Inter-Domain Routing with Multi-Dimensional QoS Requirements [*]

Lotfi Benmohamed, Bharat Doshi, Tony DeSimone, Robert Cole
Johns Hopkins University Applied Physics Laboratory
Laurel, MD

ABSTRACT

External Border Gateway Protocol (eBGP) is the ubiquitous protocol used today for advertisement of reachability information and for route selection among administrative domains (Autonomous Systems or ASs) in the Internet. However, many emerging needs in commercial and military networking have exposed limitations of the current eBGP. In particular, these IP networks of the future will carry a very diverse mix of applications, with very diverse Quality of Service (QoS) requirements (in the broad sense of the phrase). Some of these networks also have a very diverse set of component networks (wireless and wireline, fixed and mobile with different degrees of mobility, long lived and short term ad-hoc) and some of the component networks may be very dynamic in their service capabilities. These scenarios call for enhancing eBGP to allow for multi-topology and QoS-aware routing, using several QoS metrics in decision making. In such an enhanced BGP, more than one route (or path vector) may be propagated in BGP_UPDATE messages, helping optimize with respect to different QoS metrics as needed by different traffic types.

In this paper, we discuss such an enhancement of eBGP. We develop details of advertisements, route thinning, and route selection needed to make the eBGP multi-topology and QoS-aware in the sense described above. We take the eBGP and internal BGP (iBGP) advertisement and route selection process and identify any modifications needed at each stage. We also discuss interactions between eBGP and iBGP and between BGP and the Interior (intra-domain) Gateway Protocol (IGP) needed to make the approach work end-to-end. We also discuss alternative ways to ensure that packets follow the selected end-to-end routes (both within and between domains). In particular, the potential uses of MPLS, source routing, tunneling, and DiffServ/ToS bits for this purpose are discussed in the paper.

INTRODUCTION

External Border Gateway Protocol (eBGP) is the ubiquitous protocol used today for advertisement of reachability information and for route selection among administrative domains (Autonomous Systems or ASs) in the Internet. An inter-domain routing protocol needs to scale to the entire Internet, which requires that it be designed with minimal overhead traffic among administrative domains, and with minimal requirement on information storage. Moreover, it needs to account for routing policies and restrictions imposed by commercial relationships among Internet Service Providers and between providers and their customers. Finally, the reluctance of competing service providers to share details of their network internals with others further limits the quantity of information that can be distributed by the protocol. Thus, eBGP is a path vector protocol. The only information passed from an AS to its neighbors is the set of destination network prefixes reachable from that AS (and for which this AS and all intervening ASs are willing to provide transit service) and, for each such reachable destination network, the sequence of ASs involved in the route (this helps avoid routing loops). At most one route is advertised from an AS border router to any given prefix. An AS border router receiving these advertisements (BGP updates) from its neighbors (both external and internal BGP neighbors), applies its own routing policies, and then selects from among different ways (via different neighbors) to reach a given prefix. It selects one route and advertises it to its neighbors. Interior BGP (iBGP) and the Interior Gateway Protocols (IGPs) partner with eBGP in establishing end-to-end routes. Information sharing within an AS has less restrictions and this is reflected in the design of IGPs, which tend to be link state based and advertise entire network topology, and in the flexibility available in iBGP.

Many emerging needs in commercial and military networking have exposed limitations of the current eBGP. In particular, these IP networks of the future will carry a very diverse mix of applications, with very diverse requirements on the Quality of Service (QoS). QoS itself is taking on a much broader meaning than the IP standards have dealt with to date, including, for example, packet delay and losses, session set up times, session success rates, availability, security, time to reconfigure and time to provision, etc. Some of these networks may also have a very diverse set of component networks (wireless and wireline, fixed and mobile with different degrees of mobility, long lived and short term ad-hoc) and some of the component networks may be very dynamic in their service capabilities. Thus, different end-to-end routes between the same end points may offer very different QoS capabilities and these may vary with time. As a result, having the capability to use the QoS (in the broad sense discussed here) requirements in routing, session admission, packet scheduling, buffer management, service restoration priority, degree of security protection, and similar other decisions will become an important need in the IP networks of the future, especially in networks like the GIG.

