Improving the Performance of Route Control Middleboxes in a Competitive Environment

Marcelo Yannuzzi, Xavi Masip-Bruin, Eva Marin-Tordera, Jordi Domingo-Pascual,
Technical University of Catalonia
Alexandre Fonte, Polytechnic Institute of Castelo Branco
Edmundo Monteiro, University of Coimbra

Abstract

Multihomed subscribers are increasingly adopting intelligent route control solutions to optimize the cost and end-to-end performance of the traffic routed among the different links connecting their networks to the Internet. Until recently, IRC practices were not considered adverse, but new studies show that in a competitive environment, they can lead to persistent traffic oscillations, causing significant performance degradation rather than improvements. To cope with this, randomized IRC techniques were proposed. However, the proliferation of IRC products raises concerns, given that randomization becomes less effective as the number of interfering IRC systems increases. In this article, we present a more scalable route control strategy that can better support the foreseeable spread of IRC solutions. We show that by blending randomization with adaptive filtering techniques, it is possible to drastically reduce the interference between competing route controllers, and this can be achieved without penalizing the end-to-end traffic performance. In addition to the potential improvements in terms of scalability and performance, the route control strategy outlined here has various practical advantages. For instance, it does not require any kind of protocol or coordination between the competing IRC middleboxes, and it can be adopted readily today because the only requirement is a software upgrade of the available route controllers.

Today, the vast majority of the communications on the Internet are between nodes located in non-transit (i.e., stub) networks. Stub networks are primarily composed of medium and large enterprise customers, universities, public administrations, content service providers (CSPs), and small Internet service providers (ISPs). These networks exploit a widespread practice called multihoming, which consists of using multiple external links to connect to different transit providers. By increasing their connectivity to the Internet, stub networks potentially can obtain several benefits, especially in terms of resilience, cost, and traffic performance [1]. These are described as potential benefits because multihoming per se cannot improve any resilience, cost, and traffic performance. Accordingly, multihomed stub networks require additional mechanisms to achieve these improvements. In particular, when an automatic mechanism actively optimizes the cost and end-to-end performance of the traffic routed among different links connecting a multihomed stub network to the Internet, it is referred to as intelligent route control (IRC).

During the last few years, IRC has attracted significant interest in both the research and the commercial fields. Several vendors are developing and offering IRC solutions [2–4] that increasingly are being adopted by multihomed stub net-

works. Most available IRC solutions follow the same principle, that is, they dynamically shift part of the egress traffic of a multihomed subscriber from one of its ISPs to another, using measurement-driven path switching techniques. IRC systems operate in relatively short timescales — even reaching switching frequencies on the order of a few seconds — allowing IRC users to balance cost and performance criteria according to the priority and requirements of their applications.

Despite these strengths, IRC practices have one major weakness, that is, they try to achieve a set of local objectives individually without considering the effects of their decisions on the performance of the network. Recently, it was discovered that in a competitive environment, IRC systems actually can cause significant performance degradation rather than improvement. In [5], the authors show that persistent oscillations can occur when independent controllers become synchronized due to a considerable overlap in their measurement time windows. To avoid synchronization issues, the authors propose randomized IRC strategies and empirically show that the oscillations disappear after introducing a random component in the route control decision.

It is important to note that although randomization offers a straightforward mechanism to mitigate the oscillations, it cannot guarantee global stability. This issue raises concerns given the proliferation of IRC products because as the number of interfering IRC systems increases, randomization becomes

---
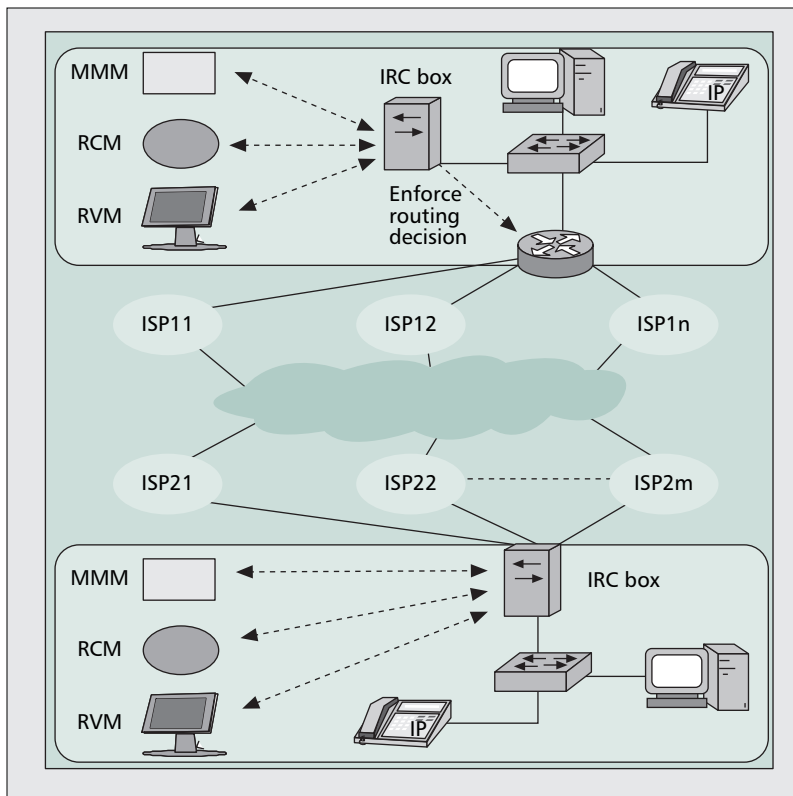
■ Figure 1. *The IRC model. IRC systems are composed of three modules: the monitoring and measurement module (MMM), the route control module (RCM), and a reporting and viewer module (RVM).*

less effective, and hence, the more likely it is that the oscillations reappear. In light of this, it is necessary to explore more scalable route control strategies that can safely support the foreseeable spread of IRC solutions.

In principle, two research approaches can be taken. On the one hand, the research community could formally study the stability properties of IRC practices and provide guidelines on how to design IRC systems with guaranteed stability. Unfortunately, several challenging stages must be completed properly before a formal study of stability can be conducted. For instance, accurate measurements are required to understand comprehensively the actions of the closed-source IRC systems deployed today (e.g., [2–4]) and thereby, model the stochastic distribution of path switches in a competitive IRC environment. Only after characterizing the distribution of path switches, is it possible to formally study the stability aspects of competitive IRC.

