Interdomain Routing with Negotiation

Ratul Mahajan David Wetherall Thomas Anderson

{ratul,djw,tom}@cs.washington.edu

University of Washington

May, 2004

Abstract

Current interdomain routing policies are largely based on information local to each ISP, in part due to competitive concerns and the lack of effective information sharing mechanisms. This can lead to routes that are sub-optimal and even to routes that oscillate. We explore a setting in which two neighboring ISPs negotiate to determine the path of the traffic they exchange. We first ask the basic question: is there an incentive to negotiate? The incentive exists only if *both* ISPs benefit relative to routing based on local information. Through simulation with over sixty measured ISP topologies, we find that negotiation is useful for both latency reduction and hotspot avoidance. Interestingly, we find that global optimization is undesirable in the sense that one ISP often suffers to benefit the other. Based on our results, we design and evaluate a negotiation protocol which works within the real-world constraints of competing and independently managed ISPs. Specifically, our protocol reveals little information and works even when ISPs have different optimization criteria. We find that it achieves routing performance comparable to that of global optimization using complete information from both ISPs.

1 Introduction

One of the hallmarks of Internet routing is that ISPs compete with one another as independently managed entities at the same time that they must cooperate to provide connectivity. This competitive relationship greatly complicates the task of routing. Because of it, ISPs do not freely share information with each other about the internal state of their network. One result is that ISPs tend to make self-interested routing decisions based on local rather than global information. For example, consider two ISPs connected to each other at a set of peering points scattered around the world. By one common convention, each will send traffic to the other via whichever peering point is most advantageous for itself, say the nearest for the "early-exit" policy. In return, the ISPs must accept traffic at whichever peering point is most advantageous to the other. That is, each ISP acts unilaterally for its benefit within an overall framework contractually agreed to by both ISPs.

Unfortunately, the combination of self-interested decision making and the lack of a global view can lead to sub-optimal Interet paths [22], and even unpredictable results [13]. Paths can be sub-optimal because decisions that appear locally sound may have adverse global effects. For instance, early-exit routing may not send packets in the direction of the ultimate destination. Behavior can be unpredictable because the actions of one ISP can have an unintended influence on the other and vice versa, and in the worst case cycles of influence can lead to oscillations.

In this paper we explore an alternative approach in which routing between ISPs is managed through explicit negotiation. Our focus is on an important subset of the overall problem: routing between two neighboring ISPs. With negotiation, these ISPs share information with each other in a controlled manner and jointly agree on a mutually-beneficial set of routes for traffic flows sent between them. This has the potential to counteract both the effects of sub-optimal paths and the unpredictable consequences of individual actions. The approach differs from both unilateral decision-making that disregards global consequences (as is mostly done today) and social decision-making that disregards local consequences (which is the hypothetical, optimal operating point that occurs if all ISPs act as a single global ISP). We use these other schemes as points of comparison.

Of course, ISPs can influence each other's route selection to some extent today through BGP-level hints such as multi-exit discriminators (MEDs) and AS-path pre-pending. This influence is, however, mostly indirect and often governed by trial and error, so operators also informally work with each other to prevent or resolve routing problems. As a concrete example, we learned of an incident involving two large ISPs peering in two locations. When one of the peering links unexpectedly became congested, one ISP reacted by shifting traffic off the overloaded link and on to the other peering link. But this change caused a link inside the other ISP to become congested, prompting it to

move traffic back to the other peering link. This impacted the first ISP in a cycle of influence that continued for two days before the ISPs jointly realized the source of the problem and manually negotiated an acceptable solution. This example serves to highlight another crucial advantage of negotiation, when automated: it could relieve operators from the time-consuming and error-prone task of reacting to short-term problems.

We make two contributions in this paper. First, we quantitatively evaluate the potential benefits of explicit negotiation in practice. Using simulation with over sixty measured ISP topologies, we show that for both latency and bandwidth metrics, ISPs can work together such that both of them benefit. For latency measures, this benefit is small on average, with half of the ISP-pairs achieving only 4% latency reduction, but can be significant for a small fraction of individual flows with circuitous default paths. For bandwidth measures, this benefit is more often substantial, lessening the likelihood that the actions of one ISP will adversely impact the other. We also show that, unlike negotiated solutions, globally optimal solutions that optimize across both ISPs as a single larger system have the undesirable property that one ISP often suffers for the global good. In an environment where resources are owned by different autonomous entities, this seems unrealistic. By contrast, when ISPs accept slightly worse routes for some flows in exchange for larger benefits for other flows, they can often reach an operating point that benefits both sides and is also close to the globally optimal solution in terms of quality.

Our second contribution is a negotiation protocol, called NP. It is a first step towards a practical inter-ISP negotiation protocol, having several properties that make it a good fit for the context of inter-domain routing. First, it is flexible enough for two ISPs to reach different operating points based on their specific relationship. Second, it allows ISPs to optimize for different properties at their discretion, e.g., latency versus bandwidth. Third, it requires ISPs to share relatively little information with each other: opaque route preferences rather than the internal state of the network. Trivially, it allows an ISP to ensure that it is no worse off than if it were using unilateral routing decisions, so that negotiating carries no risk. Through simulation, we show that the routing quality that can be achieved with this protocol, if the ISPs agree to use it, is comparable to the best that it is possible in an unrealistic setting where all ISPs act as a single larger ISP.

Our work can be seen as part of the body of work that examines the "price of anarchy" or the cost of unilateral, selfish decisions in the Internet [21, 20, 14]. This work mostly studies the extreme points of complete selfishness and global optimality, both of which may be undesirable in practice. Compared to it, we explore the viability of a middle ground acceptable to all parties.

The rest of the paper is organized as follows. In Section 2, we motivate the need for negotiation using two examples. We list the necessary requirements for an inter-ISP negotiation protocol in Section 3, and describe our protocol in Section 4. In Section 5 we empirically demonstrate the benefits of negotiation. We discuss related work in Section 6 and conclude in Section 7.