The above need calls for enhancing BGP to allow multi-topology (exposing multiple routes) and QoS-aware routing, using several QoS metrics of importance to different applications. In such an enhanced BGP, we allow for more than one route to be propa-

[*] For further information send correspondence to L. Benmohamed, E-mail: Lotfi.Benmohamed@jhuapl.edu

gated in BGP_UPDATE messages and the information propagated involves more QoS metrics than a simple sequence of ASs. Of course, we still need to keep the information transfer to the minimum needed to maintain the scalability and privacy attributes of the protocol. Note that the approach we discuss here in support of inter-domain QoS routing introduces enhancements to BGP but without any change to the inter-domain routing architecture that BGP is based upon. However, because of other BGP limitations in addition to the ones mentioned here, there are proposals for inter-domain routing that adopt a different architecture. In [1][2] the authors propose a new architecture where an AS is represented as a single logical entity and the responsibility for the exchange and selection of inter-domain routes is moved out of the routers and into a separate platform (Route Control Platform) acting as a control entity on behalf of the AS.

There has been recent work on enhancing BGP to make it QoS aware [3]. The enhancements are achieved by including a QoS attribute in BGP UPDATES and in route selection. Most of this work focuses on a single QoS metric and hence supports one route, as in the current BGP. This route can be optimized with respect to one selected QoS metric. In [4] this approach was used to compute paths with maximum available bandwidth. In an earlier paper [5], we presented simple examples illustrating the need for multi-route (multi-topology) and QoS aware BGP. We also defined the concept of non-dominated routes and used this to trim the number of routes to be propagated. In this paper, we develop details of advertisements, route trimming, and route selection needed to make BGP multi-topology and QoS aware in the sense described above. We identify modifications needed at each stage in the BGP advertisement and route selection process. We also discuss interactions between eBGP and iBGP (e.g. local preference and inter-domain metrics, which domain gateway router to use to exit from an AS) and between BGP and IGP (e.g., importing BGP routes and associated QoS attributes in the IGP) needed to make the approach work end-to-end. Finally, we discuss alternative ways to ensure that packets follow the end-to-end routes (within and between domains). In particular, the potential uses of MPLS, source routing, tunneling, and DiffServ/ToS bits for this route pinning purpose are discussed. Finally, we discuss similarities and differences among commercial Internet, commercial Intranets, and DoD networks like the GIG in terms of the network scale and complexity, relationships among domains, and factors limiting information exchange. This discussion helps evaluate the relative ease with which the suggested enhancements can be implemented in different environments.

BGP OPERATION AND INTERACTIONS AMONG EBGP, IBGP, AND IGP

In this section, we briefly describe the baseline BGP operation and interactions among eBGP, iBGP, and IGP, leading to the end-to-end routing capability in the Internet. We also summarize key features and limitations. In the next section we will discuss the changes needed to this baseline to realize the multi-topology and QoS-aware BGP.

eBGP runs between Border Routers in separate Autonomous Systems (ASs). These routers are called eBGP peers (neighbors). iBGP runs between Border Routers within one AS. These are called iBGP peers. eBGP neighbors are typically directly connected while iBGP peers need not be. Information exchange between eBGP or iBGP peers is in the form of BGP UPDATE messages which consist of: (1) Network Layer Reachability Information (NLRI), which is a list of address prefixes for the networks that can be reached from a given Border Router (BR) and about which this router wants to inform its neighbor BRs; and (2) a number of path attributes that include AS_path which lists the sequence of ASs to reach the destination (provide capabilities of detecting routing loops), these attributes provide flexibility to enforce local and global routing policies which allow the control of which routes to accept, prefer, or pass on to other BGP peers. When the NLRI changes (e.g. network becomes unreachable or a better path springs up), a new UPDATE message is generated to withdraw invalid routes and/or inject new routing information. When a border router receives an UPDATE message, it uses the received information to decide its own inter-domain routing choice for destinations listed in the UPDATE message. Path attributes in the update message can be manipulated before the decision about route selection is made and also before a route is forwarded to a BGP neighbor. Attribute manipulations include:

- AS_path (sequence of ASs to be visited in the path to destination) is manipulated to affect inter-domain routing behavior. In particular, it allows an operator to configure artificial AS number insertion so that path length is increased and route selection affected (typically repeat own AS number as many times as needed in iBGP exchanges, inserted AS numbers are removed when advertising to outside via eBGP) [6].
- Local_Pref (local preference) is used to set a preference for the exit point of an AS to reach certain destinations. Local_Pref is not passed between ASs but used internally within an AS to assign a local degree of preference.
- Multi-Exit Discriminator (MED) to help outbound decision of a neighbor AS (Local_Pref influences own outbound decision), especially useful when a AS peers with more than one eBGP neighbors in the same neighboring AS.

Origin_Type in another path attribute, which is set by the AS generating the route, and identifies how the AS learned about the destination being advertised in the NLRI. The value that this attribute can take is either IGP (NLRI internal to the end AS), EGP (learned via Exterior Gateway Protocol), or INCOMPLETE (learned by other means).

