Inter Domain QoS Routing using VS scheme

Bikramjit Walia and Girish P. Saraph, Member IEEE
Department of Electrical Engineering,
IIT Bombay, Powai, Mumbai-76, India

Abstract— Many real-time multimedia and critical applications can migrate to Internet, provided it can support better QoS in an Inter-domain environment. The main challenges on this front can be split into two parts; the first part is to develop mechanisms for information exchange, signaling and path set-up. The second part is to find an optimal inter-domain path subjected to multiple QoS constraints. We address the second part in this paper.

The QoS routing problem under multiple QoS constraints is of NP-complete class and is an area of active research. In addition, the inter-domain path selection scheme should perform well under limited information visibility and have good scalability and convergence. We look at a heuristic algorithm for inter-domain QoS routing based on Virtual Space (VS) embedding. A highly scalable VS routing scheme supports distributed local path selection and performs well even under limited information visibility. We present how VS is able to find multiple inter-AS paths under hop, delay, bandwidth and path protection constraints. We analyze the Internet in the form of a two-level hierarchy and take up large representative network topologies for our simulation study. Our heuristic approach based on VS embedding is verified by the simulation results.

I. INTRODUCTION

Many real-time applications such as VoIP, streaming, multimedia, video conferencing are being developed today. These applications require QoS from the Internet. Here, the QoS requirements are represented as a set of constraints, such as link constraints, end-to-end path constraints, or tree constraints for the entire multi-cast tree. The constraints can be based on bandwidth, delay, delay jitter, loss ratio, and so on. Common routing protocols use only the hop count or a single cost metric to find feasible paths. Not many protocols try to provide optimal routes based on multiple QoS parameters. That is because the complexity of path computation increases rapidly under multiple routing metrics. The problem of finding an optimal path subject to multiple constraints is NP-complete [1] and not tractable under two additive constraints [2].

This paper looks at inter domain QoS routing i.e. routing that takes place between autonomous routing domains. Most of the QoS routing policies have addressed the problem of routing within a single domain. Prior research points out that the peering links that connect distinct routing domains are often congestion points for network traffic [3]. Managing the resources at such points of congestion is very challenging and directly affects the end-to-end quality of service. Inter-domain QoS routing thus poses far more challenges than those faced in intra domain routing.

Internet is formed with diverse administrative domains or ASes which typically correspond to different operational set-ups, organizations or geographical regions. This is achieved by splitting the complete network into a 2-level hierarchical structure. Each domain adopts its own routing and implementation policies to ensure privacy, security and operational flexibility. It shares only limited accessibility information with its neighboring ASes through the border gateway nodes. The current de-facto standard for inter-domain routing is the Border Gateway Protocol. BGP adopts a policy based routing mechanism where each domain applies local policies to select the best route. It then decides which routes are to be advertised to neighboring domains. Business relationships between different domains at times govern path selection between ASes [4]. Also, at times, the final outcome may be a path that is locally optimal at some domains but globally sub-optimal due to the lack of a uniform policy or metric used to find an end-to-end route. It is shown [5] that for a large number of paths in the Internet there are alternate paths that exhibit superior quality as measured by round-trip, loss rate, and bandwidth.

The researchers are looking at extensions to the current BGP protocol to alleviate the problems it suffers from. In this regards, several heuristic algorithms have been proposed to solve the problem of QoS routing under multiple constraints in the current BGP-scenario. One such research has focused at source oriented topology aggregation with multiple QoS parameters in hierarchical ATM networks [6] under a ‘static’ scenario. Wang-Crowcroft have explored bandwidth-delay constrained source routing while Guerin-Orda [7] have looked at bandwidth constrained source routing. Ma-Steenkiste [8] have presented a simulation study of QoS routing for traffic requiring bandwidth guarantees. Wang-CrowCroft have also proposed distributed hop-by-hop routing algorithms. VS is our approach to solving the problem of QoS routing.

Our earlier work on VS routing [9] [10] [11] has shown that the VS routing scheme demonstrates good scalability, load balancing capability and dynamic adaptability in intra-domain routing. Here we extend VS routing algorithm from hop based routing to delay based routing. We address both intra-domain and more importantly, inter-domain QoS routing. In multi-parameter QoS routing we consider constraints based on delay (additive parameter) and bandwidth (non-additive parameter). The efficacy of our heuristic algorithm is demonstrated by simulation studies on relatively large networks with 10 and 25 administrative domains with 10 and 25 nodes each. The results for both intra and inter-domain QoS routing look equally impressive under different constraints. Unlike other algorithms, which trade off between scalability and performance, VS has no such issues. Not much research has taken up such large networks for studying QoS routing, one reason why we could
not give head to head comparison with other heuristics. In addition, VS aims at finding multiple paths under the user-specified multiple constraints.