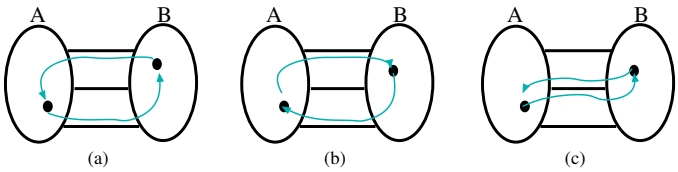

Figure 1: Illustration of the need for negotiation for performance tuning. (a) The default (early-exit) scenario. (b) The traffic pattern with MEDs (late-exit). (c) A negotiated solution beneficial for both ISPs.

2 Motivation

In this section, we use two scenarios to motivate how negotiation can help ISPs manage their routing. In each case we explain why existing BGP mechanisms are not sufficient to achieve the desired outcome.

Our first example concerns the tuning of traffic exchanged between two ISPs to use resources more efficiently or improve performance. Consider the two ISPs shown in Figure 1. It is common for upstream ISPs to use "early-exit" routing (where the nearest exit point is used) to transfer traffic to the downstream ISPs [23]. This minimizes resource usage in the upstream network. However, the gains of this strategy vanish when one considers traffic flowing in the reverse direction, if the downstream ISP also uses early-exit routing. This situation is shown in Figure 1a.

When both directions of traffic are considered, early-exit routing can lead to greater resource consumption for both ISPs than if another peering link is chosen judiciously. This is because the "early exit" may temporarily route traffic away from the ultimate destination. For example, in Figure 1c, the middle peering link is a better choice for both ISPs. Johari and Tsitsiklis [14] provide a simple graphical argument showing that the distance of early-exit routing under certain topological assumptions can be up to three times the distance of the optimal routing, though we will find that it is much less in practice the majority of the time.

Unfortunately, there is no easy way to achieve the optimized configuration of Figure 1c with BGP. The Multi-Exit Discriminator (MED) mechanism allows downstream ISPs to signal ingress link preferences to their upstream ISPs. But the use of MEDs to select peering links will lead to the "late-exit" case depicted in Figure 1b, in which traffic enters on the link that is closest to the destination. This situation is simply the reverse of early-exit routing. If both sides follow the same policy, the

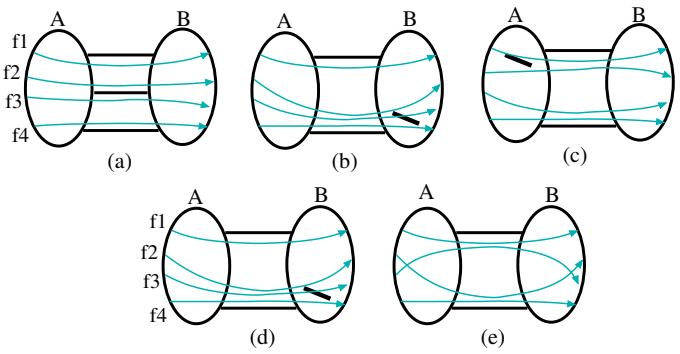

Figure 2: Illustration of the need for negotiation in response to failures. (a) The stable (no failure) scenario. (b) The situation after ISP-A responds to the failure of the middle peering link by moving the traffic to the bottom one, which congests an internal link in ISP-B. (c) ISP-B reacts by moving some traffic from the bottom peering link to the top one, which congests an internal link inside ISP-A. (d) Not knowing the cause of the traffic movement in (c), ISP-A reacts to the internal congestion by moving the traffic back to the bottom link, which again congests the internal link of ISP-B. (e) A negotiated solution that is acceptable to both ISPs.

result is the same circuitous round trip path, only in the reverse direction.

Our second example occurs when there are unexpected changes in the topology or its traffic makeup, for example, when a peering link fails or with flash crowds and denial-of-service floods. It has the same flavor as the incident described in the introduction, where the ISPs are unable to achieve a reasonable solution because they lack insight into the state of the other's network. Consider the two ISPs shown in Figure 2, with traffic flowing from ISP-A to ISP-B. Now assume that a peering link between the two ISPs fails. ISP-A re-routes the affected traffic based on the conditions in its own network. This change leads to congestion inside ISP-B, which reacts by re-distributing incoming traffic across peering links (using MEDs or selective prefix announcements, for instance). This action by ISP-B overloads a link inside ISP-A. At this point ISP-A reacts by undoing ISP-B's change (using local preferences, for instance) or perhaps shifting other traffic that still causes problems inside ISP-B. The result is to return to the original situation of Figure 2b and continue the cycle.

Figure 2e shows a solution that is acceptable to both ISPs. In general, there is no easy way in BGP to achieve this desired configuration. The ISPs might try to use MEDs. ISP-B needs to specify that the preferred entrance for $f3$ is the top peering link, and that for $f2$ is the bottom one. But ISP-B has no basis for differentiating this desired configuration from that of preferring $f3$ on the bottom link and $f2$ on the top one, as it depends on the impact of these flows on the upstream ISP. Alternately, the upstream ISP can use local preferences to attain the desired traffic flow. However, ISP-A has no reason to send $f3$ to the top peering link since its own network can handle that both $f2$ and $f3$ leave via the bottom link.

While the example above describes a peering link failure, negotiation between ISPs is useful even when an internal link fails or becomes overloaded. To achieve certain traffic engineering goals, an ISP is often required to change the paths for the traffic it exchanges with other ISPs; it is often not sufficient to reroute its own traffic internally because that constitutes only a fraction of the total traffic [7, 25]. Unilaterally rerouting external traffic impacts neighboring ISPs, at which point one may run into the problem described above. The way out of this dilemma is, of course, negotiating the desired change with the neighboring ISP.

The logical extension of the argument above is that negotiation is required not only for neighboring ISPs but for all the ISPs in the path of the traffic. While we do believe this to be the case, in this paper we restrict our attention to the more tractable but still challenging two-ISP case, and leave Internet-wide negotiation as future work. Two-ISP negotiation already provides significant leverage. It can be used whenever the effects of the negotiation are not visible beyond the pair of ISPs, which can be achieved simply by not changing the external links over which the upstream ISP receives traffic and the downstream ISP sends traffic. Winick *et al.* estimate that the fraction of such traffic flowing across large ISPs is very high (over 90% for AT&T), mainly because many customers of these ISPs are singly-homed [25].