In the absence of such characterization, the practical alternative is to find ways to drastically reduce the potential interference between competing route controllers without penalizing the end-to-end traffic performance. This is precisely the challenge addressed in this work. This article makes the following contributions:
• We show that although randomization offers a straightforward way to mitigate the oscillations, it leads to a large number of unnecessary path switches.
• We report some of our recent results on the development of strategies blending randomization with a lightweight and more "sociable" route-control algorithm. The term sociable route control (SRC) refers here to a route control strategy that explicitly considers the potential implications of its decisions in the performance of the network and can adaptively restrain its intrinsic selfishness depending on the network conditions.
• We show that a simple enhancement to randomized IRC

systems, such as endowing them with an SRC algorithm supported by adaptive filtering techniques, is enough to drastically reduce the number of path switches, and most importantly, this can be accomplished without penalizing the end-to-end traffic performance. Extensive simulations show that with SRC, it is possible to reduce the overall number of path switches between approximately 40 to 80 percent on average (depending on the load on the network) and still obtain better end-to-end traffic performance than with randomized IRC techniques in a competitive environment.

The rest of the article is structured as follows. First, we present the basics of IRC. Then, we overview the most relevant related work. Next, we analyze some general aspects of different IRC strategies and describe the SRC approach together with some of our main results. We conclude with directions for future research in the area of IRC.

The Basics of IRC

A typical IRC scenario with two different configurations is shown in Fig. 1. The IRC box at the top of Fig. 1 is connected by a span port off a router or switch so although the egress traffic is controlled by the box, it is never forwarded through it. The IRC box in the multihomed network at the bottom of Fig. 1 is placed along the data path so traffic always is forwarded through it. Typically, the former configuration offers a more scalable solution than the latter, in the sense that it is able to control and optimize a larger number of traffic flows.

Conceptually, an IRC system is composed of the following three modules (Fig. 1):
• Monitoring and measurement module (MMM)
• Route control module (RCM)
• Reporting and viewer module (RVM)

The existing IRC systems can control a moderately large number of flows[1] toward a set of *target* destination networks. These target destinations can be configured manually or discovered by means of passive measurements performed by the MMM. By using passive measurements, the MMM can rank the destinations according to the amount of traffic sourced from the local network and subsequently optimize the performance for the traffic toward the *D* destinations at the top of the rank. The MMM also uses passive measurements to monitor the target flows in real time and analyze packet losses, latency, and retransmissions, among others, as indicators of conformance or degradation of the expected traffic performance. To assist the RCM in the dynamic selection of the best egress link to reach each target destination, the MMM probes all the candidate paths using both Internet Control Message Protocol (ICMP) and Transmission Control Protocol (TCP) probes.

The set of active and passive measurements collected by the MMM enables IRC systems to concurrently assess the quality of the active and the alternative paths toward the target destinations. The role of the RCM is to dynamically

---

[1] *Typically this is on the order of several hundreds and even thousands, using a configuration like the one shown at the top of Fig. 1 with several border routers.*

choose the best egress link for each target flow, depending on the outcome of these measurements. More specifically, the RCM is capable of taking rapid routing decisions for the target flows, often avoiding the effects of issues such as distant link/node failures[2] or performance degradation due to congestion.[3]

The third module of an IRC system, namely the RVM, typically supports a broad set of reporting options and provides online information about the average latency, jitter, bandwidth utilization, and packet loss experienced through the different providers, summaries of traffic usage, associated costs for each provider, and so on.

Overall, IRC offers an incremental approach, complementing some of the key deficiencies of the Interior Gateway Protocol/Border Gateway Protocol (IGP/BGP)-based route control model. It is worth emphasizing that the set of candidate routes to be probed by IRC boxes usually is determined by IGP/BGP; so conversely to overlay networks [8], IRC boxes never circumvent IGP/BGP routing protocols. The effectiveness of multihoming in combination with IRC is confirmed not only by studies like [8], but also by the increased trend in the deployment of these solutions.

In this article we deal with the algorithmic aspects of IRC systems so hereafter we focus our attention on the RCM in Fig. 1 — the functionality of the MMM and RVM modules essentially is orthogonal to the proposals made in this work.

Related Work

In [9], the authors simultaneously optimize the cost and performance for multihomed stub networks, by introducing a series of new IRC algorithms. The contributions of that work are fundamentally theoretical. For instance, the authors show that an intelligent route controller can improve its own performance without adversely affecting other controllers in a competitive environment, but the conclusions are drawn at traffic equilibria (traffic equilibrium is defined by the authors as a state in which no traffic can improve its latency by unilaterally changing its link assignment). However, after examining and modeling the key features of conventional IRC systems, it becomes clear that they do not seek this type of traffic equilibria. Indeed, more recent studies, such as [5], show that in practice, the performance penalties can be large, especially when the network utilization increases.

In light of this, and considering the current deployment trend of IRC solutions, it becomes necessary to explore alternative IRC strategies. These new route control strategies should *always* improve the performance and reliability of the target flows, or at least, they should drastically reduce the potential implications associated with frequent traffic relocations, such as persistent oscillations causing packet losses and increased packet delays [5].

Although most commercially available IRC solutions do not reveal in depth the technical details of their internal operation and route control decisions, the behavior of one particular controller is described in detail in [10]. That work also provides measurements that evaluate the effectiveness of different design decisions and load balancing algorithms. Akella et al. also provided rather detailed descriptions and experimental evaluations of multihoming in combination with IRC tools, as in [1, 8, 11]. These research publications, along with the documentation provided by vendors, allowed us to capture and model the key features of conventional IRC techniques. A similar approach was followed by the authors in [5]. For simplicity, and as in [5, 8, 10], we consider traffic performance as the only criteria to be optimized for the target flows.[4]

The General IRC Network Model

The general IRC network model is composed of a multihomed stub network S, a route controller $\mathbb{C}$, the transit domains, and a set of target destinations $\{d\}$ with cardinality $|d| = D$ to be optimized by $\mathbb{C}$. The source domain S has a set of egress links $\{e\}$, with $|e| = E$. For the sake of simplicity, we keep the notation in the granularity of destinations (d), but the model easily can be extended to consider various flows per target d.

To dynamically decide the best egress link for each target destination d, the MMM in $\mathbb{C}$ probes all the candidate paths through the egress links e of S. Then, the collected measurements are processed and abstracted into a performance function $P_e^{(d,t)}$ at time t, associated with the quality perceived for each of the available paths toward the target destinations d. Let $N^{(d)}$ denote the number of available paths to reach d. Because $N^{(d)}$ usually represents the number of candidate paths in the forwarding information base (FIB) of the BGP border routers of S, $N^{(d)} \leq E \ \forall \ d$.

We assume that the better the end-to-end traffic performance perceived by $\mathbb{C}$ for a target destination d through egress link e, the lower the value of the performance function $P_e^{(d,t)}$.