Next we discuss the actions taken by a BGP border router when it receives UPDATEs from its neighbors. The decision process is depicted in Figure 1.

Routes received by the BR from neighbors without loops are sent to the policy filter (routes containing loops are detected by the presence of own AS number in the AS_path and are disregarded). Let S_0 represent the subset of all received routes to D that make it through the policy filter. A single best route out of S_0 is selected based on the attribute values as follows. Starting from the set S_0, the algorithm builds a sequence of sets S_i (with S_i contained in S_{i-1}) as listed in the following 7 steps, and terminates as soon as a set S_i contains a single route (in which case the process terminates at step *i*, *i*=1,2,…,7):

- S_1 is the set of routes in S_0 with maximum value of Local_Pref attribute
- S_2 is the set of routes in S_1 with minimum AS_path length
- S_3 is the set of routes in S_2 with lowest Origin_Type (with IGP < EGP < INCOMPLETE)
- Let S_{3j} be the set of routes in S_3 that are advertised by neighbor AS *j* (S_3 is the union of S_{3j}), and S_{4j} be the set of routes in S_{3j} with minimum value of MED, then S_4 is the union of S_{4j}. Note that MED values can only be compared among routes from same neighbor AS.
- If there is at least one route in S_4 which was received from a eBGP peer, remove all iBGP routes from S_4, the resulting set is S_5, otherwise S_5=S_4.
- S_6 in the set of routes in S_5 with minimum IGP distance to the BGP next_hop attribute (typically the iBGP BR peer).
- S_7 is the one route in S_6 announced by the peer with lowest router ID.

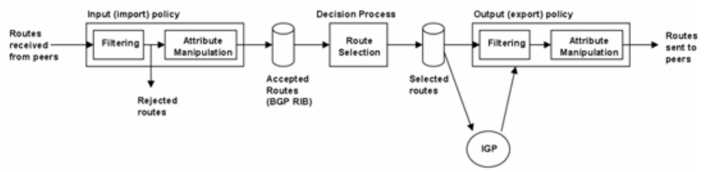


Figure 1. BGP update processing

Note that out of all possible routes to a destination prefix D, the AS learns about a smaller subset (at most as many as the number of eBGP links), selects an even smaller subset for egress to D, and advertises at most one of these egress routes to each one of its neighbor ASs.

As mentioned earlier, the route selection process described above can be controlled somewhat. We summarize these controls below so we can compare the baseline process with enhancements to be discussed later. Local_Pref is at the highest level of BGP decision process and is considered before the AS_path attribute. Thus, Local_Pref can be used to force a particular link in S_0 to be the only AS exit link to D (by configuring the Local_Pref for that link higher than for any other link). Among the links with the same Local_Pref, the AS_path attribute is used to down select. Thus, to force a set of K links out of L_0 (with corresponding K routes in S_0) to be egress links, we need to

- assign the same Local_Pref to these links, higher than all other links, and
- make AS_path length for these links the same by configuring an attribute manipulation that appends its own AS number to the route's AS_path attribute as many times as needed (the appended AS numbers are removed when the route is passed to neighboring ASs).

The above forces the decision process to come to step 5 when each of the K border routers will pick its attached link as the exit point. Each one of the other BRs will pick one of these K BRs as its exit router, and serve as an ingress node if it advertises the route to a neighbor AS. The decision process for these ingress routers will come to step 6, and possibly step 7 for tie breaking

In conclusion, if the decision process stops on or before step 4, then only one route is selected AS-wide for all traffic destined to D. If it gets to step 5, there will be more than one exit point, and all routes that pass step 4 will be selected each by its terminating BR at step 5 (these are the egress BRs). Other BRs are potential ingress nodes and will select one egress BR at step 6/7 based on the IGP shortest path.

BGP ENHANCEMENTS

Based on the BGP operation described above, we discuss in this section the protocol limitations and the enhancements needed in support of inter-domain routing with multi-dimensional QoS requirements.

Whereas link state based routing allows every node to have complete network topology information that allows it to examine and determine the feasibility of any path in the network, distance vector based routing protocols like BGP have very limited path visibility. In particular, out of all existing routes to a destination prefix D, an AS has visibility into only a subset of them, advertised to it by some of its neighbor ASs (at most one per eBGP peer, even when the peer AS knows of more than one path to D). Moreover, since reachability is the main concern and not QoS, the only path quality metric available is a rough indication of path length in terms of number of ASs along the route. Note that this attribute has a role in route selection specified by step 2 of the decision process; however when over-ridden by the Local_Pref attribute in step 1 its primary use becomes routing loop detection.