3 Design Requirements

In this section, we outline the requirements for an interdomain negotiation protocol. The goal of the protocol is to let two ISPs manage the traffic that flows between them in a way that is acceptable to both. The requirements of the protocol are driven by our problem domain: the ISPs are independently run and compete with one another, yet they are willing to cooperate in a limited way when it will benefit themselves and their customers.

We believe that a negotiation protocol for inter-ISP negotiation must have the following properties:

- **Flexible Outcomes:** ISPs are autonomous agents, and different pairs of ISPs have different relationships such as peer-peer, sibling-sibling, and customer-provider [11]. The interaction between the ISPs is governed by their relationship. For example, when negotiating with their customers, ISPs may prefer a different outcome than when doing so with their peers. Instead of defining a mechanism that produces a deterministic output given some input, the protocol should provide a flexible framework within which

ISPs will negotiate in the context of their overall relationship. Different ISP-pairs may arrive at different solutions based on their specific situation. In this regard, the negotiation between the ISPs can be seen as a "tussle" [6].

- **Controlled information disclosure:** While negotiation is a form of cooperation, the fact remains that ISPs are competing entities. For example, they often compete for the same customers. As a result, ISPs are loathe to share topological and performance information of their network with other ISPs. Negotiation protocols need to respect this constraint, and work with inputs that do not directly reveal unwanted information about one ISP to another. This sensitivity also extends to pricing information, since an ISP may not wish to tell its competitor the true marginal cost of carrying traffic [8]. We handle this concern by working with opaque preference classes, rather than transparent metrics such as latency or cost.

- **Support for different objective functions:** Different ISPs optimize their networks for different objectives, and as a result their motivation for negotiating will differ. For instance, while ISPs with capacity constraints may want to avoid overloaded links and increase resource usage efficiency, ISPs with over-provisioned networks may want to improve performance by reducing latency and jitter. Others may want the best routes for their preferred customers. Moreover, there are bound to be further considerations of which we cannot be aware. Thus the negotiation protocol should be agnostic towards the objective function used by a particular ISP. As before, we achieve this by working with opaque preference classes and letting the ISP map these classes to whatever objective function it uses internally. This has the benefit that ISPs need not reveal their optimization criteria.

The requirement for flexibility implies that all kinds of outcomes should be possible, including the social optimum solution that treats both ISPs as if they were a single larger system with a common optimization metric. For our work, however, the most interesting space is that of "win-win" solutions, where neither ISP loses. The social optimum may cause one ISP to lose compared to the default situation in which no negotiation is performed. Since the motivation for negotiation is for an ISP to gain for itself and its customers, ISPs will be less willing to negotiate if they risk losing as well as gaining. This balance can be altered by side payments, where the gaining ISP compensates the losing one, but we leave this issue for future work.

It is also desirable that the outcomes, whatever they might be, are *Pareto-optimal*. A solution is Pareto-optimal if there is no other solution that is strictly better for one ISP and at least as good for the other [10]. This criterion rules out solutions with obvious wastage, i.e., those that are worse for both the ISPs. (The current Internet is often not Pareto-optimal.) There are typically many Pareto-optimal solutions in the system, of which the social optimum is one. Of these, it is intuitively appealing to aim for a solution that is fair. However, traditional definitions of fairness are problematic in our setting because they require the gains of ISPs to be compared; this is not possible in a meaningful way if the different ISPs are using different metrics.

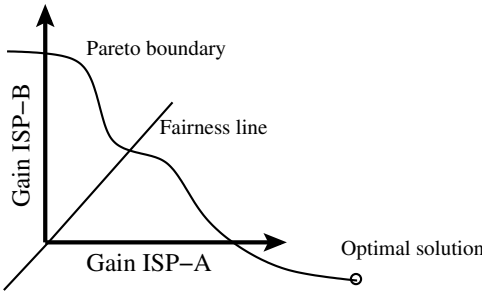

Figure 3: Illustration of the relationship between Pareto-optimal, optimal and fair division solutions.

Figure 3 illustrates the relationship between Pareto-optimal, social optimal and fair solutions. Pareto-optimal solutions lie along the Pareto-boundary. There are no solutions to the top and right of this boundary, so any deviation from this line hurts both ISPs. Win-win solutions lie in the region that is positive for both the axes, so that both ISPs gain. Fairness and social optimality are defined only when the gains of the participants can be compared. Fairer solutions are closer to the $x = y$ line. The social optimum solution lies somewhere along the Pareto-boundary and is the choice that has the highest sum of ISP gains. The figure shows one possible location for it, in which ISP-B actually loses.

4 The Negotiation Protocol

In this section, we describe our negotiation protocol NP. Instead of using transparent metrics such as latency, NP operates using $2P+1$ opaque preference classes in the integral range $[-P, P]$. Internal ISP metrics are mapped to this range as described below. P is chosen to be large enough to differentiate options with substantially different internal metric values, but small enough to avoid unnecessary information leakage. The use of these preference classes serves two purposes. First, it enables negotiation even when the optimization metrics are different. Second, ISPs reveal less information about their networks since neither the optimization criterion nor the mapping process is revealed.

Before the negotiation starts, ISPs map options to preference classes based on their own optimization criterion. An *option* is one of the several ways in which a flow can be routed. In our two ISP scenario, an option corresponds to the choice of peering link for a particular flow. So if there are three peering

links between the ISPs, each flow has three options. A *flow* is a collection of packets with the same source and destination IP prefix. The granularity of this collection can range from host-host to PoP-PoP.

The mapping from options to preference classes is done relative to the default option for the flow, which is the path it would have taken in the absence of negotiation. The default is always mapped to preference class 0. The two ISPs need not agree on the default path of a flow. The preference class for a non-default option reflects its utility compared to the default. Options that are better (worse) than the default are mapped to positive (negative) preference classes, and better options are ranked higher. ISPs are free to choose any mapping methodology that satisfies these constraints. Better solutions are obtained when, instead of simply ordering the options, ISPs assign preference classes that reflect the magnitude of the difference in their utility to the ISP.