The rest of the paper is organised as follow. The basic hierarchical network structure is defined in section II. In Section III, we extend VS embedding to enable hop-based and delay-based inter-domain routing. Section IV explains hop and delay based end-to-end routing and gives the simulation results. Section V describes the QoS routing approach under multiple QoS constraints (bandwidth and delay) and gives the simulation results. Section VI summarizes the complete research findings.

II. Hierarchical Network

The two-level hierarchy of the Internet provides the scalability and flexibility to connect diverse domains. At the lower level are the different autonomous systems each consisting of a large number of nodes. Different ASes are connected to each other through border gateway nodes. The border nodes participate in the information exchange for the inter-domain routing and set-up the inter-AS paths. This is known as the upper level or inter-domain routing level. The route selection from any given source node in a source AS to a given destination node in a destination AS also takes place at two levels. First, at the upper level an AS path is selected based on inter-domain routing and then, at the lower level intra-domain paths are selected within each AS. This two-level path selection from source S to destination D is shown in Fig. 1 for a network with 3 ASes.

![Fig. 1. Illustration of hierarchical routing in a network with 3 ASes showing the lower level intra-domain routing and the upper level inter-domain routing.](image)

The end-to-end path may traverse several ASes and consists of a series of intra-domain and inter-domain links. Within each domain, paths are selected between the ingress and egress border nodes (or the source node to border node or border node to the destination node) using domain-specific IGP routing policies. The intra-domain and inter-domain routing schemes can be different due to the hierarchical nature of the Internet. They act independent of each other and share only the relevant information. This decomposition of the end-to-end routing problem enables each domain a right to maintain privacy and independence of its internal network operation and control over the information exchange.

We initially performed our simulations on a 10 AS network, each AS having 10 nodes. After successful routing in this network, we decided to work with larger networks: 25 ASes, each AS having 25 nodes with arbitrary topologies. Each AS has 3 border nodes that are chosen from the high connectivity nodes, as a likely situation. Thus, the lower hierarchical level of the network has a total of 625 nodes split into the 25 ASes and the upper level consists of only 75 border nodes. For simplicity, all the inter-AS links are considered bi-directional and not dependent on business relationships between the ASes. The size of the network and the variations in network topologies, link parameters, and connection requests provide the requisite diversity of situations for our simulation study. We generate a series of connection requests, each from an arbitrary source node and source AS and to an arbitrary destination node and destination AS. We use different QoS parameters for individual links and connection requests. We have extended the VS routing algorithm presented in [9], [10] and [11] to carry out both, hop and delay based intra-domain and inter-domain routing. We also address the complex problem of multi-parameter based routing by extending the VS routing to include path selection based on delay and bandwidth, together.

III. Extension of VS Embedding

An essential step in VS routing involves an embedding process that transforms the network topology information into a multi-dimensional virtual space (VS) configuration. The VS configuration is a highly concise representation, with excellent directional property. A directed VS distance from any node to any destination in the VS configuration indicates available path options with the least or low number of hops. This enables simple geometric routing that is fast, simple, and efficient.

VS embedding uses an equivalent multi-body system based on the network topology. VS configuration is formed by an iterative process under the action of three types of VS forces acting on the participating nodes. The multi-body interaction process leads to minimization of the total energy of the system which automatically ensures maximization of the directivity property. Detailed treatment of VS routing is presented in [9].

Earlier work on VS routing was based on choosing single or multiple paths with the least number of hops. Here, we have extended the VS embedding step to generate the VS configuration for delay based routing. As such any additive parameter, such as hop, delay, or cost, can be used for VS embedding. We further take into account inter-domain routing with multiple border nodes in each AS forming the upper hierarchical level. We have carried out (intra-domain and inter-domain routing) simulations by using VS routing in both cases.