In this framework, IRC strategies can be taxonomized into two categories, namely, reactive route control (RRC) and proactive route control (PRC). RRC practices switch a target flow from one egress link to another only when a maximum tolerable threshold (MTT) is met. The MTTs are application-specific and typically represent the maximum acceptable packet loss, the maximum tolerated packet delay, and so on, for a given application. Beyond any of these bounds, the performance perceived by the users of the application becomes unacceptable.

PRC strategies, on the other hand, switch traffic before any of the MTTs are met and in turn, can be taxonomized into two categories: those that can be called fully proactive (FP), and those that follow a controlled proactivity (CP) approach. FP IRC practices always switch to the best path. Therefore, the dynamic optimization problem addressed by a FP route controller is to:

Find the min$\{P_e^{(d,t)}\} \ \forall$ d, t *and enforce the redirection of the corresponding traffic to the egress link found.*

The alternative offered by CP is to keep the proactivity, but switch traffic as soon as the performance becomes degraded to some extent, typically represented by a relocation threshold (R_{th}). The dynamic optimization problem addressed by CP-based strategies can be formulated as follows.

Let e^{best} *denote the egress link utilized to reach* d *at time* t, *and let* e′ *be such that* $P_e^{(d,t)} = min\{P_e^{(d,t)}\}$ *for destination* d *at time* t.[5] *A CP-based route controller would switch traffic to* d *from* e^{best} *to* e′ *whenever* $P_{e^{best}}^{(d,t)} - P_e^{(d,t)} \geq R_{th}$, *with* $R_{th} > 0$.

---

[2] *The timescale required by IRC systems to detect and react to a distant link/node failure is very small compared to that of the general IGP/BGP routing system [2–4, 6].*

[3] *This cannot be automatically detected and avoided with BGP [7].*

[4] *Cost reductions are typically accomplished by aggregating traffic toward non-target destinations over the cheapest ISPs.*
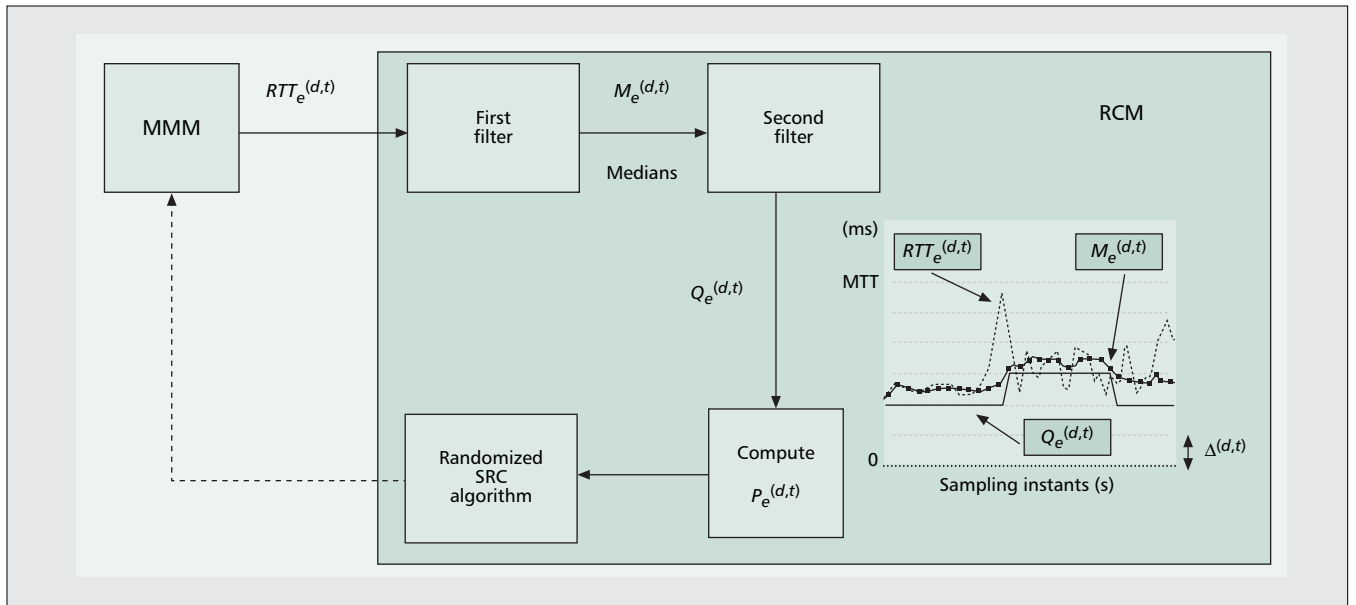
■ Figure 2. *Filtering process and interaction between the monitoring and measurement module (MMM) and the route control module (RCM) of a sociable route controller. The Randomized SRC Algorithm within the RCM is outlined in Algorithm 1.*

After extensive evaluations and analysis, we confirmed that PRC performs much better than RRC. The reason for this is that proactive approaches can anticipate network congestion situations, which in the reactive case, typically demands several traffic relocations when congestion already was reached. In addition, we found that in a competitive environment, CP-based route control strategies can outperform the FP ones. Therefore, our SRC algorithm (outlined in the following section) is supported by a CP-based route control strategy.

Sociable Route Control

In the SRC strategy that we conceive, each controller remains independent so the SRC boxes do not require any kind of coordination with one another — just as conventional IRC systems operate today. Moreover, our SRC strategy does not introduce changes in the way measurements are conducted and reported by conventional IRC systems, so both the MMM and the RVM in Fig. 1 remain unmodified. Our SRC strategy introduces changes only on the algorithmic aspects of the RCM.

High-Level Description of the SRC Strategy

For simplicity in the exposition, we focus on the optimization of a single application, namely, voice over IP (VoIP), and we describe the overall SRC process for the round-trip time (RTT) performance metric. For a comprehensive and formal analysis, the reader is referred to [12].

Our goal is that a controller $\mathbb{C}$ becomes capable of adaptively adjusting its proactivity, depending on the RTT conditions for each target destination d. To be precise, a sociable controller analyzes the evolution of the RTT, that is, $\{RTT_e^{(d,t)}\}$, and depending on its dynamics, the controller can restrain its traffic reassignments adaptively (i.e., its proactivity). To this end, the RCM processes the RTT samples gathered from the MMM using two filters in cascade (Fig. 2). The first filter corresponds to the median RTT, $M_e^{(d,t)}$, which is constantly computed through a sliding window. This approach is used widely in practice because the median represents a good estimator of the delay that the users' applications cur-

rently are experiencing in the network. These medians are precisely the input to the second filter, where the social nature of the route control algorithm covers two different facets:
• CP
• SRC

Controlled Proactivity

On the one hand, the proactivity of box $\mathbb{C}$ is controlled to avoid minor changes in the medians triggering traffic relocations at S. This prevents interfering too often with other route controllers. For this reason, our sociable controllers filter the medians.

