COMBINING INTELLIGENT ROUTE CONTROL WITH BACKBONE TRAFFIC ENGINEERING TO DELIVER GLOBAL QOS-ENABLED SERVICES

A. Fonte, M. Pedro, M. Curado, E. Monteiro, F. Boavida
Departament of Informatics/CISUC, University of Coimbra
Email: {afonte,macpedro,marilia,edmundo,boavida}@dei.uc.pt

ABSTRACT
A global quality of service enabled Internet requires standard mechanisms to support the inter-autonomous systems quality of service connection. Two core components of the required functionality are the quality of service-enabled Border Gateway Protocol and the off-line backbone Traffic Engineering. Each of these approaches is, by itself, rather limited. Although the first enables the propagation of quality of service related messages and the latter potentially helps to improve the end-to-end quality of service, they both fail to achieve fast reaction to topology or path quality changes, as well as to users' requirements. Existing solutions have shown that these limitations can be partially addressed by employing Intelligent Route Controllers at multi-homed stub autonomous systems. Unfortunately, the ability of both parties, i.e., stub autonomous systems and Internet Service Providers, to choose their own routing policies does not necessarily lead to the best routing of IP packets in the Internet. In effect, each party seeks to control its traffic according to its own goals without considering the effects over the traffic or network of the other party. In this chapter the integration of Intelligent Route Controllers and off-line Traffic Engineering mechanisms is proposed to overcome the identified issues. The results show that this combination can produce synergistic interactions between Intelligent Routing Control and Traffic Engineering.
Keywords: QoS, Inter-domain Routing, BGP, Traffic Engineering, Intelligent Route Control

1 INTRODUCTION

Providing Quality of Service (QoS)-enabled services across Autonomous Systems (AS) boundaries is a very complex problem. QoS policies and services in one AS might be significantly different from those in other AS. In addition, an AS only controls its own network resources and just discloses a small part of its internal information (e.g. configurations or policy details) due to business constraints.

Recently, some research projects have addressed the important problem of inter-domain QoS and provided a full description of the functionality required to support inter-AS QoS connections [1,2]. The two key pieces of the proposed functional architectures are QoS-enabled Border Gateway Protocol (q-BGP) and off-line backbone Traffic Engineering (TE) for inter-domain QoS. On the one hand, q-BGP enables an AS to propagate to its peers BGP messages containing the information about the QoS connectivity services that it can provide (e.g., bandwidth and latency bounds, pricing and penalties) [3]. On the other hand, the off-line backbone TE enables an Internet Service Provider (ISP) to select the neighbor AS that will carry the customer traffic according to the network resource optimization objectives (e.g., the minimization of bandwidth consumption and the improvement of load-balancing) [4,5]. Therefore, the off-line backbone TE enables an AS to provide QoS connectivity services to the customers, as well as to identify the need of establishing new peering QoS relationships with neighbor autonomous systems.

The functionalities described have the potential to improve the QoS across AS boundaries. However, they may not completely achieve the following design requirements of inter-domain QoS: (1) Fast reaction to link failure and quality degradation; and fast recovery; (2) Being centered on user's perceived QoS level. In fact, q-BGP is not prepared to react in short timescales, without decreasing significantly the end-to-end QoS. And, the off-line TE at ISPs is only focused on local and coarse-grained traffic optimizations, being executed at large timescales, due to the complexity involved in the routing optimization process. Consequently, additional mechanisms operating at shorter timescales are needed to meet finer end-to-end QoS requirements of user applications.

The limitations identified above can be partially addressed by employing Intelligent Route Controllers (IRC), also called Smart Route Controllers (SRC), at multi-homed stub AS, as they provide a holistic way in which an AS solves locally end-to-end traffic challenges (e.g., latency, or loss rate bounds) through shifting some traffic between ISPs in short timescales [6,7]. By adopting Smart Route Controllers, the end-to-end performance or quality is improved without the need to change BGP routers and without the need of cooperation of the ISPs along the data paths.

Figure 1 illustrates a simple scenario of two autonomous systems employing Smart Route Controllers, where the SRC of AS2 might improve the performance of the outbound traffic toward the remote stub AS, AS3, through switching among the AS3-ISP1-ISP3 and AS3-ISP1-ISP4-ISP5 paths, across ISP3 and ISP5, respectively. A recent study reveals that this potential improvement in performance may reach up to 40% [8].

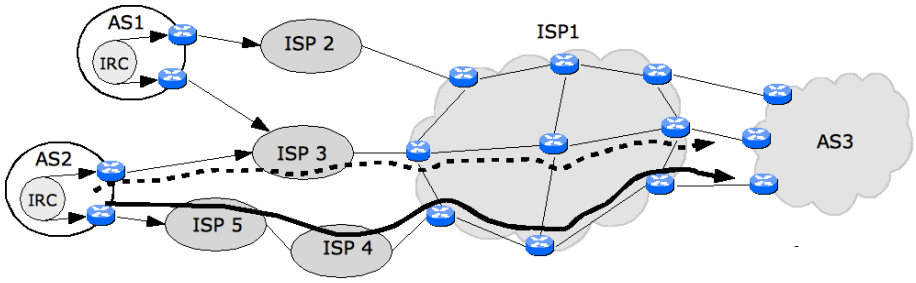

Figure 1: A simple scenario of two multi-homed stub ASes employing smart routing control.

Unfortunately, the ability of both parties, i.e., multi-homed AS and ISPs, to choose their own routing policies does not necessarily lead to the best routing of IP packets in the Internet. In effect, each party seeks to control its traffic according to its own goals without considering the effects over the traffic or network of the other party. In short, a major challenge of the research on Traffic Engineering and Internet Routing architectures is to accommodate this tussle.

This chapter comprises two main sections. In the first section, a broad overview of the use of Intelligent Route Controllers is provided, including their design principles and main algorithms. The effectiveness of using IRCs in networks where traffic differentiation is done per class is addressed by extensive simulation studies. In particular, path stability, traffic performance and reaction to path outages are evaluated in a scenario where IRCs are integrated within existing functional architectures in order to deliver QoS-enabled services. In the second section, a proposal of a novel Intelligent Routing Control-Traffic Engineering (IRC-TE) cooperative framework is described and evaluated through a large-scale performance study based on a realistic scenario. The results will show that this framework can produce synergistic interactions between Intelligent Routing Control and Traffic Engineering.

2 Intelligent Routing Control Framework

This section describes the design principles that play a key role in the success of inter-domain quality of service routing architectures and it details the main functions of Intelligent Route Controllers. Simulation results show how IRC can contribute to an improved traffic performance in a network where traffic differentiation is class-based.

2.1 Design Principles

Three key design principles of the inter-domain routing architecture, which are essential in enabling inter-domain QoS, are described in this subsection. Here, the doors of Intelligent Routing Control (IRC) to achieve the goals behind these principles are also open.

2.1.1 Decoupled performance/QoS Routing Control from BGP

The inter-domain routing of IP packets toward a given prefix involves four basic functions: Route Discovery, Route Filtering, Path Selection and Packet Forwarding [9].

- **Route Discovery** - At least a single route connecting both source and remote ASes should be discovered to ensure end-to-end connectivity between both ASes. To achieve this goal, each AS uses external BGP (eBGP) to propagate the best-learned routes to its neighbouring ASes, and uses internal BGP (iBGP) to distribute the routes inside the AS.
- **Route Filtering** - This function concerns the filtering of routes to be accepted from or to be advertised to the neighbouring ASes. Each AS can configure the import route filters of BGP routers to filter unwanted routes and/or to influence the selection of the remaining routes. It can also configure the export route filters of BGP routers to manipulate the route attributes or control whether a given route would be propagated to neighbour ASes.
- **Path Selection** - From the set of routes that passes import filters, BGP decision process ranks and selects the best routes to be installed in local RIB (routing information base). The best routes that pass export filters are then propagated to neighbouring ASes.
- **Packet Forwarding** - Once the best routes are chosen and installed in the local Routing Information Base (RIB), the BGP daemon of a router feeds the IP forwarding table so that IP packets can be forwarded along these routes.

Unfortunately, Routing Filtering and Path Selection are increasingly complex functions due to the growth of the Internet and the addition of new features to BGP [10,11]. Thus, any changes to the existing routing filtering and path decision processes to support QoS policies may difficult router configuration. In short, to face the BGP lack of QoS support, the addition of new features (e.g., new attributes and decision criteria) should be avoided.
Principle: *Path control limitations of current inter-domain routing should be addressed in a separated and distributed route control layer.*

To reduce the involved complexity needed to improve the quality of inter-domain routing of IP packets, standalone IRCs (or pairs of IRCs located at remote ASes) select the best routes on behalf of BGP, according to QoS goals. In other words, the BGP decision process is still used, but most routing intelligence is within IRCs. Therefore, signaling messages exchanged between IRCs are carried out by this out-of-BGP-band layer.