The basic nature of the three types of VS forces remains same as the earlier work [9] and the VS embedding process is also along similar lines. The definitions of the VS forces are modified to achieve the added functionality of delay based and inter-domain routing. The forces are defined as follows:
Spring force on directly connected nodes: This is a spring-type force $F_{1ij}$ that tries to keep the distance, $d_{ij}$, between two directly connected nodes, $n_i$ and $n_j$, close to $D_{1ij}$, the ideal VS distance between $n_i$ and $n_j$. The force is attractive if the distance between the nodes is more than $D_{1ij}$ and repulsive if less than it. In the hop-based intra-domain VS routing [9] each hop ideally corresponds to one unit distance in the VS space and thus, $D_{1ij} = 1$. For delay based routing, the ideal VS distances between directly connected nodes are discretized to multiple values, based on the propagation delay on each link. In inter-domain routing the discretized values of ideal VS distances between two border nodes of the same AS, and that belonging to different ASes, are discretized differently as $D_{1aij}$ and $D_{1bij}$ respectively. The forces are as follows,

$$F_{1aij} = |F_{1aij}| = k_{1a} \cdot \log(d_{ij}/D_{1aij}),$$  

$$F_{1bij} = |F_{1bij}| = k_{1b} \cdot \log(d_{ij}/D_{1bij}),$$

with $k_{1a}=k_{1b}=10$, chosen after detailed optimization.

**Repulsive force on the nodes not connected directly**: This force, $F_{2ik}$, acts on nodes $n_i$ and $n_k$ that are not directly connected to push them apart along the VS distance vector $d_{ik}$. This force falls with the VS distance between the two nodes and increases with the the least number of hops $N_{ik}$. $F_{2ik}$ is defined as follows,

$$F_{2ik} = |F_{2ik}| = k_{2} \cdot (N_{ik} - 1)/d_{ik},$$

with $k_2=0.05$, chosen through optimization.

**Random kick force**: This is a small random force, $F_{3i}$, with arbitrary amplitude and direction, which kicks each node $n_i$ independently in the multi-dimensional virtual space.

The total force, $F_{t_i}$, acting on the node $n_i$ is given by the vector sum of the three types of VS forces as,

$$F_{t_i} = F_{1i} + F_{2i} + F_{3i}.$$  

The VS embedding involves evolution of the VS configuration in an iterative process till the configuration stabilizes. At every step each node $n_i$ moves by a small distance $\Delta_i$ given by,

$$\Delta_i = k_3 \cdot F_{t_i},$$

with $k_3=0.1$

A deviation parameter, $\delta_{ik}$ indicates normalized difference between the ideal VS distance $N_{ik}$ and present VS distance $d_{ik}$ for a node pair, $n_i$ and $n_k$ given by,

$$\delta_{ik} = |(d_{ik} - D_{ik})|/D_{ik}.$$  

Here the ideal VS distance $D_{ik}$ is the sum of individual ideal VS hop distances $D_{ij}$’s along the shortest path. An average value of all $\delta_{ik}$ values in the VS configuration is defined as $\delta_{avg}$. This parameter is equivalent to the average delta parameter defined in [9] for the hop-based VS routing. The $\delta_{avg}$ decreases with the evolution of the VS configuration and then stabilizes to a low value, typically around 0.20. A low value of $\delta_{avg}$ indicates very good directionality property of the VS configuration with respect to the embedding parameter (hop, delay, or cost). $(1 - \delta_{avg})$ is a measure of effectiveness of the VS embedding for geometric routing. The embedding process is complete after $\delta_{avg}$ reaches a stable, low value.

The static network topology information is captured in the final VS configuration in the geometric form. A path selection in VS routing is based on combining the static costs based on the VS configuration and the dynamic link and node costs based on the link utilizations or failures [9]. Here,

$$C_{ij} = C_{1ij} + C_{2ij} + C_{3ij},$$

where $C_{ij}$ is the total cost of the outgoing link from $n_i$ to $n_j$. The static cost of the link, $C_{1ij} = 0.5 \cdot (1 - \cos \theta_{ij})$, depends on the angle, $\theta_{ij}$, the link subtends with the destination vector in the VS coordinate space. The dynamic link cost, $C_{2ij}$ depends on the utilisation of the link, and node cost, $C_{3ij}$, is the average of all links starting from the node $n_j$. The set of links with the lowest total cost is selected for VS routing.