The second filter in Fig. 2 works like an analog-to-digital (A/D) converter, with quantization step Δ, and its output is one of the levels of the converter $Q_e^{(d,t)}$. The right-hand side of Fig. 2 illustrates how the instantaneous samples of RTT are filtered to obtain the median $M_e^{(d,t)}$, and then, the latter is filtered to obtain $Q_e^{(d,t)}$.

As described earlier, IRC systems compare the quality of the active and alternative paths by means of a performance function $P_e^{(d,t)}$, which as shown in Fig. 2, is fed by $Q_e^{(d,t)}$. The controller $\mathbb{C}$ would switch traffic toward d only when the variations of $Q_e^{(d,t)}$ cause that $P_{ebest}^{(d,t)} = P_e^{(d,t)} \geq R_{th}$. A more detailed description of the route selection process is shown in Algorithm 1. For simplicity, only the stationary operation of the algorithm is summarized. The randomized nature of Algorithm 1 is discussed later. The timer in Step 8 is also introduced later.

For the RCM described here, we simply used the outcome of the digital conversion as the performance function $P_e^{(d,t)}$, that is, the number of quantization steps in the quantification level $Q_e^{(d,t)}$. Similarly, R_{th} represents the difference in the number of quantization steps that $P_e^{(d,t)}$ must reach to trigger a path switch.

Overall, the advantage of this filtering technique is that it produces the desired effect (i.e., controlled proactivity) because it prevents minor changes in the medians from triggering unnecessary traffic relocations at S.

Socialized Route Control

The second facet of the social behavior of the algorithm relates to the dynamics of the median RTTs; more precisely,

---

[5] *We notice that with CP, e^best might be different from e'.*

Input: d – A target destination of network S
  $\{e\}$ – Set of egress links of network S
  $P_e^{(d,t)}$ – Performance function to reach d through e at time t

Output: e^{best} – The best egress link to reach target destination d

1: Wait for changes in $P_{e^{best}}^{(d,t)}$
2: **if** $P_{e^{best}}^{(d,t)} - P_e^{(d,t)} < R_{th}\ \forall e \neq e^{best}$ **then** go to Step 1
3: /* Egress link selection process for d */
4:   Choose e' as $P_{e'}^{(d,t)} = min\{P_e^{(d,t)}\}$
5:   Estimate the performance after switching the traffic
6:   **if** $P_{e^{best}}^{(d,t)} - P_{e'}^{(d,t)}{}_{Estimate} \geq R_{th}$ **then**
7:     Wait until $T_H = 0$ /* Hysteresis Switching Timer */
8:     Switch traffic toward d from e^{best} to e'
9:     $e^{best} \leftarrow e'$
10:    $P_{e^{best}}^{(d,t)} \leftarrow P_{e'}^{(d,t)}$
11: **end if**
12: /* End of egress link selection process for d */
13: Go to Step 1

■ **Algorithm 1.** *Randomized SRC algorithm.*

with how rapid the variations are in the median values that are typically computed by IRC systems using a sliding window. The motivation for this is that when the median values start to show rather quick variations, the algorithm must react so as to avoid a large number of traffic reassignments in a short timescale. Such RTT dynamics typically occur when several route controllers compete for the same resources, leading to situations where their traffic reassignments interfere with each other. To cope with this problem, we turn the second filter in Fig. 2 into an adaptive filter. This filter is endowed with an adaptive quantization step $\Delta^{(d,t)}$ for each target destination d that is automatically adjusted by the algorithm according to the evolution of the median RTTs. If the RTT conditions are smooth, the quantization step is small, and more proactivity is allowed by the controller $\mathbb{C}$. However, if the RTT conditions could lead to instability, the quantization step $\Delta^{(d,t)}$ automatically increases, so the number of changes in the values of $Q_e^{(d,t)}$ is diminished or even stopped until the network conditions become smooth again. This has the effect of desynchronizing only the competing route controllers. Therefore, the filtering technique outlined here allows a controller $\mathbb{C}$ to "sociably" decide whether to switch traffic to an alternative egress link or not, in the sense that the degree of proactivity of $\mathbb{C}$ is constantly adjusted by the adaptive nature of the second filter.

For the sake of simplicity, we focused here on the optimization of a single performance metric (the RTT), but the concept of SRC is general and can be extended to consider other metrics, such as available bandwidth, packet losses, and jitter. When multiple metrics are used, two straightforward approaches can be followed.

On the one hand, a combination of two or more metrics can be used in the same performance function $P_e^{(d,t)}$. For instance, [12] introduces a more general performance function based on a non-linear combination of the quantification level $Q_e^{(d,t)}$ and the available bandwidth (AB) in the egress links of the source network. This, in turn, can be extended to consider the AB along the entire path to a target destination d, using available bandwidth estimation techniques like the one described in [5]. With this approach, the weights of the different metrics combined in $P_e^{(d,t)}$ can be tuned on an application basis, for example, to prioritize the role of the AB over the RTTs (or vice versa) depending on the application type.

On the other hand, multiple performance functions $P_e^{(d,t)}$ can be used (e.g., one for each metric), and the selection of the best path for each target destination can be performed by sequentially comparing the performance functions $P_e^{(d,t)}$ and tie-breaking similarly to the BGP tie-breaking rules [7]. With this approach, the order in which the performance functions are compared can be tuned on an application basis. For example, a controller might select the path with the maximum AB, and if there is more than one path with the same AB, choose the one with the lowest RTT.

In either case, adaptive filtering techniques are required to prevent rapid variations in the performance metrics considered.

Randomization

Randomization is present in Algorithm 1 in two different ways: implicitly and explicitly. On the one hand, the route control decisions in Algorithm 1 are inherently stochastic for a number of reasons, for example, due to its adaptive features along time, the fact that different controllers might have configured different thresholds R_{th}, and others. On the other hand, we explicitly use a *hysteresis switching timer* T_H that we introduced in a previous work [13] and that guarantees a random hysteresis period after each traffic relocation. More precisely, traffic toward a given destination d cannot be relocated until the random and decreasing timer $T_H = 0$. A similar approach was used in [5] for one of the randomized algorithms presented there.

Performance Evaluation

The performance of our SRC strategy is compared against that obtained with:
• Randomized IRC
• Default IGP/BGP routing

Evaluation Methodology and Simulation Set Up

The simulation tests were performed using the event-driven simulator J-Sim [14]. All the functionalities of the route controllers were developed on top of the IGP/BGP implementations available in this platform.