2.1.2 Fast link/QoS failure reaction and recovery

Inter-domain routing based on BGP is not prepared to react quickly to link failures. At the Internet scale, the BGP fail-over process may take several minutes [12,13]. To address this issue several improvements were proposed to reduce the number of messages distributed between BGP routers after a link failure. Unfortunately, even with these improvements, the convergence time scale still is on the order of a few minutes, which does not really fit the performance/QoS requirements of mission-critical or real-time traffic.
Principle: *An AS should be able to detect link/QoS failures and recovery at a short enough timescale, without degrade significantly the end-to-end quality.*

To achieve this goal, an IRC must react much faster than the BGP layer to link or QoS failures. Therefore, an IRC system relies on end-to-end QoS monitoring along every link connecting the AS to the Internet. Then, when an IRC detects quality violations against specified thresholds for a given traffic flow, it adapts on-the-fly the local routing behaviour. In this situation an alternative path that is able to bypass the network link failure or congestion is selected. In other words, IRCs enhances the end-to-end QoS of the underlying BGP layer by tweaking route attributes in very short timescales given that no BGP messages will be ever spawned.

2.1.3 Being centered on user's perceived QoS level.

Operators can employ inter-domain Traffic Engineering to meet the traffic challenges of local autonomous systems, such as load-balancing or minimizing the maximum utilization of peering links. This type of Traffic Engineering is usually performed by tweaking BGP route attributes in order to change an arbitrary number of routes according to the computed solution.

When the traffic demands are stable, the inter-domain traffic engineering optimizations potentially result in network operation points close to the optimum. However its major drawback is that traffic objectives are not centered on the user's perspective of network quality. This means that upon finding a new optimal routing pattern it gives no guarantees that the end-to-end quality properties for a particular traffic aggregate are compliant with the QoS requirements (e.g., one-way delays or round-trip times bounds). Even though quality centered traffic engineering is used, it potentially requires much more processing and route tweaks than the traditional scheme, due to the higher dynamics of end-to-end performance metrics.

Principle: *In order to improve the user's perceived QoS, the ASes should be centered on end-to-end QoS objectives.*

To achieve this goal, an IRC employs both performance monitoring and path selection processes centered on end-to-end performance objectives. By using this approach, an IRC can follow the path quality and adapt the routing of individual traffic aggregates to the real-time network conditions. As a result, this implies that IRCs are not suitable to transit ASes, in particular to large transit ASes such as Tier-1 or Tier-2 ISPs, because adapting the routing of the traffic aggregates to network conditions might require changing a large number of paths, and the effects of managing large amounts of inter-domain traffic in short timescales are unpredictable.

2.2 IRC KEY FUNCTIONS

Figure 2 presents a logical diagram of an IRC, which captures the common functionalities from BGP-based IRCs [6,14,15,29]. The main running activities of an IRC in every routing cycle are the following. First, an IRC derives the traffic demands and captures the quality of all available paths. Then, it does an on-line processing to compute the required path changes. Finally, it issues command scripts to routers with the corresponding BGP tweaking. These activities are described in detail in the next subsections.

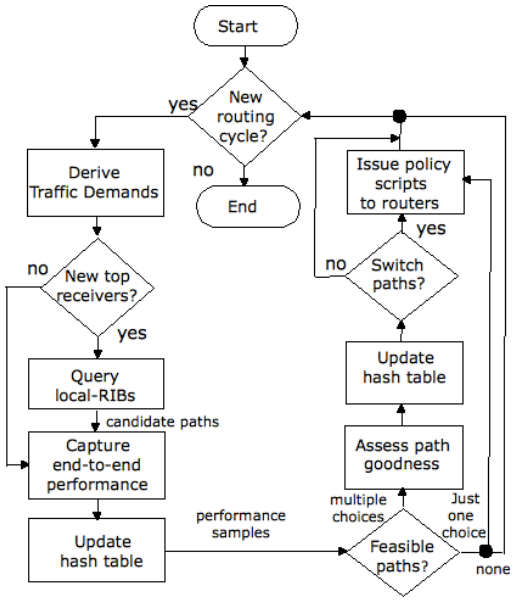

Figure 2: Intelligent Route Controller logical presentation.

2.2.1 Path Monitoring

IRCs are composed of a Path Monitor (PM) component, which has the role of capturing, on a real-time basis, the quality attributes of paths (e.g., one-way delay (OWD), one-way-loss (OWL) or avail-bw (available bandwidth)). In this process, two monitoring methods can be employed independently or combined to derive the estimates of quality attributes of paths:

- **Active measurement monitoring** - the PM component is endowed with a mechanism to spawn, at a given frequency, small probes targeting the remote destination through every available link connecting the local AS to the Internet. The streams of probes are then monitored to determine the quality of the user/network, assuming implicitly that the values measured from the probes represent accurately the perceived quality of a user/network. Methods like ICMP time-stamp Request and Reply with a specific mark on the Differentiated Services Code Point (DSCP) field of IP packets (e.g., set to AF11 (Assured Forwarding Class 1 Low)) can be used to derive quality metrics, such as the One-way delay (OWD) or Round-Trip Time (RTT) [16].
- **Passive measurement monitoring** - Usually, the PM collects samples of data packets to derive incoming or outgoing traffic volumes. This is especially important to avoid the self-load effect, and the resulting path instability. Moreover, the PM can also derive other performance or quality attributes over the available path options. For instance, instead of using active probing, the PM might use the TCP acknowledgment mechanism to derive the RTTs. More specifically, the PM can monitor pairs of TCP segments (sent by local TCP sources) and the corresponding TCP acknowledgements (sent by the remote TCP sinks).

Synchronization between probes and the overhead introduced are two issues that must be addressed. The first can be avoided by adding randomization in the sampling times and the second by using a conservative-enough frequency of probes, while keeping efficient routing [6,17].

2.2.2 Dynamic Path Switching

Dynamic path switching is the key technique used by IRC systems to get better end-to-end performance [6,14,15,29]. With this approach, the IRC selects, in every routing cycle, the next-hop ISP to forward packets depending of the quality attributes of paths. However, in practice, after collecting the quality measures, the IRC uses a cost or utility function, denote by $M(.)$ or $U(.)$, to determine whether one path performs better than another. More precisely, for instance, the $M(.)$ expression might be given by a positive quantity -a cardinal value- direct or inversely proportional to the smoothed quality measurement, depending on whether this measurement is additive or concave (e.g., $M \propto latency$ or $M \propto \frac{1}{bandwidth}$).

Knowing that an alternative path performs better than the active path is not a sufficient condition to pick it due to stability issues. Therefore, three path switching policies are considered:

- **Choose Best-Choose Best policy (CBCB)** - According to the CBCB policy, the IRC switches paths whenever it finds better paths in terms of QoS. So, for each destination, the IRC picks the path that has the smallest value of the chosen metric independently of any QoS bound;
- **Choose Best-Choose Good policy (CBCG)** - Similarly to the CBCB policy, the IRC switches paths whenever it finds better paths in terms of QoS. However, there is an important difference; here, the IRC is aware of the end-to-end quality bounds, so that if the quality of a path does not fit these bounds, it is enough to pick any alternative good path;
- **Choose Good policy (CG)** - According to the CG policy, the IRC only switches paths if their QoS characteristics are not sufficient to handle traffic QoS requirements. So, in that case, it picks any good alternative path.

2.2.3 Shifting traffic over ISPs

IRCs must then change the route BGP attributes reflecting the path descending order (i.e., from the best to the worst path). Two simple solutions to implement the traffic optimizations are by means of the `LOCAL-PREFERENCE` or `MED` route attributes change, as described next.

Local-Preference Tweaking: The `LOCAL-PREFERENCE` attribute indicates the degree of preferences for a route as compared to other available routes for the same prefix (a higher `LOCAL-PREFERENCE` means more preferred). Therefore, if we denote $P = [P_1,P_2,...,P_k]$ as the path descending order, and the corresponding vector of routing costs, denoted as $M = [C_1,C_2,...,C_k]$, the simplest mapping is the linear mapping as in equation (1). It would allow the mappings between each C_i and its `LOCAL-PREFERENCE` value of the cardinal type.

$$Local-pref(P_i) = p_{min} + \left\lfloor \left(p_{max} - p_{min}\right)\left(1 - \frac{C_i}{max\left[C_1,C_2,...,C_k\right]}\right)\right\rfloor + \Delta Tie \qquad (1)$$

, where $Local\text{-}pref(P_i)$ denotes the `LOCAL-PREFERENCE` to assign to P_i within a range $[p_{min},p_{max}]$ and ΔTie denotes a small optional Tie-breaking factor that may be add by the IRC to the `LOCAL-PREFERENCE` value to tie-break equally good routes.