IV. HOP AND DELAY BASED END-TO-END ROUTING

A. Hop based intra-AS and inter-AS VS routing

Hop-based intra-AS VS routing has been successfully demonstrated earlier. We concentrate here on adapting the VS routing scheme to perform inter-AS routing. The ideal hop distance $D_{1aij}$ between the border nodes within an AS has values of 1, 2 or at the maximum 3 hops. In order to give higher weightage to each inter-AS hop, the value of $D_{1bij}$ is fixed at 4. This serves two purposes: Firstly, the path-selection then tends to minimize inter-AS hops rather than intra-AS hops. Secondly, each individual AS with its border nodes stays together as one unit and has no or little overlap with the neighboring ASes in the VS configuration.

The VS embedding of individual ASes resulted in deviation delta parameter values of 0.15 or lower for 25-node networks with arbitrary topologies. The intra-AS VS routing successfully found intra-AS path 100% of the time. The VS embedding of the upper hierarchical level of 75 border nodes required 500 iterations to stabilize to a delta value of 0.21. End-to-end VS routing simulations were carried out by selecting 5000 random requests in the whole network. It resulted in successfully finding the intra-AS and inter-AS path 99.8 % of the time. In the 10 AS network taken up for study initially, almost 100% of the requests got serviced. A comparison of the average hop distance of an inter-domain path using VS routing and OSPF is given in Table I. The average hop distance for VS routing is only 3% higher. It shows that VS routing closely matches in performance even though, it requires significantly less information.

Another important aspect of providing QoS in inter-domain routing is to be able to find back-up path for protection against any link failure. In our simulations, we send five connection request packets from a source to a destination to try and find 5 different paths. From these five paths, we choose the path with the least number of hops as the primary path. An alternate or back-up path is chosen that has minimum overlap with the links along the primary path. In our 25 ASes simulations, the
average hop distance along the back-up path is about 30% higher as given in Table I. The examples of the primary and back-up paths for 2 sample requests are given in Table II and paths for 2 pairs are shown in Fig. 2 in the upper hierarchical level for our sample network. The ease of finding back-up paths using VS routing is a significant advantage.

We randomly break 5% of the links and carry out simulations of inter-domain routing using the original VS configuration (the one without any link failures). The VS routing was able to serve 98.2% requests under such conditions. Without the full information the failure rate for OSPF would be very high, of the order of (average hop distance X link failures). Thus VS routing is very robust under arbitrary link failures.

B. Delay based intra-AS and inter-AS VS routing

For real-time applications, end-to-end path delay is one of the most important QoS parameters. We have extended VS embedding to enable delay based inter-domain routing. Each delay unit (expressed as VS distance) used in the VS embedding may correspond to 5 or 10 ms of link or path delay. Compared to the hops that have only integer values, the delay has more variability and has to be discretized in finer values. We assume that paths internal to an AS have smaller delays (or weighted delays) compared to the inter AS path delays. The delay along the intra-AS links is discretized to 0.2, 0.6, 1.0, 1.4, and 1.8 delay units and that for the inter-AS links is discretized to 3, 3.5, 4, 4.5 and 5 delay units.

The path selection using intra-AS delay based VS routing achieved excellent results similar to hop based VS routing. However, the delay based VS embedding for the inter-domain routing is more complicated since the ideal VS distances between border nodes within an AS, $D_{1aij}$, and those between different ASes, $D_{1bij}$, can have many discrete values. In order to improve the effectiveness of VS embedding, a two-step evolution process is chosen. In the first step, the value of $k_{1a}$ is kept at 10 and $k_{1b}$ and $k_{2}$ are set to zeros. This results in quick stabilization of the intra-AS VS distances between the border nodes. In the second step, the values of $k_{1a}$, $k_{1b}$, and $k_{2}$ are 20, 10, and 0.022, respectively. The doubling of $k_{1a}$ results in keeping the intra-AS distances relatively unchanged while the inter-AS distances undergo the iterative evolution. Again, the constants are optimized by using large set of network topologies and parameters to give the best results in terms of delta. In the initial 10 AS network taken up for study, we found a success rate of nearly 100%. For the 25 AS network; a final delta of 0.232 was achieved after 700 iterations. This resulted in the success rate of 99.7 %, where

$$\text{Success rate} = \frac{\text{Number of realized connection requests}}{\text{Number of feasible connection requests}} \tag{8}$$

The delay based inter-domain VS routing results closely match with the OSPF results as shown in Table III. The average delay for inter-domain path using VS routing is only 5 % higher than the OSPF path eventhough, it requires significantly less information. VS routing also provides a back-up path that has an average delay 16 % higher and has low overlap with primary path.