Network Topology — The network topology was built using the Boston University Representative Internet Topology gEnerator (BRITE) [15]. The topology was generated using the Waxman model with (α, β) set to (0.15, 0.2) [16], and it was composed of 100 domains with a ratio of domains to inter-domain links of 1:3. This simulated network aims at representing a set of ISPs that can provide connectivity and reachability to customers operating stub networks. We assume that all ISPs operate points of presence (PoPs) through which the stub networks are connected. We considered 12 uniformly distributed stub networks across the domain-level topology as the traffic sources toward the set of target destinations. These source networks are connected to the routers located at the PoPs of three different ISPs. We considered triple-homed stub networks given that significant performance improvements are not expected from higher degrees of multihoming [1]. For the stub networks containing target destinations, we considered 25 uniformly distributed destinations across the domain-level topology. This offers an emulation of $12 \times 25 = 300$ IRC flows competing for the same network resources during the simulation run time.

Furthermore, given that IRC solutions operate in short timescales, we assumed that the domain-level topology remains invariant during the simulation run time.
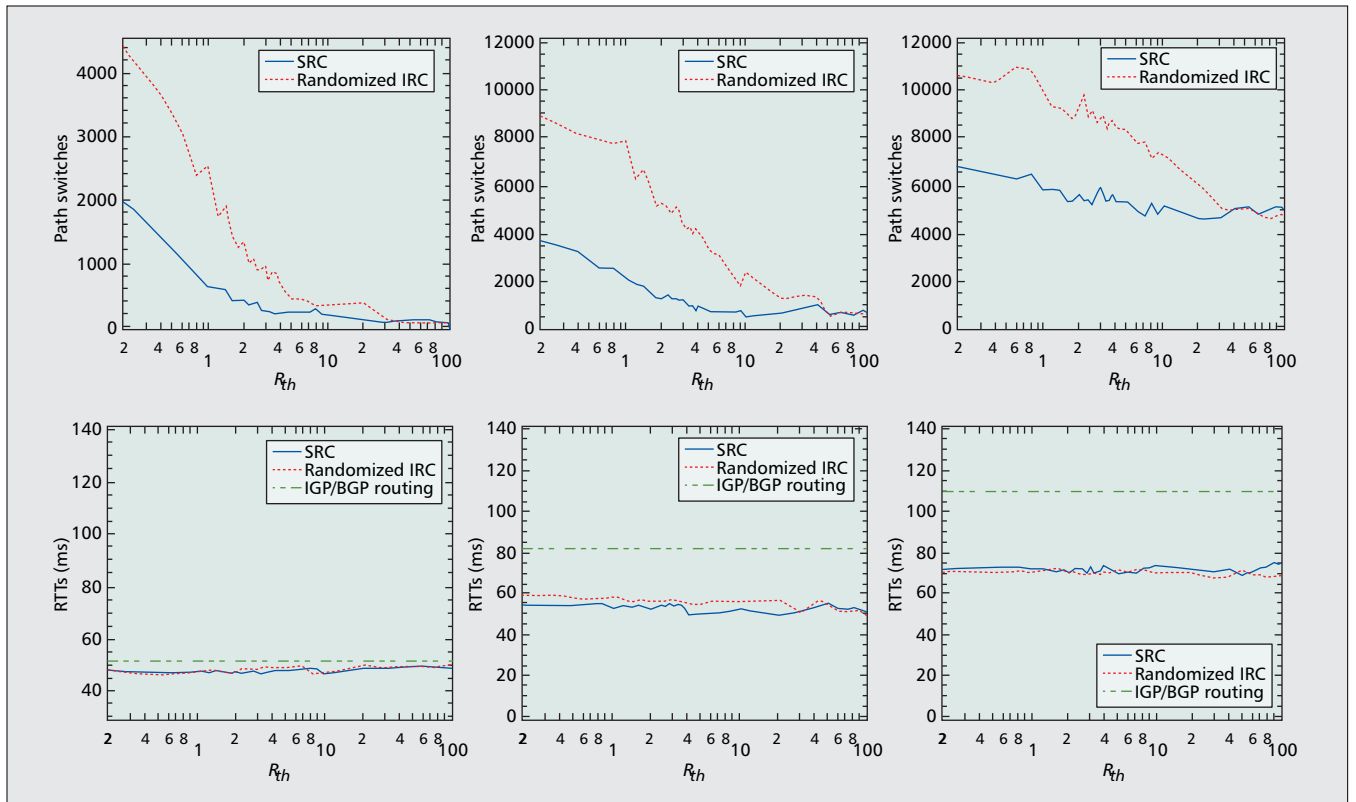
■ Figure 3. *Number of path switches (top) and <RTTs> (bottom) for* L = 0.450 (left), L = 0.675 (center), *and* L = 0.900 (right).

Simulation Scenarios — We run the same simulations separately using three different scenarios:
• Default IGP/BGP routing, where BGP routers choose their best routes based on the shortest AS-path
• BGP combined with the SRC strategy at the 12 source domains
• BGP combined with randomized IRC systems at the 12 source domains

For a more comprehensive comparison between the different route control strategies, we performed the simulations for three different network loads. We considered the following load factors (*L*):
• *L* = 0.450, low load corresponding to an average occupancy of 45 percent of the egress links capacity
• *L* = 0.675, medium load corresponding to an average occupancy of 67.5 percent of the egress links capacity
• *L* = 0.900, high load corresponding to an average occupancy of 90 percent of the egress links capacity

Simulation Conditions — The simulation tests were conducted using traffic aggregates sent from the source domains to each target destination *d*. These traffic aggregates were composed of a variable number of multiplexed Pareto flows as a way to generate the traffic demands, as well as to control the network load during the tests. The flow arrivals were modeled according to a Poisson process and were independently and uniformly distributed during the simulation run time. This approach aims at generating sufficient traffic variability to support the assessment of the different route control strategies.

In addition, we used the following method to generate traffic demands for the remaining Internet traffic, usually referred to as background traffic. We started by randomly picking four nodes in the network. The first one chosen acts as the origin (O) node, and the remaining three nodes act as destinations (D) of the background traffic. We assigned one Pareto flow for each O-D pair. This process continues until all the nodes

are assigned with three outgoing flows (including those in the multihomed stub domains and those in the ISPs). All background connections were active during the simulation run time.

Furthermore, the frequency and size of the probes sent by the route controllers were correlated with the outbound traffic being controlled, just as conventional route controllers do today [2–4].

Finally, we assume that the route controllers have pre-established performance bounds (i.e., the MTTs) for the traffic under control. For instance, the recommendation G.114 of the International Telecommunication Union-Telecommunication Standardization Sector-(ITU-T) suggests a one-way-delay (OWD) bound of 150 milliseconds to maintain a high quality VoIP communication over the Internet. Thus, for VoIP traffic, the maximum RTT tolerated was chosen as twice this OWD bound, that is, 300 ms.