The biggest advantage of a solution based on `LOCAL-PREFERENCE` tweaking is that it overrides any other route attribute. However, this implies that when an IRC uses it for traffic engineering purposes it may violate local business policy (unless two `LOCAL-PREFERENCE` ranges are used, one for enforcing business policies and other for traffic engineering purposes).

MED Tweaking: In this case, the mapping between the routing costs and `MED` values is similar to Eq. (1), except that the preferred routes have lower `MED` values. However, with this solution the MEDs are only compared if the routes have equal AS path lengths, which may limit the effectiveness of the traffic optimizations deduce by the IRC system.

2.3 SIMULATION STUDY

Intelligent Route Controllers were evaluated using the following path switching policies - choose-best-choose-best (CBCB), choose-best-choose-good (CBCG) and choose-good (CG), when the Differentiated Services (Diffserv) feature [18] is enabled or disabled, resulting on four pairs of simulation scenarios:

1. CBCB IRCs and Diffserv on (CBCB-DSon) / CBCB IRCs and Diffserv off (CBCB-DSoff);
2. CBCG IRCs and Diffserv on (CBCG-DSon) / CBCG IRCs and Diffserv off (CBCG-DSoff);
3. CG IRCs and Diffserv on (CG-DSon) / CG IRCs and Diffserv off (CG-DSoff);
4. BGP and Diffserv on (BGP-DSon) / BGP and Diffserv off (BGP-DSoff).

The simulations were performed using the J-Sim [19] simulator in which the functionalities of the IRCs were implemented.

Figure 3 illustrates a simplified picture of the network model used. The simulated network aims at representing a multi-service part of the Internet composed by access ISPs able to provide some limited QoS services to their customers, and an over-provisioned Internet core (composed by Tier 1 and big Tier 2 ISPs).

In this set-up, the tests were conducted using a traffic mix consisting of Voice over IP (VoIP) calls, video calls, prioritized data, and Web traffic. When Diffserv feature is enabled, packets are classified and forwarded using the standard Expedited Forwarding (EF), Assured Forwarding (AF11 and AF21), and Best-Effort (BE) traffic classes, respectively [20,21].

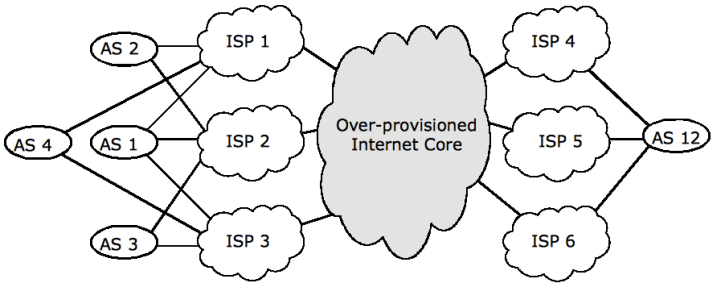

Figure 3: Network model.

During all the experiments, it is assumed that a service-level specification (SLS) was previously exchanged between remote multihomed stub domains, based on maximum OWDs for each service. These maximum OWDs tolerated per-service were chosen to represent reasonable but demanding values for the kinds of traffic sources considered. Thus, the OWD bound for voice and video traffic was set at 100ms, and at 400ms for prioritized data and web traffic, which are in accordance with the E-model Rating from ITU's G.107/G.114 recommendations [22,23].

2.3.1 Performance Metrics and Objectives

The first objective is to assess how much IRCs aid to improve end-to-end network quality. This is shown by evaluating: (i) the average end-to-end OWD (or latency), denoted as *owd*, and (ii) the traffic transfer efficiency for the different traffic flows, denoted as *Ef*. The efficiency for a traffic class is defined by $Ef = \frac{F_D}{C_O}$, where F_D is the total throughput at destinations D, and C_O is the corresponding total throughput sent by source domains O.

The second objective is to study how IRCs contribute to the overall network stability under variable QoS dynamics. The performance indicator is the total number of path switches needed to meet the latency constraint for each kind of traffic.

The third objective is to find the overall efficiency of each IRC path switching policy and BGP. The overall efficiency is given by an Efficiency index (see Eq. (2)), which represents the relative number of path switches that were needed to obtain a given latency and traffic transfer efficiency. Lower values for this index indicate better overall efficiency, i.e., that there is a better trade-off between the number of path switches, the averaged OWD and the traffic efficiency; and so the IRC path switching policy is more efficient.

$$Efficiency\ index\ =\ PSindex\left(\frac{d}{d_{max}}\right)\left(\frac{100}{Ef}\right) \tag{2}$$

, where *PSindex* is a path shifts index computed, as follows.

$$PSindex\ =\ \begin{cases} \dfrac{PS}{PS_{ref}} & ,\ if\ PS \geq PS_{ref} \\ \dfrac{abs\left(PS - PS_{ref}\right) + PS_{ref}}{PS_{ref}} & ,\ if\ PS < PS_{ref} \end{cases} \tag{3}$$

, where PS_{ref} is the number of path switches performed by the IRC adopting a given reference policy.

In the evaluations of the overall efficiency, the CG policy was used as the base reference. Thus, the *PSindex* is given by the number path switches performed by other policies normalized by the number of path switches performed by the IRC CG policy. Otherwise, the *PSindex* is given by a penalty that depends of the number of path switches that would be done by the IRC police to achieve the same performance of the IRC CG policy.

Finally, the fourth objective is to compare the path failover time that a stub AS gets when it uses an IRC system with the time BGP needs to detect and recovery from the same path outages. In the tests, the interval of times between the instant of the occurrence of the failure and the instant of the path switch at the AS source of traffic were recorded. We also recorded the times needed to recovery the throughput on the destination.

2.3.2 Results

The first results concern the latency obtained with the different IRC path switching policies and BGP. The cluster of figures 4 illustrates the Complementary Cumulative Distribution Functions (CCDF) of the traffic latencies. If the probability of the OWD is greater than or equal to x is high (i.e., $P(OWD \geq x)$ is high), it means that there is a high likelihood that the traffic will suffer a latency greater than or equal to x. In turn, figure 5 presents the averaged latencies for each traffic.

Two main conclusions can be drawn from the analysis of these results. First, the IRC architecture substantially enhances end-to-end QoS when compared with a pure BGP model. On the one hand, when Diffserv feature is disabled and IRCs are switched OFF, under stressful

traffic load, BGP is not capable of supporting ITU's performance bounds. In particular, figure 4 shows that the BGP-based scheme presents a likelihood of violation of all OWD bounds greater than 95%. On the other hand, when IRCs are switched ON, figure 4 shows that all IRC policies for all traffic classes have a likelihood of violation of the maximum bounds greater than 50%. This is not necessarily bad given that the simulations were carried out under stressful traffic load. Another observation is that the CG policy presents longer tails, so that for less important traffic classes (i.e., data prioritized and Web traffic), there is a probability greater than 10% that the latency exceeds 500 ms. This would be expected, since the CG policy only reacts to QoS violations.

Second, Diffserv clearly shows its effectiveness to protect the most important traffic classes, since all the traffic classes have OWDs within the ranges allowed, except for web traffic, and sometimes, for the interactive data prioritized traffic when BGP is enabled. Although this may sound surprising, this is only possible due to a strong efficiency penalization of the data prioritized traffic and web traffic. Figure 6 clearly shows that, especially for the BGP case; in fact, more than 60% of web traffic is dropped.

Next, the discussion of the IRC path switching policies on controlling the number of path switches is performed. Figure 7 presents the total number of path switches registered for each policy. As one would guess, the IRC path switching algorithm based on CBCB policy presents the larger number of path switches, since according to CBCB policy, an IRC switches paths whenever a better path is found. This is especially evident when Diffserv is disabled, and particularly for the less important traffic classes. As it would be expected, the IRC path switching algorithm based on CG policy provides the best performance in terms of the number of path switches. However, this is not necessarily good given that this policy only reacts when there is a clear violation of the performance bound. As mentioned before, longer tails in the cluster of figures 5 clearly show that this policy might allow unacceptable latencies.

Figure 8 depicts the overall performance index results. As it would be expected, BGP presents the weaker overall efficiency. On the other hand, the CG policy appears to be the most efficient scheme, at least in absolute terms. These results show that just a small number of path switches is enough to improve dramatically the performance of traffic in terms of latency and efficiency.

However, combining these results with the ones from the figure 4, where, as earlier mentioned, it is possible to observe that the CB policy presents long tails, we consider the CBCG policy as the most balanced scheme. In other words, a few number of extra path switches performed by an IRC using this policy allows to avoid episodes of high latency. On contrary, the results of figure 8 also clearly show that a number of path switches larger than necessary might not lead to performance improvements. This is especially evident when the Diffserv feature is enabled.