V. TWO QoS PARAMETERS

A user might have two or more QoS requirements with constraints of different types. One type of constraint could specify path boundedness, e.g. “The end-to-end delay between node 5 and node 34 should be less than 30 delay units”. A second type could require path optimization, e.g. “The selected

---

**TABLE I**

<table>
<thead>
<tr>
<th>Path selection</th>
<th>Average number of hops traversed on this path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main VS path</td>
<td>5.85</td>
</tr>
<tr>
<td>Back-up VS path</td>
<td>4.76</td>
</tr>
<tr>
<td>OSPF path</td>
<td>3.75</td>
</tr>
</tbody>
</table>

**TABLE II**

<table>
<thead>
<tr>
<th>Src</th>
<th>Dest.</th>
<th>Path selected</th>
<th>AS Hops</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>62</td>
<td>28</td>
<td>62-63-45-30-28</td>
<td>2</td>
<td>Main path</td>
</tr>
<tr>
<td>62</td>
<td>28</td>
<td>62-61-4-11-28</td>
<td>3</td>
<td>Back-up path</td>
</tr>
<tr>
<td>46</td>
<td>11</td>
<td>46-13-14-4-11</td>
<td>3</td>
<td>Main path</td>
</tr>
<tr>
<td>46</td>
<td>11</td>
<td>46-47-51-72-60-12-11</td>
<td>4</td>
<td>Back-up path</td>
</tr>
</tbody>
</table>

**TABLE III**

<table>
<thead>
<tr>
<th>Path selected</th>
<th>Average delay units along the selected path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main VS path</td>
<td>9.00</td>
</tr>
<tr>
<td>Back-up VS path</td>
<td>12.41</td>
</tr>
<tr>
<td>OSPF path</td>
<td>8.23</td>
</tr>
</tbody>
</table>

Fig. 2. Illustration of hop based inter-AS VS routing. The main path and back-up VS paths are shown for two different requests.
path must have the maximum bottleneck bandwidth or least number of hops amongst all feasible paths”. As seen from the previous section, a VS configuration can be constructed for any additive type QoS parameter, such as hop, delay, or cost, etc. In case of two additive QoS parameters two different VS configurations can be constructed, one for each parameter. The criticality of one constraint over the other could determine which configuration is used for primary path selection.

However, when one of the QoS parameters is non-additive, such as bandwidth, the path selection approach changes significantly. For e.g., the Wang-CrowCroft algorithm for bandwidth-constraint source routing prunes all links which cannot satisfy the bandwidth requirement for a given path setup request. It then finds the shortest delay path [2] [12].

We consider a fully distributed QoS routing approach based on VS routing for a scenario in which a user has both delay and bandwidth requirements. By comparing the VS distance between the source and the destination in a delay based VS configuration a probable delay can be estimated. This primary check is done without actually finding the path. When the maximum delay requirement is less than 80% of the estimated delay then it can be concluded with 92% confidence level that the request cannot be satisfied.

For finding a path based on both delay and bandwidth constraints, we have incorporated a pruning based technique [2] [7] into the VS routing scheme. We first prune all links which cannot satisfy the bandwidth requirement by making the link cost in (7) very high. We then find multiple paths using delay based VS routing and then choose the widest path, i.e. the maximum bandwidth path, that satisfies the delay requirement. Fig. 3 gives an illustration of this for the same source-destination pair as that picked for hop-based routing in Fig. 2. Here, some low bandwidth links have been pruned and therefore the paths chosen in Fig. 2 are not feasible.

On the 25 AS network, we try and setup paths for different requests from a random source to a random destination. The desired bandwidth request is randomly selected from 0.0 to 1.0 and the desired delay limit is also randomly selected between 10 units to 40 delay units. Various links in the network are initialized randomly to three available bandwidth levels of 0.5, 0.75 and 1.0, implying the utilisation levels of 50 %, 25% and 0% of the total bandwidth. The following two approaches are proposed for solving the problem:

**Bandwidth based 3 VS configurations:** We assume that the available information is broadcasted to every node in the network. We consider three different network topologies after bandwidth based pruning, corresponding to 3 bandwidth levels. We thus form three different VS configurations for each topology using delay based VS embedding. Depending on the user bandwidth request, we can switch to the appropriate configuration to find delay based paths. We considered 5000 random requests and found feasible requests with a success rate of 94.0%. The 6% unsatisfied requests violated the delay constraint only by a small fraction. This approach works very well when the available link bandwidths are discretized to a few levels and information is updated periodically.