In order to be able to support inter-domain routing with multi-dimensional QoS requirements, the following enhancements are needed:

- BGP neighbors need to be able to exchange more than one route to a destination.
- Exchanged routes need to carry a list of associated QoS attributes.
- Select routes based on QoS requirements.
- Enforce selected route through appropriate end-to-end forwarding.

We now discuss in more detail each one of these enhancements.

Exchanging multiple paths

A BGP update message received from a neighbor AS contains only one route consisting of the address of the advertised destination network D (NLRI) and its associated path attributes, in particular AS_path which lists the sequence of ASs in the selected path from the neighbor's AS to the destination AS where D belongs (this is the path that was selected in a distributed manner by all intervening ASs in AS_path). When multiple paths to D are maintained by an AS in support of QoS routing, a BGP update message will need to list multiple AS_path attributes, one per maintained path. Moreover, the 7-step path selection process described in the previous section will need to be updated accordingly. Specifically,

- Steps 1-4: In today's commercial Internet with best-effort service, heavy use of policies is made to influence routing. As we described above, this includes manipulating attributes such as Local_Pref or AS_path in such a way that only routes from one or some neighbors are enabled. In order to allow QoS routing to be effective by being able to choose among a larger set of routes, ideally routes from all neighbors should be enabled so that QoS path selection can be done at step 5. To allow for routes from each neighbor in a given set (ideally the set of all neighbors) to be considered for QoS routing, interfaces to those neighbors should be configured with an attribute manipulation policy that assigns to routes advertised on those interfaces the same and highest Local_Pref as well as the same AS_path length.
- Step 5: In step 5 of the current decision process each BR that terminates an enabled route (one that passes steps 1-4) selects itself among all enabled routes as an AS egress point. Instead, under QoS routing, all enabled routes to a destination D received at a BR (from both internal and external peers) undergo a QoS-aware route selection process as described below where a set of dominant routes is selected.
- Steps 6-7: In the current selection process, non-egress nodes get to steps 6-7 to select one of the egress nodes from step 5. Under the proposed QoS enhancements, the set of dominant routes selected in step 5 by each BR is maintained instead (this set is used for routing by the BR, and advertised to its neighbors).

Maintaining path QoS parameters

In addition to listing multiple AS_path attributes per advertisement (BGP update message), we need to list the QoS metrics associated with each path. These metrics include well known metrics such as bandwidth, packet delay and delay variation, packet loss, as well as other metrics such as path security, path availability, time to reconfigure, etc. When a BR receives an update message, the path QoS attributes included in the message correspond to the path from the BGP peer to the destination. Therefore the BR needs to first update the QoS attributes by accumulating the attributes of the segment from itself to its peer with those of the segment from the peer to the destination (which is captured in the update message). There are two cases:

- If the BR's peer is an eBGP neighbor, then there's typically one physical link between the BR and the neighbor.
- If the BR's peer is an iBGP peer (belongs to the same AS), then there are typically multiple intra-AS paths with potentially different QoS attributes between the two peers. In this case, issues similar to dominant path selection discussed below arise here in the context of link state routing protocols as well (IGPs are typically link-state-based). This potentially results in paths that have the same AS_path attribute but different QoS metrics due to differences in intra-AS segments.

Different QoS metrics might have different accumulation rules such as the following:

- Additive metrics where the path metric is equal to the sum of the metric values for all segments in the path (delay is an additive metric).
- Multiplicative metrics where the path metric is equal to the product of the metric values for all segments in the path. The composition rule for packet loss rate L of a k-segment path is given by $1\text{-}L=(1\text{-}L_1)(1\text{-}L_2)\ldots(1\text{-}L_k)$ where L_i is the loss rate of the *i*-th segment. Transforming the loss metric L into a metric Q given by $Q=\text{Log}(1/(1\text{-}L))$ makes the new metric both positive and additive ($Q=Q_1+Q_2+\ldots+Q_k$).
- Min metric: a metric follows the Min composition rule if the path metric is equal to the minimum of the metric values for all segments in the path. Bandwidth is a Min metric. Path security can be defined in such a way that it follows the Min rule as well: if each security level is assigned an integer metric such that a higher value corresponds to higher level of security, the path security metric is given by the minimum metric value of all segments in the path.

