, where each of N and M is the size of destination and source address space.

Let d and s denote the destination and source addresses, p_d and p_s denote the destination and source prefixes. Let a denote an action, more specifically, the next hop. The storage structure should have entries of 3-tuple (p_d, p_s, a).

Definition 1: Assume a packet with source address s and destination address d arrives at a router. The destination address d should first match p_d according to the Longest Match First (LMF) rule. Then source address s should match p_s according to the LMF rule among all the 3-tuple given p_d is matched. The packet is then forwarded to the next hop a.

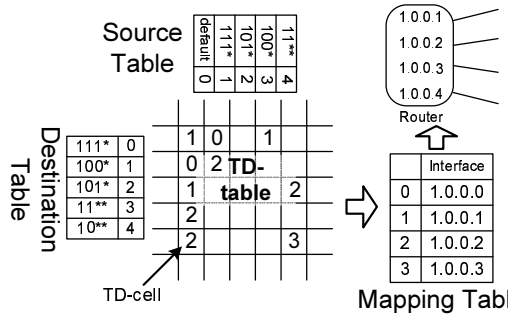

Fig. 5: FIST: A forwarding table structure for TwoD-IP

The new structure FIST is made up of two tables stored in TCAMs and other two tables stored in SRAM (see Fig. 5). One table in TCAM stores the destination prefixes (we call it *destination table* thereafter), and the other table in TCAM stores the source prefixes (we call it *source table* thereafter). One table in SRAM is a two dimensional table that stores the indexed next hop of each rule in TwoD-IP (we call it *TD-table*

thereafter) and we call each cell in the array *TD-cell* (or in short *cell* if no ambiguity). Another table in SRAM stores the mapping relation of index values and next hops (we call it *mapping-table* thereafter).

For each rule (p_d, p_s, a), p_d is stored in the destination table, and p_s is stored in the source table. We can obtain a row address in TD-table through p_d, and a column address in TD-table through p_s. Combining the row and column addresses, we can access a cell ((p_d, p_s) is used to denote the cell) in TD-table, and obtain an index value. According to the index value, a is stored in the corresponding position of mapping table. We store the index value rather than the next hop a in the TD-table, because next hop information is much longer.

For example, in Fig. 5, for rule $(100*, 111*, 1.0.0.2)$, $100*$ is stored in destination table and is associated with 1_{st} row, and $111*$ is stored in source table and associated with 1_{st} column. In the TD-table, the cell $(100*, 111*)$ corresponding to 1_{st} column and 1_{st} row has index value 2. In the mapping table, the next hop that is related to index value 2 is 1.0.0.2.

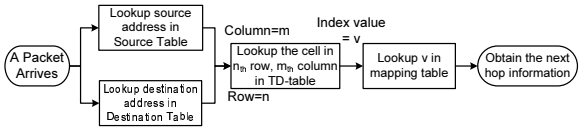

Fig. 6: Lookup action in FIST

The lookup action $lookup(d, s)$ is shown in Fig. 6. When a packet arrives, the router first extracts the source address s and destination address d. Using the LMF rule, the router finds the matched source and destination prefixes in both source and destination tables that reside in the TCAMs. According to the matched entry, the source table will output a column address and the destination table will output a row address. Combined with the row and column addresses, the router can find a cell in the TD-table, and return an index value. Using the index value, the router looks up the mapping table, and returns the next hop that the packet will be forwarded to.

Theorem 1: The look up speed of FIST is one TCAM clock cycle plus three SRAM clock cycles.

Proof: Source and destination tables can be accessed in parallel, one TCAM clock cycle is enough. Getting the row and column address costs one SRAM clock cycle, Accessing TD-table and mapping table costs two SRAM clock cycles. ■

As a comparison, the conventional destination-based routing usually stores destination prefixes in one TCAM, and accesses both TCAM and SRAM once during a lookup process. Note that the SRAM clock cycle is much smaller than TCAM cycle [15], and the bottleneck of a router normally happens during delivering packets through the FIFO. Thus two more accesses in SRAM will not have a significant impact on throughput.

In the TD-table of FIST, there are cells that do not have index values, e.g., cell $(101*, 111*)$ in Figure 5. We call these cells *conflicted cells*, i.e., confliction happens when packets match them. To address the problem, we should pre-compute and fill the conflicted cells. For example, we should fill $(101*, 111*)$ with 2, which is the index value of $(101*, 11**)$.

As a proof-of-concept, we implement the FIST forwarding table structure on a commercial router, Bit-Engine 12004. Our implementation is based on existing hardware, and does not need any new hardware. We re-design the hardware by rewriting about 1500 lines of VHDL code (not including C code) of the original destination-based version. The evaluation results show that FIST can achieve line speeds.