**Single VS configuration:** When the updated link bandwidth information is not available throughout the network then this routing approach based on a single VS configuration and distributed path selection is very useful. Here we assume that only connectivity and propagation delay information is available at all inter-AS nodes. VS is thus expected to perform under limited visibility. In this approach a single VS configuration of the original network is used for delay-based routing. However, while performing local path selection; the individual ASes select paths along links which can satisfy the bandwidth constraint. We again send 5 packets from the source to the destination. For 5000 requests with random delay and bandwidth requirements, the results are as follows:

- Out of 5000 requests only 3100 requests are feasible
- 2825 times (91.1% success), the requests are satisfied
- 204 times (6.6%), the delay constraint is violated by a small fraction
- 71 times (2.2%), the paths are not found

Thus for 91% of the times, we succeeded in finding such a path that meets both delay and bandwidth constraints.

### A. Dynamic routing

Next, we look at a dynamic scenario of successive connection requests and updating of the link bandwidths based on the paths selected. We start with the same network topology and with all link utilizations set to zero. We try and setup 600 connection requests from random source-destination pairs and randomly chosen constraints. The links in the inter AS network are assigned bandwidth 3.4 and 5Mbps. The maximum delay on an inter-AS link is around 5 delay units. The requests are typically in the range of 0-0.3 Mbps. For the requests, the desired end-to-end delays is between 20-40 delay units. The available link bandwidths in the network are updated with

---

**Fig. 3. Illustration of bandwidth pruning in the inter-AS network topology and a sample path.**
every feasible path that is found. The throughput achieved using VS routing is plotted in Fig. 4, where
\[
\text{Throughput} = \frac{\text{Number of realized connection requests}}{\text{Total Number of connection requests}}.
\]

For comparison we have run simulations using CSPF for the same set of requests under what could be considered as ideal conditions. CSPF updates all bandwidth information every 80 requests and the information has instant convergence. The results are plotted in the same figure, Fig. 4. We also plot the behaviour of CSPF (with continuous information update) in the same figure. This gives us an indication of the number of feasible requests at any instant of time. When around 200 requests have been serviced, the average utilisation of the network is around 0.20 in all the three plots.

However, the assumption of instant convergence while performing CSPF routing is not valid. In fact, some of the problems BGP commonly suffers from are information scaling and long convergence times following link failures [13]. The damping mechanisms for route flapping, filtering and selective information advertising have been applied to bring about faster convergence and as a consequence of these efforts, BGP convergence times have improved from around 5 minutes to 1-1.5 minutes. Under practical scenario of periodic and triggered updates of link bandwidths with modest link loading the CSPF success rate has been reported to be 85% [14]. As mentioned earlier, the implementation of this in inter-domain routing would be very difficult due to the need for full information exchange and issues related to scalability and convergence.

VS routing on the other hand, works with limited visibility and does not require bandwidth updates and information convergence (primarily due to local path selection). Another implementation advantage is that ASes do not need to share the information and thus they maintain their privacy.

VI. CONCLUSIONS

We have adapted VS for constrained based inter-domain path selection. We have simulated hop-based and delay-based VS routing for a network with 25 ASes and 625 nodes. The success rate for finding feasible paths is 99.7 % for both. We have looked at back-up path selection and robustness to arbitrary failures. The VS routing scheme can also be used for selecting paths under two or more additive QoS parameters.

For path selection under bandwidth and delay constraints we have incorporated bandwidth based pruning into VS routing. The routing scheme is analyzed under both static and dynamic network scenarios and shows excellent success-rate in finding feasible paths meeting the requests with bandwidth and delay requirements. The routing scheme can work well for distributed path selection under limited information visibility. It is highly scalable and does not have extreme requirements of convergence and information integrity.

Significantly, VS is able to do multi-path selection. Recent research is looking at extending BGP to multi-path routing rather than just forwarding of requests as employed in IP routers today. VS is the right scheme for finding multiple feasible paths. This can easily facilitate inter-domain routing and traffic engineering to support QoS over the internet.

REFERENCES