Finally, we note that some of the QoS metrics are dynamic in nature (i.e., load-dependent such as available bandwidth) whereas others are static (such as path security level). The uncertainty in the dynamic QoS parameters could affect the quality of QoS routing when metric values available at decision time are out of date. Choosing the metric update frequency involves a tradeoff between the accuracy of QoS metrics and protocol overhead. These issues and others related to dynamic routing will be addressed in a subsequent paper. Note also that when only static metrics are used (such as link capacity instead of available bandwidth and propagation delay instead of actual packet delay which includes a variable component) composite path metrics can still be useful in making some routing decision such as the desire to avoid paths with delay longer than some given number. For instance, paths with satellite segments would likely have longer propagation delay and would be avoided in favor of other paths with smaller propagation delay.

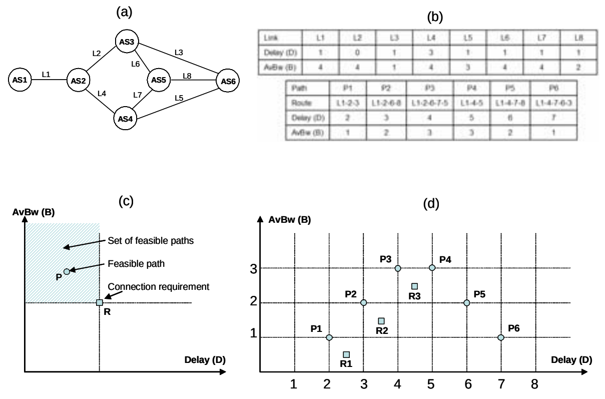
Select dominant routes

For QoS requirements with a single metric such as bandwidth, just knowing a path with the largest available bandwidth is sufficient, since if this path is not feasible no other path is. Thus, in this case we can always find a feasible path when one exists and distance vector protocols can be adapted to compute paths that have a bandwidth property such as shortest-widest (path with largest available bandwidth, with an additive distance metric as a tie breaker to select shortest path among multiple widest paths). However, for QoS requirements with multiple metrics such as bandwidth and delay, there may not be a single "best path." Indeed, consider the network shown in Figure 2.a with 6 ASs and 8 inter-AS links. Assuming just for simplicity that only inter-AS links are limiting (i.e., high available bandwidth and no delay within the ASs), then the bandwidth delay characteristics of all 6 possible paths between AS1 and AS6 are as given in Figure 2.b. For any given AS1-to-AS6 connection request given by the delay-bandwidth QoS requirements vector R=(D,B), a feasible path

P has to have a delay at most D and available bandwidth at least B as shown in Figure 2.c. Figure 2.d shows three connection requirements R1, R2, and R3 with none of the 6 paths being feasible for all three connections simultaneously since P1 is the only feasible path for R1, P2 for R2, and P3 for R3. The shortest-widest path is P4 and is not feasible for any of the three connection requests, and the minimum delay path is P1 and can support only R1. This example shows that a distance vector protocol, with only one route to a destination advertised from each AS border node, cannot support multi-dimensional QoS routing which involves multiple QoS metrics.

Therefore, we need to allow for multiple paths instead of only one to be advertised per destination. The next question is which paths need to be advertised. Fortunately, not all paths will need to be advertised as this would result in an unscalable solution: in a full mesh of N nodes there are $\sum_{i=1}^{N-1}(N-i)$ links but $1+\sum_{k=2}^{N-1}\prod_{i=2}^{k}(N-i)$ different paths between any source-destination pair. We introduce the notion of dominant paths as follows: A path P dominates a set S of paths if it can provide better QoS than any path in S for *all* QoS metrics of interest. If a path P dominates a set S of paths, then paths in S do not need to be advertised as path P is the only one needed to make QoS routing decisions. Indeed, if a connection request cannot be supported by P then no path in S can satisfy the request. Using the example of Figure 2, only three paths (P1, P2, and P3) out of all six paths need to be exposed to AS1. Paths P4, P5, and P6 are covered by P3 since any connection request that can be routed over P4, P5, or P6 can also be routed over P3.

In summary, instead of exchanging one path we need to have each border router exchange the set of dominant paths, and given the dominant paths from all of its neighbors, a border node can derive its set of dominant paths (this can be formally captured in a dominant-path version of the Bellman-Ford algorithm [7]). Note that we need to keep the number of different paths small enough to maintain scalability, and our future protocol evaluations through simulations (as discussed in Section 4 below) will examine this property. We can also place a limit on the maximum number of paths exchanged and examine its performance impact. However, most military networks are significantly smaller than the public Internet and can afford BGP even with many different routes between source and destination.