Our negotiation framework is shown in Figure 4. ISPs start negotiating by exchanging their preference lists. Negotiation works in rounds. In each round, one ISP proposes an option, and the other decides if that option is acceptable. Below, we discuss the various steps in the negotiation and describe example implementations of them. The choice of implementations is agreed on in advance by the negotiating ISPs.

- *Decide turn:* Decide which ISP gets to propose an option in the current round. This function is agreed on by the ISPs before the negotiation begins. One possibility is that ISPs alternate. Another possibility that is geared towards fairness is that the ISP with the lower cumulative gain (computed using the sum of preference classes for the flows negotiated so far) gets the next turn. Yet another possibility is to randomly choose which ISP gets the next turn.

- *Pick an option:* The ISP whose turn it is picks an option to propose. For successful negotiation this ISP should take into account the preferences of the other ISP too. Otherwise, the second ISP is likely to discount the preferences of the first ISP, leading to solutions that are akin to default routing. A method that maximizes social gain is to pick an option from the set that maximizes the sum of preference classes of the two ISPs. Ties in this set can be broken using the proposing ISP's own preferences. Another method is to propose the best option for the proposer that has minimal negative impact on the other ISP.

- *Accept option?* The other ISP decides whether to accept the selected option. This function gives ISPs veto power over the selected options, which they might use if the preference for this option has changed since last advertised or if they perceive that the proposer is not playing by the mutually agreed rules. When an option is accepted, the preference lists are updated to reflect that the flow has been tentatively pinned.

- *Reassign preferences?* Reassignment occurs when either of the two ISPs wants to update its preference list. This is needed when the preference classes are based on constraints such as available bandwidth that may change after one or more flows have been pinned to a route.

- *Stop?* ISPs decide whether they want to continue negotiating over more flows. ISPs can choose to stop when they perceive that there would be no additional gain in negotiating more flows. We call this the *early* termination point. Alternatively, an ISP may continue as long as their cumulative gain is positive, even though it may be lower than what might have been with early termination. We call this the *full* termination point, and it might be preferred in the interest of overall welfare and the expectation of reciprocal altruism which would lead to more self-gain over time. The socially best outcome is for the ISPs to continue until all the flows have been negotiated, even if that means a reduction in one ISP's overall gain. This is unlikely to be the common case.

- *Accept solution?* After the negotiation phase is over, the ISPs decide if they are satisfied with the outcome. If both of them are satisfied, they implement the outcome in their respective networks. If one of them is not satisfied, for instance if it perceives that the outcome is unfavorable or unfair, the solution is not implemented, at which point the ISPs can either renegotiate or decide to not cooperate at all.

This framework defines a family of possible protocols. For our experiments, we selected the following specific options, which are geared to explore the potential benefit of negotiation. We used a preference class range of [-10,10]. ISPs took turn to select one option, and did so by picking the option that maximized the gain across the sum of the two ISPs, breaking ties using its own preferences. The selected options were always accepted by the other ISP. Once assigned, preferences were not reassigned for the latency experiments; Section 5.2.3 explains how preferences were reassigned for the bandwidth experiments. We simulated both early and full termination points. Both ISPs always accepted the computed solutions.

4.1 An Example

We illustrate the working of NP using the second example (Figure 2) presented in Section 2. Since the two flows impacted by the failure are $f2$ and $f3$, we simulate the two ISPs negotiating for those two only. In doing this we are assuming that the ISPs prefer stability and do not want to disturb the flows not directly impacted by the failure. Each flow has two options – the top and bottom peering links. For simplicity, we assume that the preference class range is [-1, 1].

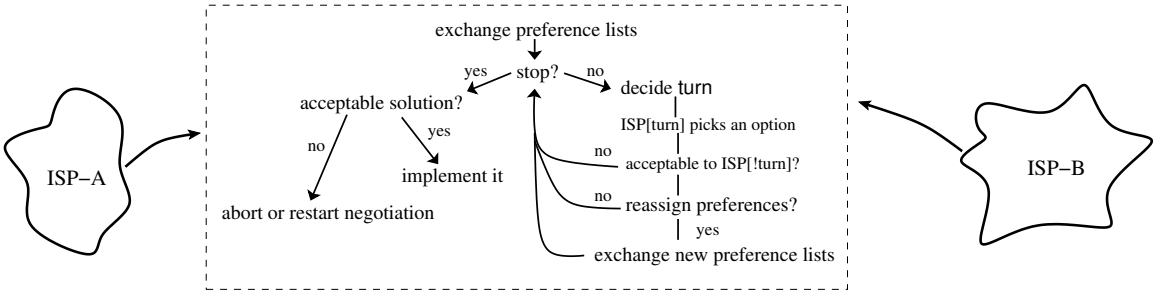

Figure 4: The negotiation framework. Both the ISPs have a say in all of the functions and decision points in the framework.

Initial preference lists

Class	**ISP-A**	**ISP-B**
1		
0	$\mathbf{f2_b}$ $f3_b$, $f3_t$	$\mathbf{f2_b}$, $f2_t$, $f3_b$, $f3_t$
-1	$f2_t$	

Reassignment after pinning $f2_b$

Class	**ISP-A**	**ISP-B**
1		$\mathbf{f3_t}$
0	$f3_b$, $\mathbf{f3_t}$	$f3_b$
-1		

Figure 5: Preference lists for the example in Figure 2. The subscript represents the peering link of the flow - b implies bottom, and t implies top. The option selected at each step is shown in bold.

In Figure 5, the top table shows the initial preferences lists for the two ISPs. These are relative to the default of both flows traversing the bottom link. The subscripts for the flows denote the peering point. Recall that, in that example, ISP-A is averse to $f2$ traversing the top peering link, and ISP-B is averse to both flows coming in via the bottom peering link.