Objectives of the Performance Evaluation
Our evaluations have two main objectives.

Assess the Number of Path Switches — The first objective of the simulation study is to demonstrate that the sociable nature of our SRC strategy contributes to drastically reducing the potential interference between competing route controllers. To this end, we compared the number of path switches that occurred during the simulation run time for the 300 competing IRC flows for the SRC and randomized IRC scenarios. The number of path switches is obtained by adding the number of route changes that are required to meet the desired RTT bound for each target destination *d*.

It is worth emphasizing that in both the randomized IRC and SRC strategies, the route controllers operate independently and compete for the same network resources. This allows us to evaluate the overall impact on the traffic caused by the interference between several standalone route con-
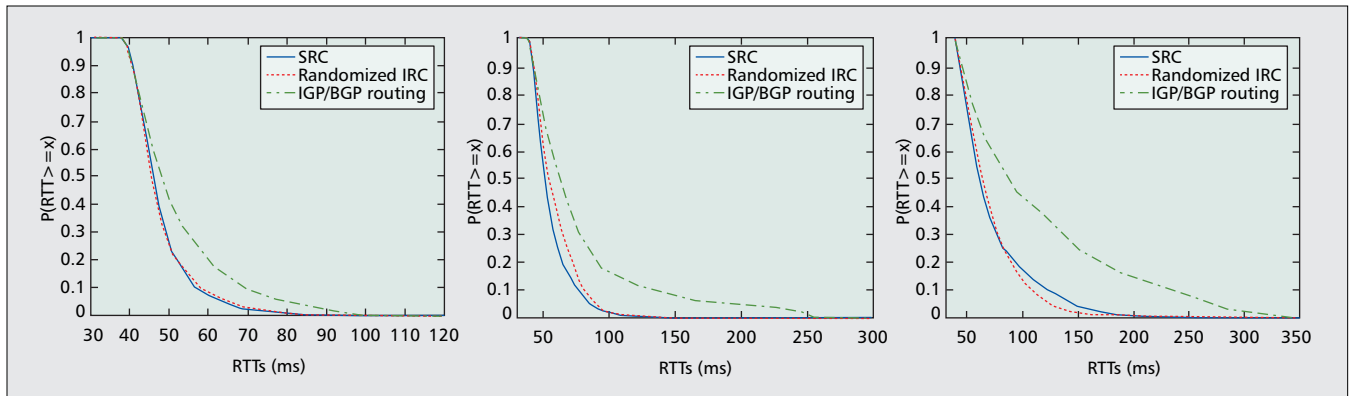
■ Figure 4. *Complementary cumulative distribution function (CCDF) of the RTTs for the 300 competing IRC flows, for $R_{th} = 1$, and for L = 0.450 (left), L = 0.675 (center), and L = 0.900 (right).*

trollers running at different stub domains. Thus, when analyzing the results for the different route control strategies, it is important to keep in mind that we take into account all the competing route controllers present in the network.

To contrast the number of path switches under fair conditions, we made the following decisions. First, both the randomized IRC and SRC controllers are endowed with the same (explicit) randomization technique [5, 13]. This approach avoids the appearance of persistent oscillations that might lead to a large number of path switches in the case of conventional IRC [5]. Second, both types of controllers follow a controlled proactivity approach. We have conducted the simulations modeling the same triggering condition R_{th} for both of them. The main difference is that in the SRC case, the social adaptability of the controllers can result in the trigger being reached more often, or less often, depending on the variability of the RTTs on the network.

End-to-End Traffic Performance — The second objective of the simulation study is to demonstrate that the drastic reduction in the number of path switches obtained with our SRC strategy can be achieved without penalizing the end-to-end traffic performance. To this end, we compared the RTTs obtained for the 300 flows in the three different scenarios, namely, default IGP/BGP, SRC, and randomized IRC.

Main Results

The top of Fig. 3 illustrates the total number of path switches performed by both the randomized IRC and SRC strategies, in all the stub networks, and for the three different load factors: $L = 0.450$ (left), $L = 0.675$ (center), and $L = 0.900$ (right). The number of path switches is contrasted for different triggering conditions, that is, for different values of the threshold R_{th} (shown on a logarithmic scale).

Several conclusions can be drawn from the results shown in Fig. 3. In the first place, the results confirm that SRC drastically reduces the number of path switches compared to a randomized IRC technique.[6] An important result is that the reductions are significant for all the load factors assessed. For instance, when compared with randomized IRC, our SRC strategy contributes to reductions of up to:

• 77 percent for $R_{th} = 1$ and 71 percent for $R_{th} = 2$ when $L = 0.450$
• 75 percent for $R_{th} = 1$ and 74 percent for $R_{th} = 2$ when $L = 0.675$
• 34 percent for $R_{th} = 1$ and 36 percent for $R_{th} = 2$ when $L = 0.900$

The second observation is that the reductions in the number of path switches offered by the SRC strategy become more and more evident as the proactivity of the controllers increases, that is, for low values of R_{th}, which is precisely the region where IRC solutions operate today. It is worth recalling that these results were obtained when both route control strategies were complemented by the same randomized decisions. This confirms that in a competitive environment, SRC is much more effective than pure randomization in reducing the potential interference between route controllers.

On the other hand, our results show that when the route control strategies become less proactive, that is, for higher values of R_{th}, randomized IRC and SRC tend to behave comparatively the same so SRC does not introduce any benefit over a randomized IRC technique.

To assess the effectiveness of SRC, it is mandatory to confirm that the reductions obtained in the number of path switches are not excessive, resulting in a negative impact on the end-to-end traffic performance. To this end, we first analyze the performance of randomized IRC and our SRC "globally," that is, by averaging the RTTs obtained by "all" competing route controllers. This is shown at the bottom of Fig. 3 and in Fig. 4. The end-to-end performance obtained by "each" route controller individually, is shown in Fig. 5.

The bottom of Fig. 3 reveals that as expected, both SRC and randomized IRC perform much better than IGP/BGP for all values of L and R_{th}, and the improvements in the achieved performance become more evident as the network utilization increases. In particular, SRC is capable of improving the $\langle RTTs \rangle$[7] by more than 40 percent for $L = 0.675$ and by more than 35 percent for $L = 0.900$ when compared with IGP/BGP.

Moreover, the $\langle RTTs \rangle$ obtained by SRC and IRC are comparatively the same and particularly for $L = 0.675$, SRC not only drastically reduces the number of path switches, but also improves the end-to-end performance for almost all the triggering conditions assessed. It is worth emphasizing that a low value of R_{th} together with a load factor of $L = 0.675$ reasonably reflect the conditions in which IRC currently operates in the Internet.