Link	L1	L2	L3	L4	L5	L6	L7	L8
Delay (D)	1	0	1	3	1	1	1	1
AvBw (B)	4	4	1	4	3	4	4	2

Path	P1	P2	P3	P4	P5	P6
Route	L1-2-3	L1-2-6-8	L1-2-6-7-5	L1-4-5	L1-4-7-8	L1-4-7-6-3
Delay (D)	2	3	4	5	6	7
AvBw (B)	1	2	3	3	2	1

Figure 2. Support for \bandwidth-delay QoS requirements

Enforce selected route

The enhancements described so far allow every BGP BR to maintain the set of dominant paths needed to make QoS routing decisions. The two questions we address in this section are (1) how can IGP routers be made aware of the dominant paths to a destination D, and (2) once a path is selected for a particular traffic flow to D how can packets of that flow be made to follow the selected path.

The answer to the first question depends on whether the IGP is QoS-aware. It also depends on whether each router in the AS runs iBGP (potentially with BGP route reflectors for scalability [8]) to acquire all routes learned by border nodes. We are not making this assumption as this is not typical, but when it is the case then all nodes in the AS including non-border nodes would be aware of all routes learned from outside the AS.

IGP's are typically link state protocols (OSPF and ISIS) and traffic engineering extensions to these protocols are available (OSPF-TE [9] and ISIS-TE [10]) where link QoS attributes can be exchanged as part of link state advertisements. Since we are extending inter-domain routing to make it QoS-aware, it makes sense to assume that the IGP is QoS-capable as well, unless there are significantly more intra-domain than inter-domain resources so that end-to-end QoS is not affected or constrained by intra-domain resources but rather solely dictated by the inter-domain resources.

When the IGP is QoS-aware, instead of just importing in the IGP reachability to D (with an associated single static metric as in OSPF's AS-external-LSA advertisements [11]), each BR can now inject in the IGP one link per dominant route with the link's QoS parameters being those of the dominant path (doing so is equivalent to summarizing to the AS the route information to D in the topology outside the AS). This way any IGP node can select the most appropriate end-to-end path. This can be done as follows: using the intra-domain topology (provided by the IGP) and the additional links to D advertised by border nodes, any IGP node has enough link state information and can use a link state based QoS routing algorithm to select a path to D. Note that border nodes can use the same approach to compute dominant paths: being part of the IGP they have internal topology information, and through BGP (eBGP and iBGP) they acquire external routes. Although dominant path selection can be used by non-border nodes as well, a feasible non-dominant path can be selected instead [7]. Since scalability of IGP routing might be a concern if routes to all destinations are advertised in the IGP, in particular for large inter-domain networks, it can be mitigated by importing only the main destinations for which QoS routing can be most beneficial (reachability to other destinations would rely on best-effort routing).

Once a path is selected for a particular traffic flow to a destination prefix D, the second question is how can packets of that flow be made to follow this selected path. The answer to this question depends on the packet forwarding model available: either explicit path forwarding (EPF) or hop-by-hop forwarding (HHF). EPF refers to either connection-oriented MPLS forwarding with LSPs that are signaled or to IP source routing where the explicit path is inserted in the header of each packet. In HHF the source node cannot specify or control the path. We distinguish the following three scenarios: (A) end-to-end EPF, (B) AS-level EPF, and (C) HHF.

Scenario A assumes the existence of MPLS end-to-end across all ASs. In this case, the source node signals the selected path which is made up of both an intra-domain segment from the source to a border node of the source AS and a sequence ASs leading to the destination. Since there may be different paths that correspond to the same sequence of ASs (such as due to different intra-AS paths, or different inter-AS links when two ASs are interconnected with more than one link), a path-ID is needed to differentiate between these paths that have the same AS_path attribute. When a border node advertises a set of dominant paths to a neighboring AS, it also advertises a path-ID associated with each dominant path. These path-IDs are local to the border node and need not be global. When BR *i* in AS *j* learns about routes to a destination prefix D from internal and external neighbors and selects a set of K dominant paths, it maintains the following information: the intra-AS path to the egress BR in AS *j* (if the selected path was advertised to BR *i* by an iBGP peer), the path-ID as advertised by the BR in the neighboring AS, and a unique path-ID locally assigned from the set {1,2,…,K}. Therefore, the details of the end-to-end path are distributed among the ingress nodes to each AS in the path with each ingress BR maintaining path details of the segment within its AS. When a source node wants to setup a path, it signals the path as a sequence of intra-domain nodes leading to an egress node in its AS, along with the path-ID as advertised by the BR in the neighboring AS. When this BR receives the signaling message, it can use the path-ID to lookup the details of the intra-AS path as well as the path-ID in the next AS along the path to the destination. This process continues as many times as ASs in the path. Once an end-to-end LSP is setup (control path), the task of the data path is simply to map data packets to the LSP.