Finally, in this section we try to assess how the IRCs are able to manage and react to a remote link failure. Figure 9 contrasts the results obtained when the IRCs are in use against the results obtained by BGP. We have aggressively configured small `KEEPALIVE` and `HOLD` timer values (30 and 90 seconds respectively) so as to increase the speed of reaction of BGP. From Figure 9 we can observe that in case of links failures occurring a few hops away from a certain AS, the IRCs are able to react, on average, between 3-4 times faster than the time that BGP needs to converge to new route. In addition, since the IRCs gather end-to-end measurements their responsiveness becomes independent of the place where the remote link failure occurs. As a result, IRCs are also able to reduce performance degradations (e.g. packet loss and latency) or service outages arising from inaccurate routing states during BGP transient fail-over periods.

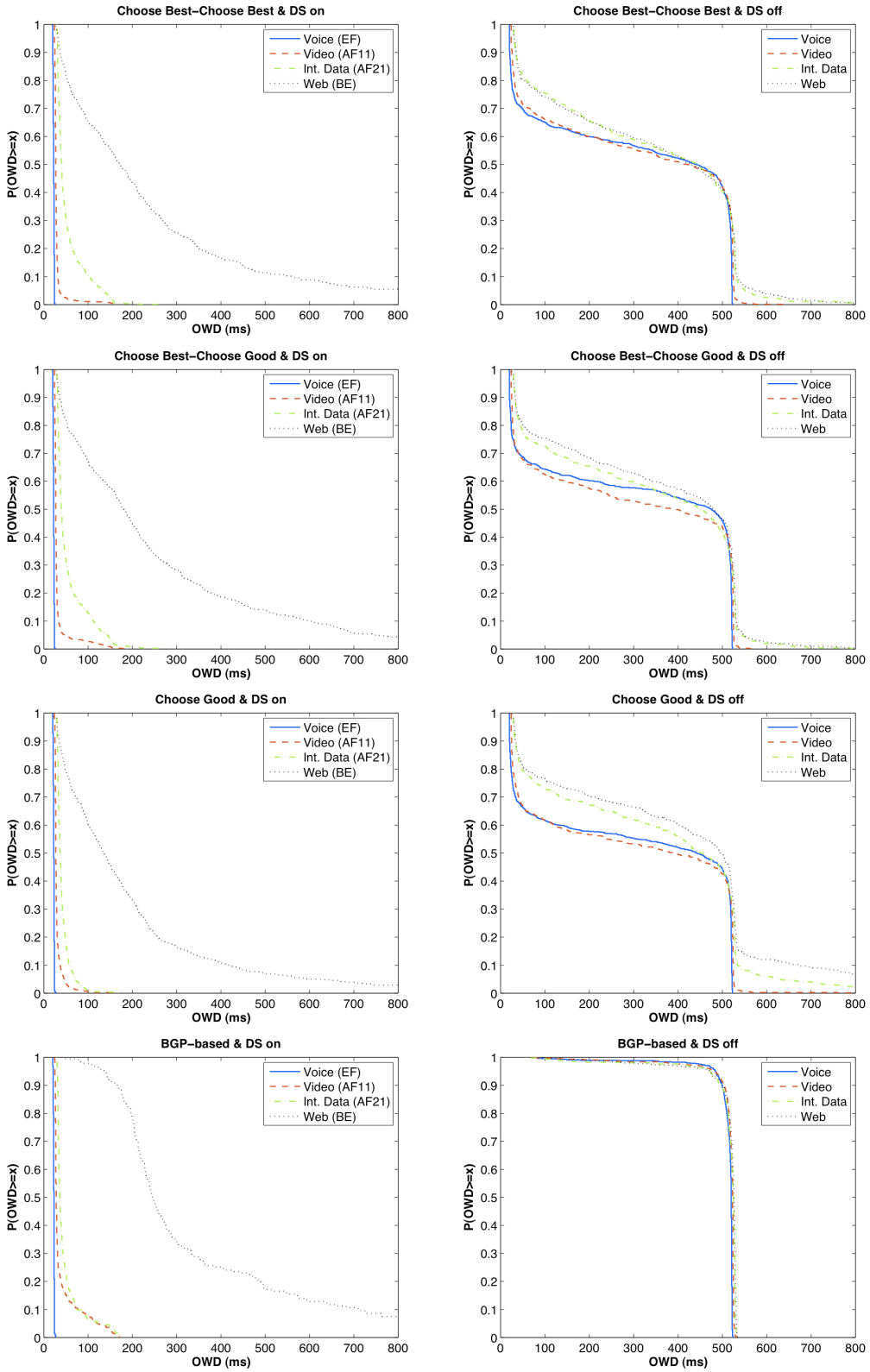

Figure 4: Complementary Cumulative Distribution Function (CCDF) of OWDs for each traffic (Left) whether Diffserv feature is enabled or (Right) Diffserv feature is disabled.

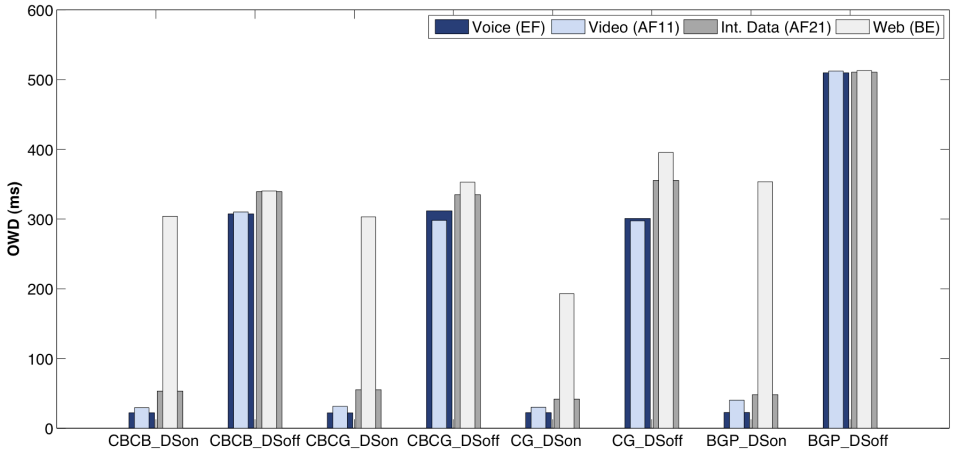

Figure 5: Average OWD for each traffic.

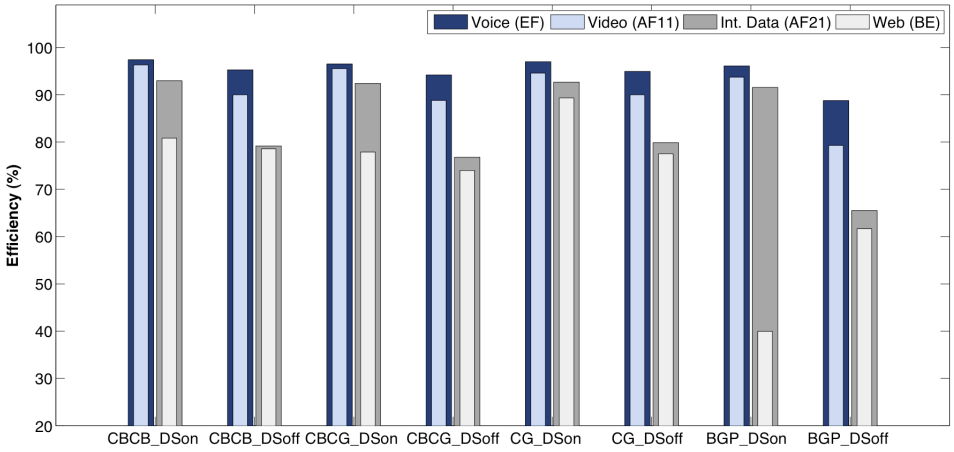

Figure 6: Transfer Efficiency for each traffic.

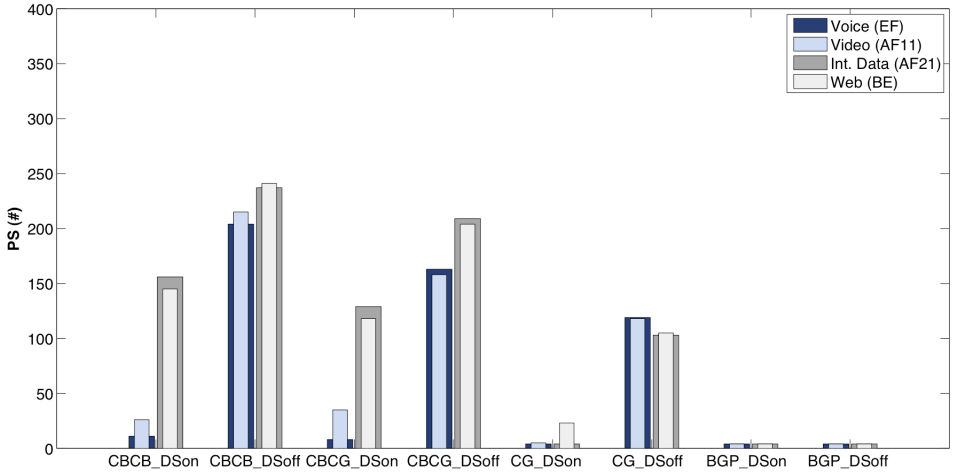

Figure 7: Total path shifts for each traffic.