B. PORPT: Routing Protocols Design

The TwoD-IP architecture provides great opportunities and flexibility for the ISPs to deploy routing protocols for different purposes. In this section, we design a policy routing protocol PORPT (**Po**licy **R**outing **P**rotocol for **T**woD-IP), which illustrates a simple example for a TwoD-IP routing protocol.

Our goal is to divert traffic from some specified customer network to any edge router. For example, in Fig. 7, customer networks are connected to ISP network through provider edge routers (PE routers, e.g., B_0 and B_1), and ISP network is connected to foreign Internet through edge routers (e.g., E_0, and E_1). Besides the PE and edge routers, there are other routers (P routers, e.g., I_0, I_1, I_2, I_3) in the network. Currently, traffic from customer networks to the foreign Internet all passes through E_0. The objective of the ISP is to move the traffic from B_1 towards the foreign Internet to E_1.

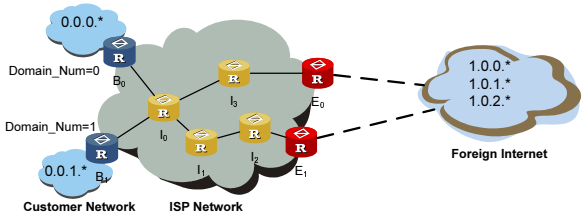

Fig. 7: Example of policy routing

We design an intra-domain routing protocol combined with OSPF. Additional information is disseminated and received through extensions of OSPF [16]. Such that routers can know the preference of customer networks, which can be obtained through manual configuration or automatic selection.

With these conditions, the edge router will announce foreign Internet prefixes information to intra-domain, including the identity of the edge router itself. The PE router will announce its preferences on edge routers, and the binding information between its customer prefixes and customer domain number. The routers of the ISP can compute the TwoD-IP forwarding table based on these information. We first describe the PORPT protocol details and then describe how to transform the information into the two dimensional routing table.

Let $Foreign\_Prefix$ be a foreign Internet prefix, $Customer\_Prefix$ be a customer prefix, $Router\_ID$ be the IP address of a router and $Domain\_Num$ be the domain number of a customer network. We define messages as follows.

- $Announce(Foreign\_Prefix, Router\_ID)$: This message is sent by an edge router of $Router\_ID$ to announce a foreign Internet prefix IP_Prefix.
- $Bind(Customer\_Prefix, Domain\_Num)$: This message is sent by a PE router to announce the binding in-

formation between a customer prefix and domain number of this customer network.

- $Pref(Domain\_Num, Router\_ID)$: This message is sent by a PE router, to announce the preference for a customer network on an edge router.

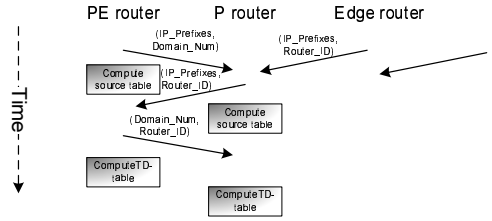

Fig. 8: Time line of the policy routing protocol

Fig. 8 shows the time line of PORPT. The edge routers just have to announce the foreign Internet prefixes combined with its own router identification to intra-domain. The PE routers have to announce the binding between its customer domain number and customer prefixes, PE routers also have to announce the preferences on edge routers. After obtaining the foreign Internet prefixes and preferences of customer networks, both PE and P routers should compute the two dimensional routing table.

For example, in Figure 7, PE router B_0 will announce the binding information by sending $Bind(0.0.0.*, 0)$, B_1 will announce $Bind(0.0.1.*, 1)$. Edge router E_0 will announce three foreign Internet prefixes combined with its own identification by sending $Announce(1.0.0.*, E_0)$, $Announce(1.0.1.*, E_0)$, $Announce(1.0.2.*, E_0)$, E_1 will announce $Announce(1.0.0.*, E_1)$, $Announce(1.0.1.*, E_1)$, $Announce(1.0.2.*, E_1)$. At last, B_1 will announce $Pref(1, E_1)$. Receiving these messages, PE and P routers can construct the two dimensional routing tables, we show the routing table on router I_0 in Table I.

TABLE I: Two dimensional routing table on the P router I0

Destination Prefix	Source Prefix	Next Hop
1.0.0.*	0.0.1.*	I1
1.0.1.*	0.0.1.*	I1
1.0.2.*	0.0.1.*	I1

We have developed a prototype of PORPT, set up tests under VegaNet [17], a high performance virtualized testbed.