Our results also reveal an important aspect: by allowing more path switches, some route controllers can improve slightly their end-to-end performance, but such actions have no major effect on the overall $\langle RTTs \rangle$. Indeed, a certain number of path switches is always required, and this number of path switches is what actually ensures the average performance observed in the RTTs at the bottom of the Fig. 3 (this becomes clear as the proactivity decreases).

By analyzing Fig. 3 as a whole, it becomes evident that the

---

[6] *Clearly, no results are shown for the default IGP/BGP routing scenario here because BGP does not perform path switching actively.*

[7] *As mentioned previously, this average is computed over the RTTs obtained by all competing route controllers in the network.*
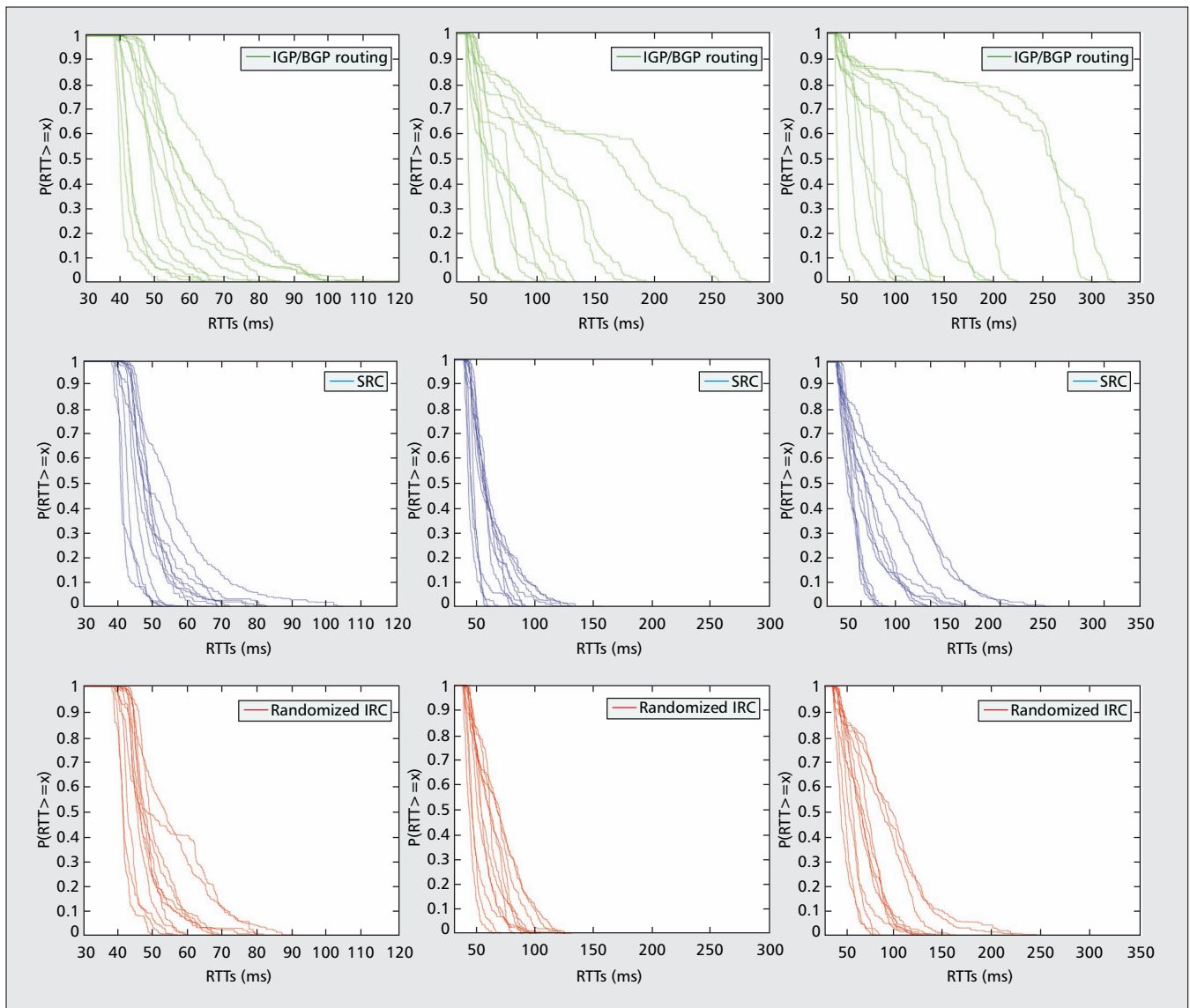
Figure 5. *CCDFs for IGP/BGP routing (top), SRC (center), and randomized IRC (bottom), for* L = 0.450 *(left),* L = 0.675 *(center), and* L = 0.900 *(right).*

selection of the best triggering condition actually depends on the load present in the network. For this particular case, the best trade-offs are $R_{th} = 30$ for $L = 0.450$, $R_{th} = 10$ for $L = 0.675$, and $R_{th} = 7$ for $L = 0.900$, which is a reasonable progression to lower values of R_{th} because the route controllers require less proactivity when the network utilization is low. The corollary is that the triggering condition should be adaptively adjusted as well, depending on the amount of traffic carried through the egress links of the domain. We plan to investigate this in the future.

Figure 4 compares the distribution of the RTTs obtained by IGP/BGP, SRC, and randomized IRC for the 300 competing IRC flows, for the three different load factors assessed, and for $R_{th} = 1$, which as mentioned above is in the range of operation of the IRC solutions presently deployed in the Internet. To facilitate the interpretation of the results, we use the complementary cumulative distribution function (CCDF).

An important observation is that under high egress link utilization, that is, $L = 0.900$, there is a fraction of $\langle RTTs \rangle$ for which the bound of 300 ms is exceeded in the case of IGP/BGP; whereas both SRC and the randomized IRC fulfill the targeted bound.

To complete the analysis, Fig. 5 provides a more granular picture than Fig. 4 because it shows the CCDFs of the RTTs obtained by each of the 12 competing route controllers. The figure shows the results for the three studied scenarios and for all the load factors assessed when $R_{th} = 1$. Our results show that the targeted bound of 300 ms is satisfied by both SRC and randomized IRC in all cases and for all controllers. IGP/BGP, however, shows a distribution of large delays given that the shortest AS-paths are not necessarily the best performing paths. Figure 5 also shows that when considering boxes individually, randomized IRC achieves slightly better end-to-end performance for some of them but at the price of a much larger number of path switches:

- $\approx \approx 435$ percent larger for $L = 0.450$
- $\approx \approx 400$ percent larger for $L = 0.675$
- $\approx \approx 80$ percent larger for $L = 0.900$ when $R_{th} = 1$.