Initially, all the options for ISP-A are as good as the default except $f2$ going over the top link. ISP-B is indifferent to all the options because preference classes to flows are assigned independently of each other. So ISP-B can handle either of the two flows entering via the bottom link (the problem arises only when they both do). Suppose ISP-A gets the first turn, and picks an option that maximizes the total gain. This is any of the three options in class 0 for ISP-A. Assume that $f2_b$ is selected, and is accepted by ISP-B. At this point, the ISPs reassign their preferences. The new lists are shown in the bottom table. Now ISP-B prefers $f3_t$ over the default. ISP-B get the next turn, and selects $f3_t$. This option in accepted by ISP-A, leading to the final solution shown in Figure 2e.

A different solution is possible in the toy example above if ISP-A picks $f3_b$ the first time. At this point, we have a situation

Class	**ISP-0**	**ISP-1**
2	$f1_a$	$f2_a$
1	$f1_b$, $f2_b$	$f1_b$, $f2_b$
0		
-1		
-2	$f2_a$	$f1_a$

Figure 6: A preference list example with two flows, each with two options. In fi_x, i is the flow index, and x is the option index.

in which ISP-A does not want the flow $f2$ to use the top link, and ISP-B does not want it use the bottom link. Whichever way this flow is routed, one ISP would be unsatisfied (it can either accept this solution, or call for a renegotiation). Thus, because of its hill climbing nature, NP may not always arrive at the most desirable solution when the flows are not independent. Yet we show in Section 5 that it approximates the optimal well in practice.

4.2 Discussion

The key observation behind the design of NP is that there are paths in the system that are much better than the default for one ISP but only slightly worse for the other. The cumulative impact of using such paths is that both ISPs benefit significantly by trading small losses for big gains. For example, in Figure 1 moving the flow $A \rightarrow B$ to the middle peering link presents slight additional cost for ISP-A but leads to a significant gain for ISP-B, and the movement of both flows leads to significant gains for both ISPs. Thus, by sharing preferences that reflect the additional cost, ISPs can find such paths without revealing detailed information about their network. We show in Section 5 that trading across flows is central to computing good solutions in practice.

The above observation motivates the need to select options that maximize the sum across utilities, rather than maximizing individual gain. For example, consider Figure 6, a case where two

flows with two options each are up for negotiation. Individual maximization picks $f1_a$ in the first round, and $f2_a$ in the second round, leading to a net gain of 0 for both ISPs. With the preference sum method, ISP-0 picks $f1_b$ and ISP-1 picks $f2_b$, leading to a net gain of 2 for both ISPs.

In Section 3, we argued that ISPs should be able to compute solutions tailored to their situation. NP can be used to find a wide variety of solutions as long as the preference classes reflect the optimization metrics being used by the ISPs. Optimal solutions are approximated when the ISPs' metrics are compatible, ISPs select gain maximizing options and continue negotiating until all flows have been negotiated (which might mean a loss for one of the ISP). If the ISP with lesser cumulative gain gets to select the option, giving it a chance to catch up with the other ISP, NP will compute fair solutions. If the ISPs select gain maximizing solutions, NP can approximate Pareto-optimal solutions. This is because if NP's solution were not Pareto-optimal, there must exist another solution that is better for both ISPs. Thus, this hypothetical solution must have a strictly higher sum of gains than NP's solution, which is not possible if NP was picking gain maximizing options.

A concern when competing entities negotiate is that one may try to manipulate the solution in their favor. On this issue we differ philosophically from traditional game-theoretic solutions. We design our negotiation protocol so that good solutions are computed when ISPs cooperate. Mechanism design, on the other hand, aims to neutralize cheating via "strategy-proof" solutions in which truth-revelation is provably the best course of action. This is powerful when possible but typically difficult to achieve and usually requires that the participants not know anything about each other's internal information. This is not the case in our setting. Information is hard to hide completely: there are out-of-game channels that exist in the real world (when ISPs learn about each other's networks) and negotiation among ISPs is a repeated game in which exchanges in one round inform the next. Further, as well as being very difficult to achieve, strategy-proof solutions do not appear to be necessary. Most ISPs cooperate today using back-channels that are not strategy-proof. Rather, there is value to reputation and a cost to cheating. If an ISP persistently tries to manipulate its peers, it is likely to be caught sooner or later based on the history of past declarations and outcomes. Because ISPs can collectively punish a miscreant (e.g., by not responding to problems or disconnecting it from the global Internet), these factors deter ISPs from egregiously selfish actions today, and we expect they would help to deter manipulative actions in our context.

We make two more observations on the issue of cheating in inter-ISP negotiation. First, a cheating ISP can never cause the other ISP to lose, only gain less, because the option to walk out of the negotiation is always there. Second, various functions within NP can be fixed (at the cost of flexibility) to make cheating harder. For instance, assume that the option selection criterion is fixed. Since both ISPs make their selection based on the same information, an incorrect preference list might end up hurting the cheater. An interesting avenue for future work is to assess and reduce the ability to cheat in NP while retaining most of its flexibility.

4.3 Deployment

We have described the negotiation protocol independent of the exact mechanism used to implement it. However, for the protocol to be deployed in the Internet, it should be possible to integrate it with the current routing infrastructure. While this has not been the focus of our work to date, we can see two ways to achieve this integration. The first is to integrate it in-band with BGP, for instance, through the use of communities. A second, and likely more attractive option, is out-of-band integration. Here, the negotiation process uses current data to decide the path in the network a particular flow should take. Once the path has been decided, ISPs use low-level BGP mechanisms such as MEDs, localprefs and communities to implement it. This architecture in similar in spirit to OPCA [2]. It has several advantages in our context. First, it keeps the negotiation framework agnostic of the implementation mechanism and avoids overloading an already fragile BGP. Second, out-of-band negotiation implies that there is less of a chance that the information exchanged between the ISPs would be leaked globally, as can happen today with MEDs and communities [16]. Third, as we show in Section 5, the benefits the negotiation are best realized when done across flows, since individually optimizing each flow does not lead to much gain for either ISP. Such a global view of negotiation is much more cleanly accomplished with an out-of-band architecture than by embedding negotiation information in individual route advertisements.