Under scenario B, paths can only be nailed down within each AS as opposed to end-to-end as in scenario A. Consequently, data packets will need to carry the path-ID so that intra-AS path information can be retrieved when the packet enters an AS, and this path-ID will need to be updated at each AS egress BR when the packet is being sent to the next AS on the path. Some bits from the packet header (such as the DiffServ DSCP bits) can be borrowed for this purpose (two bits would allow each BR to maintain four paths). Note that this scenario involves a change in the IP forwarding paradigm because of the new path-ID handling required.

In the absence of any explicit path forwarding, as under scenario C, packets will follow the hop-by-hop forwarding of the IGP, typically the shortest path to an AS egress node. In this case, one approach to direct the traffic to a specific egress node in the AS is to use IP tunneling techniques such as GRE (generic routing encapsulation) and IP-in-IP. However, the paths followed by these tunnels cannot be arbitrarily controlled as they will follow the IGP. Another potential approach is the use of the type-of-service (ToS) routing capability of the IGP (OSPF and ISIS) to direct traffic to specific egress points. In this case every BR with egress paths to a destination D would inject reachability to D in a different ToS (a ToS assignment rule will be needed for one-to-one mapping of egress node to ToS value). If a node needs to exit the AS through a BR, it needs to populate the ToS value in the IP packet with the one corresponding to the exit BR before handing the packet to its routing module. The routing function maintains one routing table per ToS value for hop-by-hop forwarding. Note that this approach is less attractive than the tunneling solution, in particular due to the fact that ToS-based routing has been removed from the OSPF version-2 [11] specification due to very little deployment of ToS-based routing. Note that there are attempts to develop mechanisms for explicit path forwarding on top of hop-by-hop forwarding. In [12] the authors propose an approach that requires changes to IP packet header by inserting two new fields for holding path IDs (32 bits each), one for intra-domain and one for inter-domain, with associated changes to the forwarding mechanism. The path IDs are computed as a hash of the sequence of nodes or ASs in the path. Since the output of the hash function is not unique, the authors show that it has a low collision probability.

In summary, we have tried to list in this section a number of options for pinning traffic to a particular path. Obviously these options have different degrees of viability. We did not get into the QoS control aspects including scheduling, buffer management, and admission control. We conclude this section with a particular solution tailored to supporting the DiffServ architecture. In this case we need to compute and maintain paths for specific DiffServ classes as opposed to all dominant paths, and each border node would maintain one path per DiffServ class. The DiffServ code point (DSCP) values would be used as global path labels instead of path-ID introduced above. In particular, label manipulation at ingress and egress border nodes is no longer needed.

MODELING AND SIMULATION

In this section, we discuss our work in developing a simulation model to evaluate the effectiveness of our QoS enhancements to BGP. These enhancements will be evaluated in terms of the additional control overhead incurred due to their implementation, the overall network performance improvements, and their effect on improving the end-user perceived performance of applications running over the network.

To evaluate the protocol overhead on the network communications and the router resources, we require a high fidelity simulation model of BGP. In order to assess the impact on applications, we additionally require a simulation model which couples the control protocol routing decisions to the packet level transport over the network. Further, we require application models on which to assess end-user perception of the network performance.

No one simulation tool meets our requirements; hence we have embarked upon a simulation model development program to build the necessary models. We have chosen to base our simulation development upon the BGP++ [13][14] simulation tool. The BGP++ simulation is a port of the GNU Zebra BGP Daemon [15] and hence offers an extremely high fidelity representation of the BGP protocol exchanges, policies and capabilities. The BGP++ tool is distributed as a patch to the popular Network Simulation 2 (NS2) [16] and the Parallel and Distributed NS2 (PDNS) [17] event driven simulation packages. The NS2 package offers broad support for network protocol simulations. The PDNS package offers a path to large scale simulations. These tools, with some modifications and enhancements, will allow us to address the question of protocol overhead. Specifically, we are modifying the BGP++ code to incorporate our proposed QoS enhancements. This includes modifying the packet structures, the Routing Information Bases (RIBs) and the forwarding decision processes. Next, we must integrate the BGP++ RIB with the packet forwarding structure within the NS2 simulation tool.