Conclusion

In this article, we examined the strengths and weaknesses of randomized IRC techniques in a competitive environment. We proposed a way to blend randomization with a sociable route control (SRC) strategy, where by sociable, we mean a route control strategy that explicitly considers the potential

implications of its decisions in the performance of the network and with the ability to adaptively restrain its intrinsic selfishness depending on the network conditions. We have shown that in a competitive scenario, our SRC strategy is capable of drastically reducing the potential interference between controllers without penalizing the end-to-end traffic performance. This makes SRC more scalable and promising than pure randomization, given the proliferation of IRC systems in the Internet.

SRC strategies, like the one described in this article, also have a number of practical advantages; for example, they do not require any kind of coordination between the competing IRC boxes; and they can be supported by a lightweight software implementation based on well-known filtering techniques, with no additional requirements to be adopted other than a software upgrade of existing IRC systems.

Among the open issues in the area, the most important is the lack of a stochastic model characterizing the distribution of path switches in a competitive environment. Studies like [5] have shown that randomized techniques are effective in desynchronizing some route controllers when their measurement windows are sufficiently overlapped; however, they cannot guarantee stability. Only after characterizing the distribution of path switches will it be possible to formally study the local and global stability aspects of competitive IRC. Furthermore, the proposals and results described here apply to the optimization of VoIP traffic, but the conception of blending randomization with an SRC strategy is general in scope so our work can be extended to control other kinds of traffic flows concurrently, as well as consider other performance metrics besides the RTT.

References

[1] A. Akella *et al.*, "A Measurement-Based Analysis of Multihoming," *Proc. ACM SIGCOMM*, Karlsruhe, Germany, Aug. 2003.
[2] Avaya, Inc., "Converged Network Analyzer."
[3] Cisco Systems, Inc., "Optimized Edge Routing."
[4] Internap Networks, Inc., "Flow Control Platform."
[5] R. Gao, C. Dovrolis, and E. W. Zegura, "Avoiding Oscillations Due to Intelligent Route Control Systems," *Proc. IEEE INFOCOM 2006*, Barcelona, Spain, Apr. 2006.
[6] C. Labovitz *et al.*, "Delayed Internet Routing Convergence," *Proc. ACM SIGCOMM*, Stockholm, Sweden, Aug. 2000.
[7] M. Yannuzzi, X. Masip-Bruin, and O. Bonaventure, "Open Issues in Interdomain Routing: A Survey," *IEEE Network*, vol. 19, no. 6, Nov.–Dec. 2005, pp. 49–56.
[8] A. Akella *et al.*, "A Comparison of Overlay Routing and Multihoming Route Control," *Proc. ACM SIGCOMM*, Portland, OR, Aug. 2004.
[9] D. K. Goldenberg *et al.*, "Optimizing Cost and Performance for Multihoming," *Proc. ACM SIGCOMM*, Portland, OR, Aug. 2004.
[10] F. Guo *et al.*, "Experiences in Building a Multihoming Load Balancing System," *Proc. IEEE INFOCOM '04*, Hong Kong, China, Mar. 2004.
[11] A. Akella, S. Seshan, and A. Shaikh, "Multihoming Performance Benefits: An Experimental Evaluation of Practical Enterprise Strategies," *USENIX Annual Technical Conf.*, Boston, MA, June 2004.
[12] M. Yannuzzi, "Strategies for Internet Route Control: Past, Present, and Future," Ph.D. diss., Tech. Unive. of Catalonia, Barcelona, Spain, 2007.
[13] M. Yannuzzi *et al.*, "A Proposal for Inter-Domain QoS Routing Based on Distributed Overlay Entities and QBGP," *Proc. QoFIS '04*, LNCS 3266, Barcelona, Spain, Oct. 2004.
[14] J-Sim homepage; http://www.j-sim.org
[15] A. Medina *et al.*, "BRITE: An Approach to Universal Topology Generation," *Proc. MASCOTS*, Aug. 2001.
[16] B. Waxman, "Routing of Multipoint Connections," *IEEE JSAC*, Dec. 1988.

Biographies

MARCELO YANNUZZI (yannuzzi@ac.upc.edu) received a degree in electrical engineering from the University of the Republic (UdelaR), Uruguay, in 2001, and DEA (M.Sc.) and Ph.D. degrees in computer science from the Department of Computer Architecture, Technical University of Catalonia (UPC), Spain, in 2005 and 2007, respectively. He is with the Advanced Network Architectures Lab at UPC, where he is an assistant professor. He held previous positions with the Physics Department of the School of Engineering, UdelaR, from 1997 to 2003, and with the Electrical Engineering Department of the same university from 2003 until 2006. He worked in industry for 10 years at the national telco in Uruguay (1993–2003).

XAVI MASIP-BRUIN (xmasip@ac.upc.edu) received M.S. and Ph.D. degrees from UPC, both in telecommunications engineering, in 1997 and 2003, respectively. He is currently an associate professor of computer science at UPC. His current research interests are in broadband communications, QoS management and provision, and traffic engineering. His publications include around 60 papers in national and international refereed journals and conferences. Since 2000 he has participated in many research projects: IST projects E-NEXT, NOBEL, and EuQoS; and Spanish research projects SABA, SABA2, SAM, and TRIPODE.

EVA MARIN-TORDERA (eva@ac.upc.edu) received M.S. degrees in physics in 1993 and electronic engineering in 1998, both from Barcelona University, and a Ph.D. from UPC in 2007, where she works as an assistant professor. She has published many papers in national and international conferences. Her main interests focus on QoS provisioning and optical networks. She is now actively participating in the BONE and DICONET international projects, and in the national project CATARO.

JORDI DOMINGO-PASCUAL (jordid@ac.upc.edu) is a full professor of computer science and communications at UPC. He is co-founder of and a researcher at the Advanced Broadband Communications Center (CCABA) of the university. His research topics are broadband communications and applications, IP/ATM integration, QoS management and provision, traffic engineering, IP traffic analysis and characterization, and QoS measurements.

ALEXANDRE FONTE (afonte@dei.uc.pt) graduated in electrical engineering from the University of Coimbra, Portugal, in 1995, and received his M.Sc. degree in electronic and telecommunications engineering (distributed systems specialty) from the University of Aveiro, Portugal, in 2000. He is currently a Ph.D. student in computer engineering at the Department of Informatics Engineering, University of Coimbra. His Ph.D. research activity is focused on interdomain quality of service routing and traffic engineering in IP networks.

EDMUNDO MONTEIRO (edmundo@dei.uc.pt) is an associate professor at the University of Coimbra, Portugal, from which he graduated in 1984 and received a Ph.D. in electrical engineering (computer specialty) in 1995. His research interests are computer communications, QoS, mobility, routing, resilience, and security.