In a competitive situation, ISPs will probably want to verify that the negotiated settlement was actually implemented. This should be straightforward to accomplish: the ISPs can probabilistically verify that the flows coming over a peering link are consistent with what was negotiated.

5 Experiments

In this section, we report the results of simulation experiments we designed to evaluate the potential benefits of negotiation. Our goal is to explore and answer two high-level questions:

1. *How much better is the social optimal than selfish routing by both ISPs?* That is, if we consider two ISPs to be a single larger system in which information is completely shared, then how much better can the routing be? Is it better for both ISPs, or better for one but worse for the other? This result places an upper bound on what can be achieved

by negotiation. We will show that, while the societal optimal is often better for one ISP but worse for the other, there do exist Pareto-optimal solutions that benefit both ISPs.

2. *How much of the potential gain of the social optimal can be realized in practice, assuming that the ISPs cooperate to their mutual benefit?* That is, given the restrictions we have placed on NP (e.g., limited information sharing) how closely can its outcomes track the benefits of the social optimal? We will show that nearly all of the potential gain can be realized the majority of the time.

The answers to these questions depend on many aspects of ISP operation. Our approach is to use measured data to model ISPs where it is available, e.g., ISP topologies and geographies, and to postulate a range of alternative models drawn from the literature where it is not, e.g., the internal ISP optimization metric. In this way, we hope to focus on realistic rather than theoretical best- or worst-case settings, while avoiding results that are sensitive to incidental choices in our setup.

We divide our experiments below into two major classes according to the choice of optimization metric. The first, based on a latency metric, explores the steady-state reductions in overall network resource usage that can be achieved, implicitly assuming that the network capacity is well-matched to the traffic it carries. The second, based on a bandwidth metric, explores how negotiation can reduce the impact of "hotspots" that occur in the short-term when the traffic is no longer well-matched to the network, e.g., due to a failure. Since we are interested in evaluating the potential of negotiation by comparing it to the optimal, we restrict ourselves to experiments where both ISPs use the same metric so that the optimal is well-defined.

5.1 Latency and Cost

Improved routing in the steady-state compared to "early exit" is valuable for two reasons. It reduces the overall network resource consumption, allowing a smaller network to support a given set of external traffic demands, and it can also provide end users with higher performance paths.

5.1.1 Methodology

To assess improvements in steady-state routing, we define a metric that reflects the total resource consumption in the network. Specifically, we use the sum of the cost of all upstream-PoP to downstream-PoP flows, where the cost of a flow is the sum of the latencies of the links along its path. That is, higher latency paths send traffic through a greater portion of the network and have a correspondingly higher operating cost for the ISP.

To calculate this metric, we require ISP topology and latency information. We use the dataset released by Spring *et al.* [23].

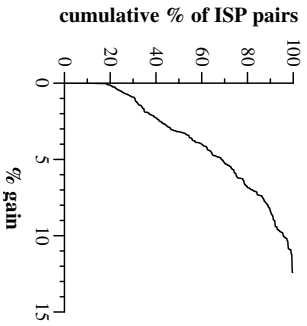

Figure 7: The societal benefit of optimal routing. The x-axis is the percentage reduction in the total cost of routing across both ISPs when moving from the default to optimal routing.

This dataset includes the PoP-level topologies of 65 ISPs, along with geographic coordinates and an estimate of inter-PoP link weights. It is diverse in terms of ISP size and geography and is the largest dataset of its kind that we were able to obtain. We estimate the latency of links with each ISP using the geographical distance between its endpoints; earlier work shows that this is a reasonable approximation [19]. We use the link weights to compute paths internal to an ISP, i.e., the paths between the traffic sources or sinks that are selected as part of our experiments.

In the experiments below, we consider pairs of ISPs, restricting ourselves to those that peer at two or more locations and so allow a choice of peering points. We also exclude eight ISPs that use circuit technologies such as MPLS in their network. In all, we carried out these experiments for 225 ISP pairs. Each pair has many unidirectional flows going in both directions that have a PoP-level source in one ISP and a PoP-level sink in the other ISP. Peering point selection depends on the experiment, and is described below.

5.1.2 Social Optimum

We first quantify the gain that would be achieved by socially optimal routing compared to the default routing, where each ISP acts independently. Default routing uses the "early-exit" policy; the peering point chosen by the upstream ISP for that flow is the one that is closest to the source PoP. Spring *et al.* found that most traffic between ISPs is carried using early-exit routing [23]. The socially optimal routing is computed by choosing the peering point that minimizes the total latency for each flow across the ISPs. This will minimize the overall metric that we have defined since the costs of individual flows do not interfere with one another.

Figure 7 shows the results of this experiment. It plots the cumulative distribution function (CDF) of the gain of optimal routing

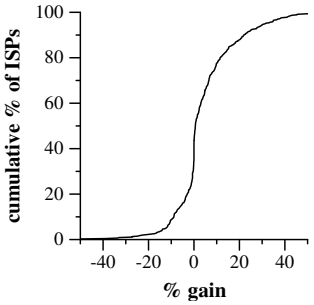

Figure 8: The individual benefit of optimal routing. The x-axis is the percentage reduction in the cost of routing for an ISP when moving from the default to optimal routing.