In order to address the question of improved application performance, we may need to abstract the packet events into packet flows due to the vastly different time scales involved. The BGP protocol actions and changes are measured in terms of minutes, while packet transmission events in today's data networks are measured in tens of microseconds (a 1500-octet packet transmitted on an OC-12 link has a transmission time of about 20 microseconds). Therefore, we plan on using a fluid flow model, in particular the Integrated Fluid Flow Models (IFFM) [18], to model the majority of the traffic flows on high speed links while relying on packet level simulation for the low speed links and control packet events. The IFFM models are also available as patches to the NS2 simulation tool. The addition of this tool, with some modifications and enhancements, will allow us to study the application level performance. Specifically, we must integrate the routing of the fluid flows with the routing functions embedded within the NS2 simulation. Further, we need to develop a specific set of application models for our investigations. Our development efforts are currently underway. We plan on first addressing the control path and the issue of protocol overhead, and are currently modifying the BGP++ code for this purpose. We will then proceed with the rest of the development efforts discussed above.

SUMMARY

In this paper, we have provided a description of the issues involved in supporting multi-domain QoS routing with multi-dimensional QoS requirements. We provided a description of BGP operation and discussed its limitation. With routing based on the distance vector paradigm, BGP nodes have very limited visibility of the multi-path topology. We also identified where in the BGP decision process changes are needed and discussed some of the tradeoffs involved. In particular, the need to exchange multiple routes with associated QoS attributes that need to be accumulated across intra-domain and inter-domain segments, and the need to select and exchange the subset of dominant routes. We also discussed different options to ensure that packets follow the selected end-to-end routes depending on the IGP's QoS and forwarding capabilities. Ongoing work will get into further details of the enhancements and quantification of a number of issues identified in this paper including extensive performance evaluations using the simulation capabilities described above.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Phil Chimento and Brian Haberman with the Applied Physics Laboratory of Johns Hopkins University and Jennifer Rexford with the Computer Science Department of Princeton University for their valuable comments.

REFERENCES

[1] N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, and J. van der Merwe, "The case for separating routing from routers," Proc. ACM SIGCOMM Workshop on Future Directions in Network Architecture, pp. 5-12, Aug. 2004.

[2] M. Caesar, D. Caldwell, N. Feamster, J. Rexford, A. Shaikh, and J. van der Merwe, "Design and implementation of a Routing Control Platform," Proc. Networked Systems Design and Implementation, May 2005.

[3] G. Cristallo and C. Jacquenet, “The BGP QoS_NLRI Attribute”, draft-jacquenet-bgp-qos-00.txt, IETF work in progress, February 2004.

[4] L. Xiao, K. Lui, J. Wang, and K. Nahrstedt, “QoS Extensions to BGP,” Proc. IEEE Int’l Conf. on Network Protocols, pp. 100-109, 2002.

[5] L. Benmohamed and B. Doshi, “QoS Routing in Multi-Level Multi-Domain Packet Networks”, IEEE Milcom 2004.

[6] T. Bressoud, R. Rastogi, and M. Smith, "Optimal configuration for BGP route selection," Proc. IEEE INFOCOM, pp. 916-926, March 2003.

[7] L. Benmohamed et al., “Routing Algorithms for Dominant Path Selection,” Work in progress.

[8] T. Bates and R. Chandra, “BGP Route Reflection: an alternative to full mesh iBGP,” IETF RFC 1966, June 1996.

[9] D. Katz, K. Kompella, and D. Yeung, “Traffic Engineering Extensions to OSPF Version 2,” IETF RFC 3630, September 2003.

[10] H. Smit and T. Li, “IS-IS Extensions for Traffic Engineering,” IETF RFC 3784, June 2004.

[11] J. Moy, “OSPF – Anatomy of an Internet Routing Protocol,” Addison-Wesley 1988.

[12] H. Kaur et al., “BANANAS: An Evolutionary Framework for Explicit and Multipath Routing in the Internet,” Proc. ACM SIGCOMM Workshop on Future Directions in Network Architectures (FDNA), pp. 277-288, August 2003.

[13] Dimitropoulous, X. and G. Riley, "The BGP++ simulation", Georgia Tech University, February 2005.

[14] Dimitropoulous, X. and G. Riley, "Simulation Tool for Border Gateway Protocol Studies", Internal Georgia Tech University Report, November 2004.

[15] GNU Zebra, "The GNU Zebra Routing Demon - Border Gateway Protocol", http://www.gnuzebra.org/, January 2005.

[16] NS2, "The Network Simulator 2", USC - ISI, http://www.isi.edu/ns2/nam/, March 2003.

[17] Fujimoto, R., "The Parallel and Distributed NS2", Georgia Tech University, http://www.ece.gatech.edu/, January 2005.

[18] D. Townsley, "Integrated Fluid Flow Models", UMass - Amherst University, http://www.umass-amherst.edu/, January 2005.