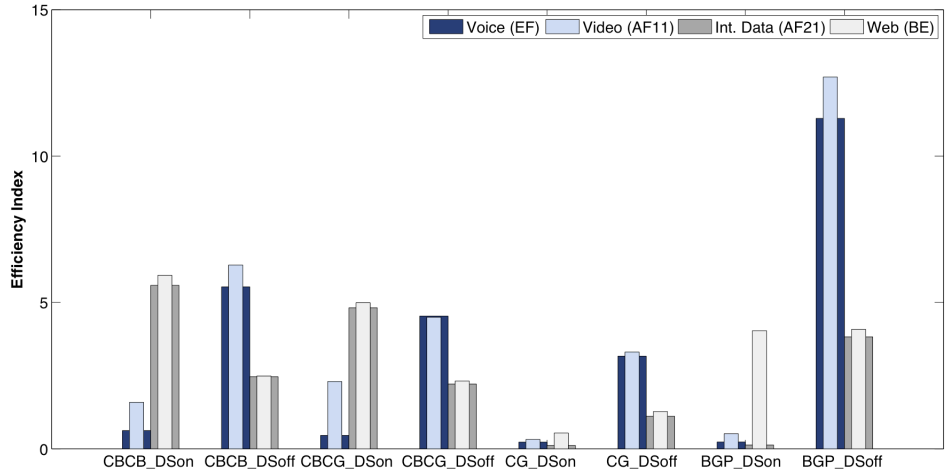

Figure 8: Global Efficiency Index for each traffic.

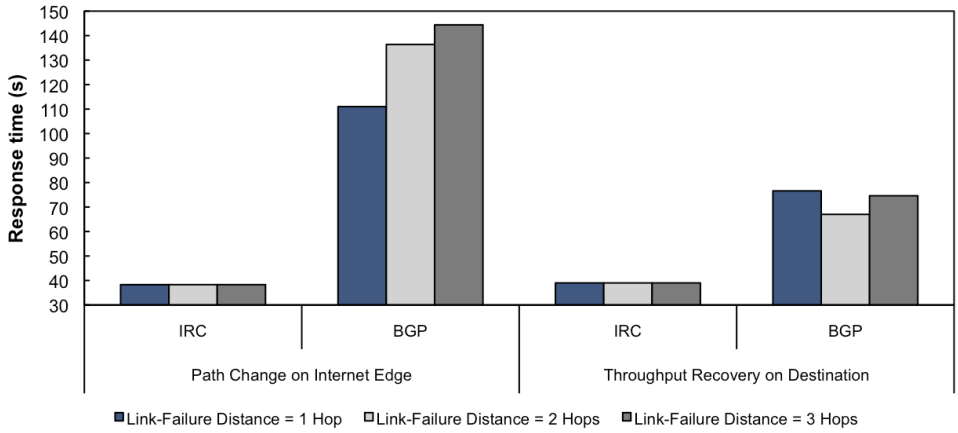

Figure 9: Contrasting the response time to a link failure.

3 Combining IRC with Backbone Traffic Engineering

The benefits of employing Intelligent Routing Control (IRC) are evident from the technical and economical perspectives [14,15,24]. However, IRC mechanisms commonly rely on the selfish routing approach. They allow stub ASes to perform outbound traffic optimization, but the target optimum is only local since the paths (or transit providers) are greedily selected. In other words, selfish routing makes IRCs unaware of the effects of their route choices over transit ASes. The practical result is that the networks of transit ASes may become far away from optimal performance regimes, and congestion can occur on intra or inter-domain links of their networks. This is caused by the increased difficulty to get accurate forecasts of traffic demands, due to rapid route changes performed by IRC [26].

Similarly, selfishness is also a common attribute to Traffic Engineering boxes for transit ASes. So, in turn, these controller boxes compute a new local routing solution to return networks of transit ASes to the optimal regime taking into account current traffic demands and local traffic objectives. Unfortunately, resulting route changes can then interact with IRCs by changing the end-to-end quality of traffic; so that IRCs react to this quality change by adapting the routing in

stub ASes. This way uncoordinated routing changes attempting to face congestion can produce cycles of influence between stub ASes and ISPs. Figure 10 summarizes the control actions of IRC and TE boxes over the traffic, and corresponding interactions between both boxes.

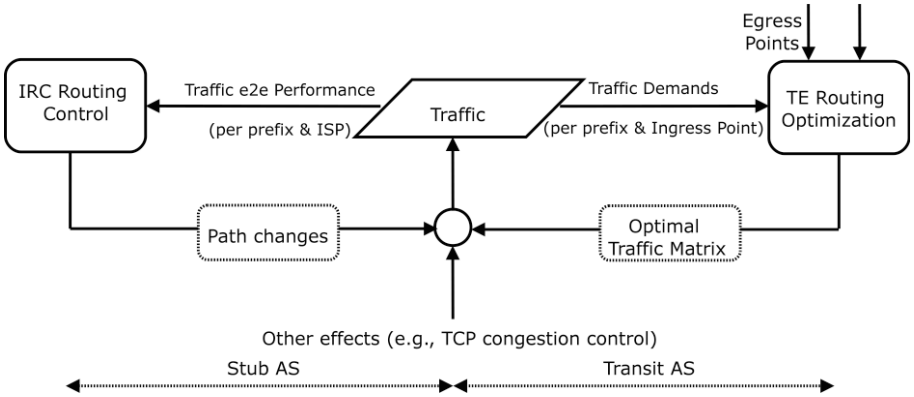

Figure 10: Model of actions of IRC and TE boxes over traffic.

To address the above issues, this section considers a conceptually general and simple distributed out-of-BGP-band cooperative approach able to support in some sense coordinated inter-domain routing decisions among stub and transit AS for the context (but not limited) of the future QoS-aware Internet. The particularity of this approach is that it allows IRCs finding 'social' optimums by honouring both individual stub and transit AS preferences (or constraints) regarding the carrying (or admission of new) traffic amounts of certain traffic class (TC) aggregates.

To summarize, the proposed approach is motivated and well suited to:

1. Protect stubs ASes from performance and QoS violations against end-to-end performance and quality bounds;
2. Improve the traffic demands predictability over transit ASes;
3. Reduce selfish costs, i.e., traffic performance losses, TE performance losses and routing instability.

Starting from some background on inter-domain Traffic Engineering heuristics, this section provides a broad overview of an IRC-TE cooperative scheme, including a description of the main mechanisms and algorithms that integrate the IRC-TE cooperation framework, and finally some results demonstrating the framework's feasibility.

3.1 Backbone Traffic Engineering Heuristics

Let us first briefly describe a typical inter-domain backbone TE process in an ISP network with a set of ingress points I, a set of egress points E and a set of reachable prefixes P. The inputs of TE process are the incoming traffic demands (TD), the egress point choices for each prefix $p \in P$ (given by BGP) and the egress point capacities. Its output is the optimal ID routing i.e., the routing that ensures the optimal performance regime according to a given traffic objective. Then, at a given timescale t this process is repeated as the traffic demands might fluctuate over the time. In this section, we focus on fluctuations due to routing changes in multihoming users employing IRC. Following, these users are referred, simply as IP network customers (INCs).

More formally, the predicted TD over the ISP $k \in K$ for the slot time t is represented by the matrix $D_{(k,t)}$, where K is the set of ISPs and each entry $D_{(k,t)}(i,p)$ is the predicted TD over ingress

point i of ISP k for prefix p. If each INC $h \in H$ has T data transfers at rates $x_h(p)$ to distribute over its ISPs, then each entry of $D_{(k,t)}$ is defined as in equation (4).

$$D_{(k,t)}(i,p) = \sum_h \sum_p x_h(p).r_{(h,k,t)}(i,p) \tag{4}$$

, where H is the set of INCs and $r_{(h,k,t)}(i,p) \in \{0,1\}$ is an indicator function to select whether the route from INC h to prefix p via ingress point i of ISP k is active (i.e., True (1), False(0)).

On the other hand, we represent the egress resources of ISP k as C_k, where each entry $C_k(e)$ is the capacity at each egress point e. And, we represent the inter-domain routing from ISP k as $\varepsilon_{(k,t)}$, where each entry $\varepsilon_{(k,t)}(i,e,p) \in \{0,1\}$ is an indicator function that tells whether the $D_{(k,t)}(i,p)$ is assigned to the egress point e.