C. Deployment

It is widely known that making changes to the network layer is notoriously difficult. We consider two important problems in the deployment. First, during the deployment, the proposed protocols should have small impact on the Internet protocols and infrastructure. Second, at the initial stage, a node-by-node incremental deployment scheme is highly preferred to minimize error and support efforts.

Currently, we mainly focus on a node-by-node incremental deployment scheme. We consider the most important factor for the success is that the deployment should have visible benefits after each node is deployed. We have a separate study on this problem [18]. The key investigated problem is that without full deployment, the resulting paths for traffic from some sources may deviate from the required ones, (i.e., pre-defined by users or ISP providers), then how to find node deployment sequences to minimize the deviation. We rigidly defined the deviation and mathematically formulated the problem.

We developed several algorithms for different practical scenarios and a case study on CERNET2. Our main observation is that we can gain the majority of the performance when only a small percentage of carefully selected nodes are deployed.

VI. Conclusion and Future Work

We presented the TwoD-IP architecture, which is closely combined with source address during routing. With TwoD-IP, the semantics that the routing system can provide are greatly enriched. There are also great challenges that we should face during designing and implementing TwoD-IP. In this paper, we described our initial design for TwoD-IP.

References

[1] E. Chen and S. Sangli, "Avoid BGP Best Path Transitions from One External to Another," RFC 5004 (Standards Track), Internet Engineering Task Force, Sep. 2007.

[2] D. Estrin, T. Li, Y. Rekhter, K. Varadhan, and D. Zappala, "Source Demand Routing: Packet Format and Forwarding Specification (Version 1)," RFC 1940 (Informational), Internet Engineering Task Force, May 1996.

[3] C. Perkins, "IP Encapsulation within IP," RFC 2003 (Standards Track), Internet Engineering Task Force, Oct. 1996.

[4] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," RFC 3031 (Standards Track), Internet Engineering Task Force, Jan. 2001.

[5] S. Yasukawa, A. Farrel, and O. Komolafe, "An Analysis of Scaling Issues in MPLS-TE Core Networks," RFC 5439 (Informational), Internet Engineering Task Force, Feb. 2009.

[6] C. Metz, C. Barth, and C. Filsfils, "Beyond mpls ... less is more," *Internet Computing, IEEE*, vol. 11, no. 5, pp. 72 –76, Sep 2007.

[7] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient Overlay Networks," in *Proc. ACM SOSP'01*, Banff, Canada, October 2001.

[8] *Policy-Based Routing (white paper)*, Cisco, 1996.

[9] J. Fu and J. Rexford, "Efficient ip-address lookup with a shared forwarding table for multiple virtual routers," in *Proc. ACM CoNEXT'08*, Madrid, Spain, Dec 2008.

[10] P. Laskowski, B. Johnson, and J. Chuang, "User-directed routing: from theory, towards practice," in *Proc. ACM NetEcon'08*, Seattle, WA, USA, Aug 2008.

[11] N. Wang, K.-H. Ho, and G. Pavlou, "Adaptive multi-topology igp based traffic engineering with near-optimal network performance," in *Proc. IFIP-TC6 NETWORKING'08*, Singapore, May 2008.

[12] *Multi-topology routing (white paper)*, Juniper, Aug 2010.

[13] J. Wu, J. Bi, M. Bagnulo, F. Baker, and C. Vogt, "Source address validation improvement framework," Internet Draft, Mar 2011, draft-ietf-savi-framework-04.txt.

[14] P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," RFC 2827 (Best Current Practice), Internet Engineering Task Force, May 2000.

[15] J. Kim, M.-C. Ko, H.-K. Kang, and J. Kim, "A hybrid ip forwarding engine with high performance and low power," in *Proc. ICCSA'09*, Seoul, Korea, Jun 2009.

[16] A. Zinin, A. Roy, L. Nguyen, B. Friedman, and D. Yeung, "OSPF Link-Local Signaling," RFC 1940 (Standards Track), Internet Engineering Task Force, Aug. 2009.

[17] C. Wenlong, X. Mingwei, Y. Yang, L. Qi, and M. Dongchao, "Virtual network with high performance: Veganet," *Chinese Journal of Computers*, vol. 33, no. 1, pp. 63–73, 2010.

[18] S. Yang, D. Wang, M. Xu, and J. Wu, "Efficient two dimensional-ip routing: An incremental deployment design," Tsinghua University, Tech. Rep., July 2011. [Online]. Available: http://www.wdklife.com/tech.pdf