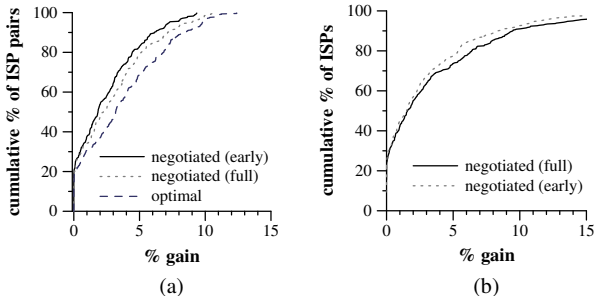

Figure 9: (a) The potential social benefit of negotiation. The x-axis is the percentage reduction in the total cost of routing across both ISPs relative to the default routing. (b) The potential individual benefit of negotiation. The x-axis is the percentage reduction in the cost of routing for an ISP relative to the default routing.

relative to the default summed across the two ISPs.[1] Each point on the graph corresponds to the latency for all flows between the PoPs of a specific pair of ISPs. We see that the gain from optimal routing is more than 4% for half of the ISP pairs, and more than 10% for one in ten ISP pairs. This aggregate gain is not large, and it suggests that pairs of ISPs already route well in an overall sense by using "early exit." That is, the "price of anarchy" is low in practice for pairs of ISPs, well below its theoretical bound [14]. The main value of negotiation in this setting is likely to improve in an automated way the performance of the small number of flows that suffer significantly under default routing; we consider flow-level gains shortly. However, even a small aggregate gain may be worthwhile. It is possible that a 4% decrease in overall path length translates to a corresponding reduction in the number of routers, circuits, and facilities in the network. The economic savings of even 4% of the total cost of the network infrastructure could still be significant. We plan to explore this issue in future work.

We now investigate the distribution of the gain from socially optimal routing. In Section 3, we argued that the social optimum may not be the desired operating point (even if it can be computed) because it may cause one ISP to lose compared to default, independent routing. Figure 8 shows that this is indeed the case. It plots the gain in cost calculated separately for the two ISPs instead of across the pair. Roughly a third of the ISPs actually lose by opting for the social optimum, with some losing by more than 30%. These ISPs will have little incentive to move to the optimal solution.

5.1.3 Negotiated Solution

We now show that negotiation can provide gains that are comparable to that of the social optimum without penalizing either ISP. We do this by using NP to negotiate peering points over the same set of ISP pairs and traffic flows. The metric used by each ISP is based on latency: the peering point options are assigned to preference classes in the integral range [-10, 10] based on their internal latency compared to the default. The ISPs alternate to select an option. Since we are exploring the maximum potential gain from negotiation, we simulate ISPs picking an option that maximizes the sum of the preference classes. Selected options are always accepted by the other ISP. Flows not accepted before the termination point of the negotiation are default routed. We show results from two termination strategies below.

Figure 9a shows the results of this experiment. It plots the gain of negotiated routing, for both early and full termination conditions, relative to the default routing. The previous curve for optimal routing from Figure 7 is included for comparison. We see that the negotiated routing is not only better than the default routing, but it is also very close to optimal routing. Surprisingly, early termination does almost as well as full termination. This means that there is not much "social cost" (or gain) for ISPs to terminate the negotiation when they expect no further gain for themselves. That is, most of the benefit of negotiation comes early on in the negotiation process.

Figure 9b breaks down the gain for individual ISPs, rather than across the pair. Individual ISPs in the pair do not lose when using negotiated routing, which follows from the definition of our protocol. The top 10% of the individual ISPs experience an efficiency gain of more than 10%.

Next, we observe that the gains for both ISPs depend on negotiation across a large set of flows. A simpler alternative strategy would be to consider bi-directional flows individually and discard obviously bad options. We experimented with two strategies – *flow-Pareto* and *flow-both-better*. The former rejects peering point pairs that are worse for both ISPs compared to picking early-exit peering points on both sides, while the latter rejects those that are worse for any one ISP. For example, in

[1] For visual clarity, a maximum of two outliers have been removed from some of the graphs in this paper.

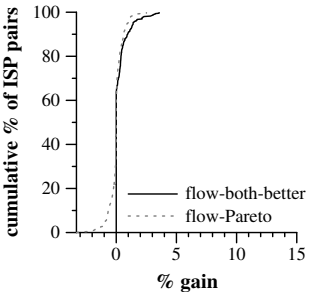

Figure 10: The social benefit of two alternate routing strategies that simply discard bad options. The x-axis is the percentage reduction in the total cost across both ISPs relative to the default routing. Neither achieves nearly the potential benefit of negotiated or optimal routing.

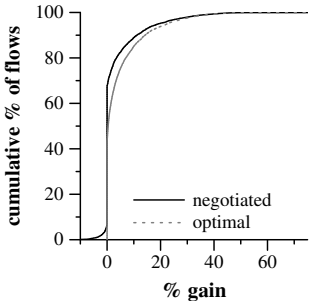

Figure 11: A flow-level view of optimal and negotiated routing. The x-axis is the percentage reduction in the cost of the flow. This graph aggregates all flows across all ISP pairs.

Figure 1, using the top link for $A \rightarrow B$ and the middle link for $B \rightarrow A$ is flow-Pareto, and using the middle peering for both directions is flow-both-better. If multiple peering point pairs satisfy the required criterion, one is picked at random. Figure 10 plots the gain results for these strategies. It shows that these seemingly reasonable strategies which avoid obvious wastage at flow-level are not effective at reducing the cost of routing. In fact, their cost is close to that of the default itself. This implies that, for mutual gain to be realized, negotiation must be done across flows and ISPs must be willing to trade minor losses on some flows for significant gains on other flows.

We close this section with a flow-level view of negotiation. Figure 11 shows the CDF of gains experienced by individual flows when moving to optimal and negotiated routing. We make two interesting observations. First, the performance of negotiated routing is largely similar to the optimal even at the flow-level, but a small fraction of flows suffer by a small amount. Second, individual flows gain much more than the ISP-pair-level aggregates shown in Figure 9a. Over 7% of the flows gain by more than 20%, and 1% of the flows gain by over 50%. We speculate that it is the flows that suffer heavily due to default routing are the ones that are manually optimized by operators today (Spring *et al.*observed that a small fraction of flows were non-default routed among many ISP-pairs [23]). The graph shows that automated negotiation can improve the performance of these flows just as well, thus saving precious operator time.

5.2 Bandwidth and Congestion

We now evaluate the benefits of negotiation in a different setting where the ISPs are interested in controlling overload or hotspots. Even when ISP networks are well-engineered, overloaded links can be a concern during failures and sudden changes in traffic demands, as might be caused by a flash crowd [5].