The border egress router selection (BERS) problem being addressed is the following: *How to assign each entry of traffic demands $D_{(k,t)}(i,p)$ to an egress point e, so as to optimize a certain traffic performance objective*.

In this section, a typical objective, the min-max link utilization, is encoded in BERS problem to ensure that egress link utilization is at lowest levels and thereby to minimize congestion. Other objectives can also be encoded in BERS (e.g., min-cost routing or maximum business profit) [27]. Before proceeding, let first define the link utilization of e for a routing $\varepsilon_{(k,t)}$ as:

$$U_e = \sum_i \sum_p \frac{\varepsilon_{(k,t)}(i,e,p).D_{(k,t)}(i,p)}{C_k(e)}(i,p),\ \varepsilon_{(k,t)}\ is\ a\ routing. \tag{5}$$

Objective - Minimizing the Maximum Link Utilization (min-MLU). One possibility for the BERS problem would be to minimize the maximum link utilization, i.e.,

$$min\, max\, U_e,\ \forall e \in E,\ \varepsilon_{(k,t)}\ is\ a\ routing. \tag{6}$$

This objective is subject to the following constraints:

$$\sum_i \sum_p \varepsilon_{(k,t)}(i,e,p).D_{(k,t)}(i,p) \le C_k(e),\ \forall e \in E \tag{7}$$

$$with \sum_e \varepsilon_{(k,t)}(i,e,p) = 1,\ \forall i \in I \tag{8}$$

The BERS goal to be addressed is to minimize traffic objective (6). The capacity constraint (7) ensures that the total resource requirements of the traffic flows assigned to each egress point do not exceed the available contracted capacity. The assignment constraint (8) guarantees that each traffic flow is assigned to exactly one egress point e.

The BERS algorithm used is based on a genetic single objective version of [5]. The algorithm, belonging to the class of evolution strategies used in optimization, resembles the process of biological evolution, where each individual is described by its genetic code, called a chromosome. In turn, each chromosome is composed of individual genes. In the problem in hand, a gene is the assignment of a single aggregate traffic flow to an egress point of the ISP, and an individual (i.e., a chromosome) is a potential solution.

The basic algorithm steps are presented in Algorithm (1). It starts with the creation of the initial generation, where the individuals are created randomly. Then, an evaluating step based on

the proposed objective function (6) follows. After that, and for a number of generations, a new generation of children is created and compared with the corresponding generation of parents. From this comparison the better elements will compose the next generation of parents. The last of the generations is the aimed TE solution.

Algorithm 1: Basic steps of the genetic Backbone Traffic Engineering algorithm.

```
1: Create the initial parent generation;
2: Evaluate the generation;
3: for a number of generations do
4:    Create the child generation;
5:    Evaluate both generations together;
6:    Rank both generations together;
7:    Replace worst parents with better children;
8: end for
```

3.2 IRC-TE Cooperative Framework

Generically introducing a certain degree of cooperation in the routing decisions can reduce selfishness. In the case addressed, this corresponds to incorporate ISP feedback signals into the path decision process of IRCs, so that the paths would be selected given the observed path quality and the associated ISP feedback. A path is thus characterized by a tuple of attributes including the path quality measures and ISP feedback signals.

The mechanisms used aims, on the one hand, to compute ISP feedback signals from the current ISP network regime; and, on the other hand, to compare a pair of path tuples. However, the exact mechanisms design depends of the nature of the ISP feedback signals. To narrow the space of possible solutions, the following design decisions were made:

1. *ISP feedback signals advertise ISP route prices* to carry the amount of traffic from stub AS customers employing IRC. These prices depend of local target traffic objectives of the ISP (see (e));
2. *ISP route prices are advertised through an opaque metric.* The exact mapping function used to calculated this metric is not shared between ISPs and stub AS customers employing IRC. This way stub AS customers employing IRC have no possibility to uncover the goals or procedures employed by ISPs;
3. *The opaque metric values are bounded to* [0,1] to enable fair comparisons between different ISP route prices formulations, and to avoid normalization to mix these prices with path performance attributes;
4. *ISP route prices are dynamically adapted by ISPs* in response to ISP load changes and deviations from a predefined target range. This target range is defined taking into account the optimal regime of the ISP network previously set by the ISP Traffic Engineering process for a given traffic objective;
5. *Use of MIAD (Multiplicative Increase Additive Decrease) policy to adapt the maximum values for ISP route prices*. This mechanism has been introduced to ensure that the ISP can effectively repulse/attract traffic from/to its network in order to keep ISP load within the target range. Further details are given in figure 11.

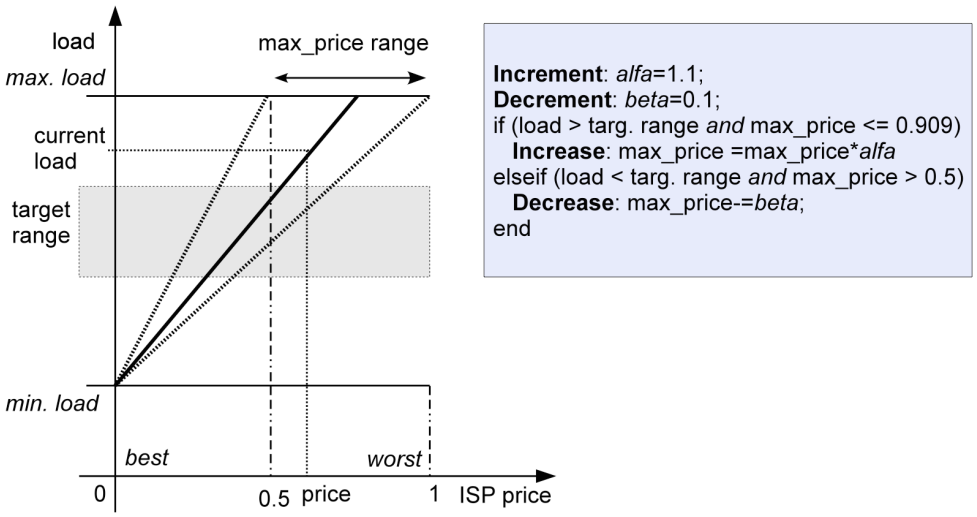

Figure 11: Mapping of ISP performance measures to agnostic values.

3.3 UTILITY-BASED IRC ALGORITHM

The Utility-based Intelligent Routing Control algorithm relies on the notion of utility function [32]. In the design of the solution proposed, it is consider that each IRC middlebox has a utility function associated with each traffic class. Therefore, it can determine the IRC/end-User degree of satisfaction after shifting a traffic aggregate to a given path.

Denoting generically d as the delay measured (OWD or RTT) at the INC_i throughout the path offered by the ISP_k, there are a reasonable number constraints concerning utility values $U(d)$:

1. $U(d):[0,d_{max}] \rightarrow [0,1]$, If the path offers a delay higher than the delay bound d_{max} its utility is zero;
2. $lim_{d \rightarrow 0^+} U(d) = 1$, since the delay tends to zero, the utility tends to zero;
3. $U'(d) \leq 0$, since the utility should decrease as the delay increases;
4. $U''(d) > 0$, to make the utility more sensitive as the delay approaches the delay bound.

In this design, a modified sigmoid utility function is used, since it fits these constraints (see equation 9). In addition, its parameter α is set to 0.0125 to allow the utility to follow the shape of the E-model Rating from ITU's G.107/G.114 recommendations [22,23].

$$U(d) = \begin{cases} 1 - \dfrac{2}{1 - exp(-\alpha(d - d_{max}))} & , \; if \; d \in [0, d_{max}[\\ 0 & , \; if \; d > d_{max} \end{cases} \quad (9)$$

, where α determines the steepness the curve.

As described previously, ISPs use feedback signals (i.e., route prices) to indicate their willingness regarding the traffic being injected into their networks by a given stub AS customer. So, each Cooperative IRC (C-IRC) computes the surplus, defined as the difference between the utility and the route price, and selects the path with the maximal surplus. Algorithm 2 details the Utility-based IRC algorithm.

Algorithm 2: IRC($\{P,A,C\}$)

Require: $\{P\}$ – vector of the set of AS paths for a prefix p
$\{A\}$ – matrix of the set of performance/QoS attributes
$\{p\}$ – vector of criteria representing the ISP route prices
$\{d_{max}\}$ – delay traffic bound

Ensure: P_x - the active path fits traffic goals toward prefix p

1: *Wait* for changes in the QoS attributes
2: /* Basic IRC path selection process */
3: Compute the vector of Utilities U
4: Identify the set of feasible paths P'
$[P',U'] \leftarrow \{P_1 \in P : U_i(d_i) > 0\}$
5: **if** $\|P'\| \neq 0$ **then**
6: /* Identify the highest rank path $P_i \in P'$, which gives the maximal surplus*/
$x' = arg\,max\ \chi(\{P',U'\}) = U_i(d_i) - p_i, \forall P_i \in P'$ /* the ranking function χ compares the surplus of all feasible paths*/
7: **if** $\|x'\| > 1$ **then**
8: /* If there is more than one path equally good, apply the standard BGP process to break the ties*/
$P'' \leftarrow \{P_i \in P' : i = x'\}$
$x' \leftarrow BGPTie(P'')$
9: **end if**
10: Switch traffic towards p from P_x to $P_{x'}$
11: $P_x \leftarrow P_{x'}$
12: **end if**
13: /* End of IRC path selection process */

3.4 SIMULATION EVALUATION

For performance studies, four IRC-TE combinations were considered depending on whether IRC and TE mechanisms were switched ON or OFF, and the IRC-TE cooperation was enabled or disabled. The results presented were obtained using a realistic simulation model built from BGP routing table dumps and data traces collected at each of the 23 nodes of the GÉANT pan-European academic network [30]. In this model, the GÉANT network employs a genetic inter-domain TE algorithm, and ASes (representing European countries) connected to GÉANT employ IRC.

In accordance with the best traffic engineering practices, it is assumed that both IRC and TE boxes only focus on the traffic toward top receivers, the so called popular prefixes or destinations, which are a small fraction (i.e., about 5 - 10%) of the total number of receivers. This assumption arises from the property of traffic demands of GÉANT being consistent with a Zipf-like distribution [25]. Thus, after a route summarizing process followed by the flow rankings computation based on the individual contributions of each flow to the total volume of traffic, 296 popular destinations among a total of 16150 prefixes were identified. The popular destinations were identified such the sum of their individual traffic volumes is a fraction of 99% of total traffic volume.

A last point to note concerning the simulation model is that the AS-level topology used in the simulations is composed of 731 ASes and corresponding peerings, which were inferred from the GÉANT BGP routing table dumps taking only into account the popular destinations. Further details about GÉANT traffic and the AS-level topology are presented in figures 12 and 13.

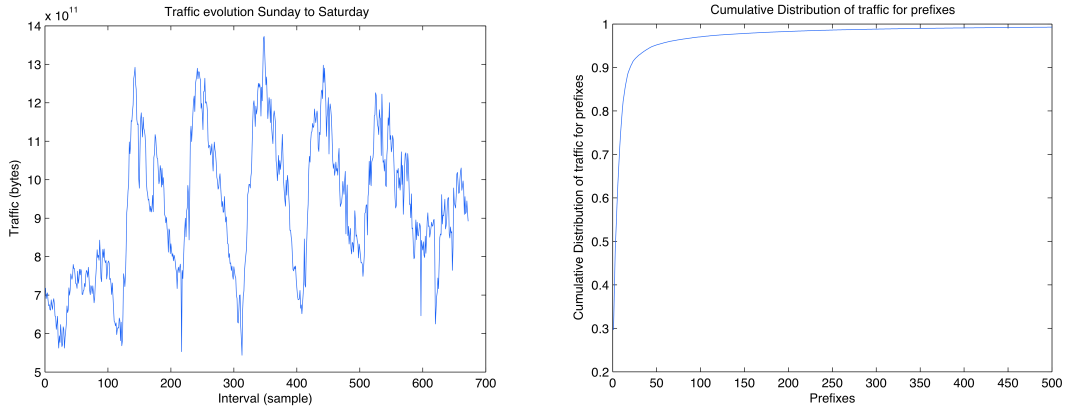

Figure 12: GÉANT Traffic: (left) Weekly Traffic evolution; (right) Distribution of Traffic volumes.

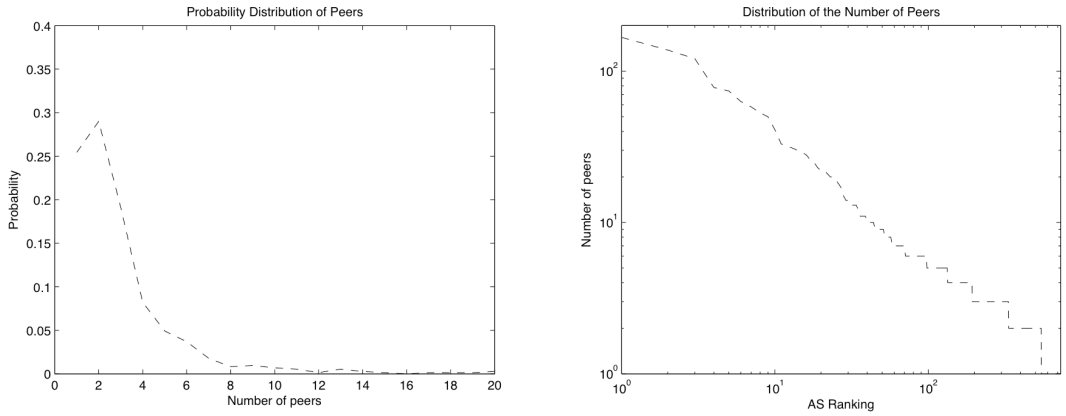

Figure 13: AS-Level topology: (left) Probability of a number of peers; (right) Distribution of the number of peers.

The results reported in figures 14-16 allow an evaluation of the IRC-TE cooperative framework and its ability to produce synergistic interactions between IRC and TE. Synergistic interactions has a threefold meaning:

i. latency measured with IRC-TE cooperation is equal or better than the latency measured without IRC-TE cooperation;

ii. link utilization with IRC-TE cooperation is lower than link utilization without IRC-TE cooperation;

iii. and, the total number of path switches with IRC-TE cooperation is smaller than the number of path switches without IRC-TE cooperation.

Otherwise, the interactions between IRC and TE are antagonistic.

The results reported in figures 14 and 15 show the averaged latency for each destination and the median of the link utilization in egress links of GÉANT for all four IRC-TE combinations. They reveal two significant observations concerning the cooperative IRC-TE scheme. First, it can ensure a performance similar to the one obtained by the scheme without IRC-TE cooperation. Second, it can improve the link utilization of the ISP network. In figure 15, the median link utilization presents lower values across the simulation. This is particularly evident in figure 15 across the interval of samples [150,440]. On contrary, the selfish IRC may hurt the TE performance especially for higher loads, which is across the interval of samples [550,640].

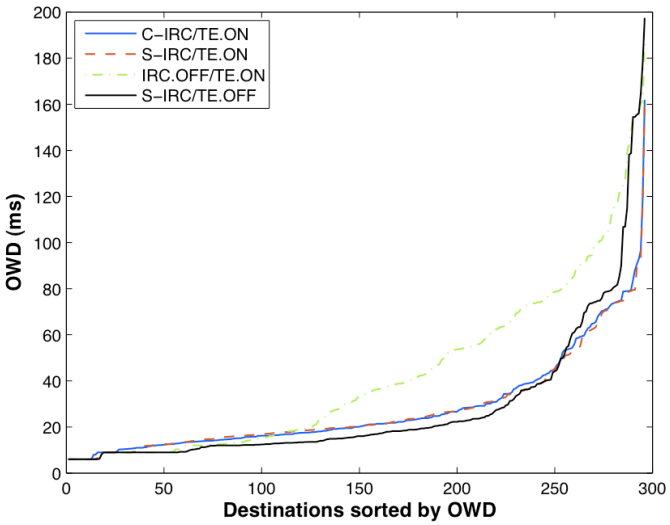

Figure 14: Latency measured for each destination.

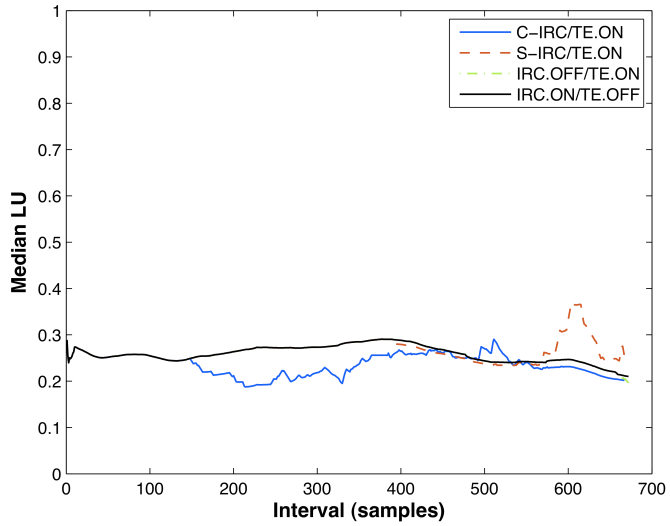

Figure 15: Median of Link Utilization.

IRC stability is discussed next. Figure 16 shows the total number of path switches for all IRC-TE combinations. What we see is, first, TE is clearly antagonistic to IRC, since both IRC/TE.ON combinations need a few more thousands of path switches than the IRC.ON/TE.OFF combination in order to meet the same OWD constraint. However, it appears that this is compensated by better OWD distributions. Second, there is an overall stability benefit from combining IRC and TE according to a cooperative scheme. In fact, this combination roughly needs less 7% of the total number of path shifts needed by the IRC-TE combination without cooperation.

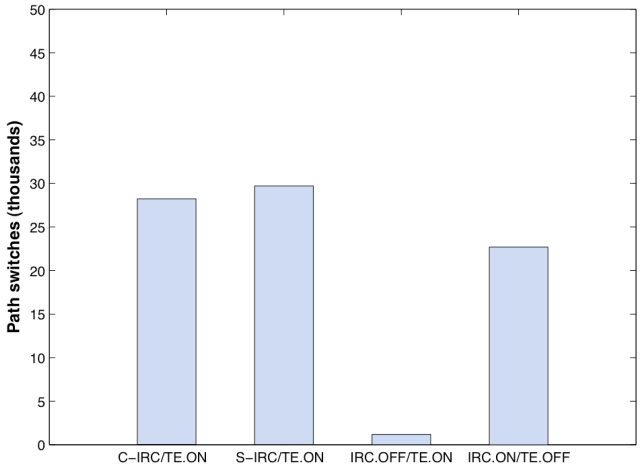

Figure 16: Total number of path switches performed by IRCs.

To sum up, these results, as a first main conclusion, they substantiate the claim on the reduction of the effectiveness of TE due to IRC route control. In other words, this means that although TE is able to adapt efficiently the inter-domain routing to traffic demands fluctuations due to the sources, it might be unable to accommodate traffic changes due to IRCs. As a second main conclusion, these results show that a cooperative IRC-TE scheme can produce synergistic interactions between IRC and TE.

4 Conclusion

In this chapter, multihoming intelligent routing control has been introduced as an alternative approach in enabling inter-domain quality of service routing. First, the main design principles of intelligent route controllers and their algorithms were presented. Results from a simulation study have shown that intelligent routing control can substantially enhance end-to-end quality of service when compared with a pure Border Gateway Protocol (BGP) model.

Given that intelligent routing control and inter-domain traffic engineering can interact due to selfish routing changes, a novel framework that combines both mechanisms in order to produce synergistic interactions was described.

The novelty of this framework is due to the mutual cooperation between intelligent routing control and traffic engineering, and on the use of out-BGP-band signaling. Specifically, this mechanism supports the exchange of feedback signals between traffic engineering and intelligent routing control, that notify intelligent route controllers from route price modifications due to traffic load changes over the internet service provider (ISP) network. A significant feature of feedback signals is their agnosticism to ISP traffic objectives.

Based on a utility-based algorithm, an intelligent route controller can integrate ISP feedback signals into its routing decision process, and find ’social’ optimums, honouring both individual stub and transit preferences. Performance results were presented and demonstrated the feasibility of this framework both in terms of stability and traffic performance.

References

[1] X. Masip-Bruin, M. Yannuzzi, R. Serral-Gracia, J. Domingo-Pascual, J. Enriquez-Gabeiras, M. Callejo, M. Diaz, F. Racaru, G. Stea, E. Mingozzi, A. Beben, W. Burakowski, E. Monteiro, L.

Cordeiro, The EuQoS System: A Solution for QoS Routing in Heterogeneous Networks, in IEEE Communications Magazine, 45(2):96-103, February 2007.
[2] M.P. Howarth, P. Flegkas, G. Pavlou, N. Wang, P. Trimintzios, D. Griffin, J. Griem, M. Boucadair, P. Morand, H. Asgari and P. Georgatsos, Provisioning for Inter-domain quality of service: the MESCAL approach, IEEE Communications Magazine, June 2005.
[3] G. Cristallo and C. Jacquenet, Providing Quality of Service Indication by BGP-4 Protocol: the QOS NLRI attribute, IETF draft, IETF, 2002.
[4] K.H. Ho, N. Wang, P. Trimintzios and G. Pavlou, Traffic Engineering for Inter-domain Quality of Service, Proc. London Communications Symposium (LCS), London, UK, September 2003.
[5] Pedro, M. and Monteiro, E. and Boavida, F. , An Approach to Off-line Inter-domain QoS-Aware Resource Optimization, in Proc. of the Networking 2006, pp. 247-255, Networking 2006, Coimbra, Portugal, May 2006.
[6] R.Gao et al., Avoiding Oscillations due to Intelligent Route Control Systems. In the Proc. of IEEE INFOCOM 2006, Barcelona, Spain, April 2006.
[7] M. Yannuzzi, X. Masip-Bruin, E. Marin-Tordera, J. Domingo-Pascual, A. Fonte, and E. Monteiro, Improving the Performance of Route Control Middleboxes in a Competitive Environment, in IEEE Network, Vol. 22, no. 5, Sep./Oct. 2008.
[8] A. Akella et al., A Measurement-Based Analysis of Multihoming. In the Proc. of ACM SIGCOMM 2003, Karlsruhe, Germany, 2003.
[9] Y. Rekhter, A Border Gateway Protocol 4 (BGP-4), IETF RFC 4271, January 2006.
[10] http://bgp.potaroo.net/index-bgp.html. Web page accessed at June 2009.
[11] T. Bates, R. Chandra, D. Katz and Y. Rekhter, Multiprotocol Extensions for BGP-4, IETF RFC 4760, IETF (2007).
[12] C. Labovitz, R. Wattenhofer, S. Venkatachary, and A. Ahuja, The impact of internet policy and topology on delayed routing convergence, In Proc. of IEEE INFOCOM 2001, April 2001.
[13] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanianitz, Delayed internet routing convergence, In IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 9, NO. 3 (2001).
[14] Cisco Systems, Inc., Performance Routing.
[15] Internap Networks Inc., Flow Control Platform.
[16] J. Postel, Internet Control Message Protocol, IETF RFC792, September 1981.
[17] Fonte, A., Martins, J., and Curado, M. and Monteiro, E. , Stabilizing Intelligent Route Control: Randomized Path Monitoring, Randomized Path Switching or History-Aware Path Switching?, in Proc. of the 11th IFIP/IEEE International Conference on Management of Multimedia and Mobile Networks and Services Management of Converged Multimedia Networks and Services (MMNS 2008), Samos Island, Greece, September 2008.
[18] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z. and W. Weiss, An Architecture for Differentiated Services, IETF RFC 2475, December 1998.
[19] J-Sim Homepage, http://www.j-sim.org.
[20] V. Jacobson, K. Nichols and K. Poduri, An Expedited Forward-ing PHB, RFC 2598, IETF, June 1999.
[21] J. Heinanen, F. Baker, W. Weiss, J. Wroclawski, Assured Forwarding PHB Group, RFC 2597, IETF, June 1999.
[22] ITU-T Recommendation G.107: The E-Model, a computational model for use in transmission planning, 2003.
[23] ITU-T Recommendation G.114: One-way transmission time, 2003.
[24] R. Dai, D. Stahl, and A. Whinston, The economics of smart routing and QoS, In Proc. of the Fifth Inter. Workshop on Networked Group Comm (NGC'03), 2003.
[25] W. Fang and L. Peterson, Inter-as traffic patterns and their implications. In Proceedings of the 4th Global Internet Symposium, December 1999.
[26] R. Teixeira, N. Duffield, J. Rexford, and M. Roughan, Traffic Matrix Reloaded: The Impact of Routing Changes, in Proc. of PAM, March 2005.
[27] T. Bressoud and R. Rastogi., Optimal Configuration for BGP Route Selection. In Proc. of IEEE INFOCOM 2003, San Francisco, USA, April 2003.
[28] A. Fonte, M. Pedro, E. Monteiro, and F. Boavida, Analysis of Interdomain Smart Routing and Traffic Engineering Interactions, in Proc. of IEEE Globecom 2007 Internet Protocol Symposium, Washington, DC, USA, November 2007.

[29] M. Yannuzzi, A. Fonte, X. Masip-Bruin, M. Curado, J. Domingo-Pascual, E. Monteiro, and S. Sanchez-Lopez, On the Advantages of Cooperative and Social Smart Route. In Proc. of ICCCN 2006, Arlington, Virginia, USA, October 2006.
[30] S. Uhlig, B. Quoitin, J. Lepropre and S. Balon, Providing public intradomain traffic matrices to the research community, Journal of SIGCOMM Comput. Commun. Rev, 2006.
[31] G. Almes et al., A one-way delay metric for IPPM, IETF, Request for Comments 2679, September 1999.
[32] H. R. Varian, Microeconomics Analysis, third edition. W. W. Norton and Company, Inc., 1992.