5.2.1 Methodology

We consider scenarios where a peering link fails, and simulate ISP negotiation only for flows that are impacted by the failure; in the interest of stability ISPs are likely to restrict their negotiations only to such flows. We do this experiment only for those ISP pairs that have three or more peering links, because for negotiation to apply there must be at least two working peering links after the failure. There are 247 such ISP pairs in our dataset. Our results may also apply to internal link failures and changes in traffic matrices.

We use the same measured ISP topologies, geographies and link weights as before. However, overload is more difficult to evaluate than latency and cost for two reasons. First, calculating bandwidth measures requires estimates of ISP link utilizations and traffic matrices, neither of which is readily available. Second, the choice of metric to represent overall ISP cost in terms of individual, congested links is less clear.

To handle the first issue, we postulate a range of workload and utilization models and simulate with each of them. Since we are dealing with traffic that flows between two ISPs, our traffic matrix is an estimate of the amount of traffic from upstream-ISP PoPs to downstream-ISP PoPs; we consider only one direction of traffic at a time. We use the gravity model to derive this matrix [17, 26]. This model predicts that the amount of traffic between a pair of PoPs is proportional to the product of the weight of the PoPs. Thus, we reduce the problem of generating a traffic matrix to that of assigning weights to PoPs.

The results presented below use a model in which the weight of a PoP is proportional to the population of its city. We use data from CIESIN [1] to estimate the population of a city as the number of people in a 50×50 square mile grid centered on the geographical coordinates of the city. The motivation for this model is that it leads to a skewed traffic matrix in which larger cities consume more bandwidth, both hallmarks of real Internet traffic [15, 4]. We experimented with two other weight

assignment models: i) *constant*: all PoPs in an ISP have the same weight; and ii) *uniform*: the weights are derived from a uniform distribution. We obtained similar results for all three models, but omit results for the last two due to space constraints.

We still need a model for link capacity to calculate the ISP metrics below. Since we model only the traffic going between the two ISPs, we interpret link capacities as the capability of the link to carry traffic of that class. The traffic matrix model combined with default routing within an ISP lets us compute the load on each link. We then make the assumption that the load on a link before a failure is proportional to its capacity [26]. That is, in steady-state a well-designed network tends to be matched to its traffic to some extent so that links that carry more traffic tend to be of higher capacity.

A complication with this method is that it does not produce capacity information for links that exist in the ISP topology but were not carrying any traffic before the failure; these apparently unused links exist because we model only the traffic between the two ISPs. The issue is that we can neither remove these links, since they may be used after failures, nor assign them minimal capacity, since they may then cause spurious overloads. Instead, the results presented in this paper are derived by assuming that the pre-failure load on such links is the median of the links with non-zero load. The intuition here is that the unused links are in a way backup links for this class of traffic, and their capability for carrying this traffic varies between the minimum and maximum load among the links that are used. We pick the median as the representative metric in that range. We found that other choices (namely the maximum load and average load instead of median load) produce similar results. As a similar precaution, to preclude the possibility of our results being dominated by links that carry little traffic to begin with, we experimented with "upgrading" all links below the median load to the median load. Our results were insensitive to these choices; we present the results using the median load assignment and median upgrading.

To handle the second issue, the choice of ISP optimization metric to control overload, we use two different models. Intuitively, ISPs prefer routing that does not significantly increase the load on links after a failure. Note that load here is relative to the link capacity, since a 100 Mbps traffic increase on a 1 Gbps link does not have the same impact as a 100 Mbps traffic increase on a 100 Mbps link. All ISPs overprovision to some extent, so the link capacity of well-engineered networks is likely to be some small multiple of its average load. A much higher offered load after a failure implies that either the link becomes congested or it was significantly over-provisioned to begin with, which is expensive. Thus our metric should penalize large increases in link load after a failure relative to the link load before the failure.

The first ISP metric we use minimizes the maximum multiplicative increase in the load across any link in the topology. The second is based on Fortz and Thorup's linear programming formulation of optimal routing [9]. It minimizes the sum of link costs,

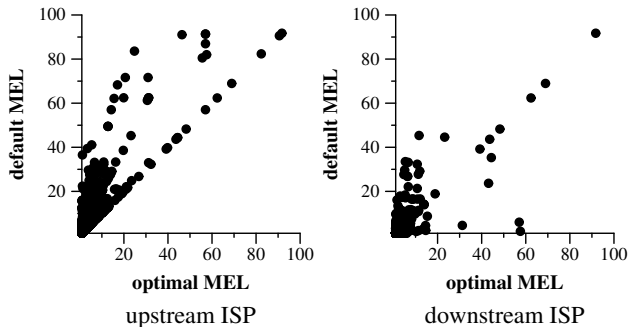

Figure 12: Comparison of the default and optimal routing after a failure. The x and y values are the MELs for the two cases.

where the cost of a link is a piecewise linear, convex function of multiplicative excess load with increasing slope. The difference between the two metrics is that while the former minimizes the maximum, the latter tries to optimize the whole network [9]. We found that our results were insensitive to the choice of the metric; we present only those obtained with the former metric because it is more intuitive and easier to compare across topologies.

5.2.2 Social Optimum

We first quantify the potential for optimizing routing across both ISPs without regard to organizational boundaries. Initially, we compare this to the case where the ISPs use "early-exit" over the topology without the failed peering link. To compute the social optimum, given the topology and optimization metric described above, we use a linear program [18] where the constraints encode the traffic matrix and flow conservation properties. For computational tractability, we allow flows to be fractionally divided among peering links. (Otherwise we must solve a computationally harder integer linear program.) Our results are thus an upper bound on the social optimum without fractional routing. Finally, we report the quality of routing using maximum excess load or *MEL*. This is the maximum ratio of load after and before the failure on any link in the topology.

The results of this experiment are shown in Figure 12, which compares the MEL of default routing with that of optimial routing after a peering link failure. Each data point corresponds to one hypothesized peering link failure. So there are four distinct points for ISP pairs with four peering links between them. We see that in many instances, the MEL for the default case is significantly larger than the optimal case. This is true even in cases where the optimal MEL is high, suggesting that overloading due to default routing is not limited to thin links in the topology. Data points where the optimal MEL is more than the default MEL represent cases where global optimization hurts an individual ISP.

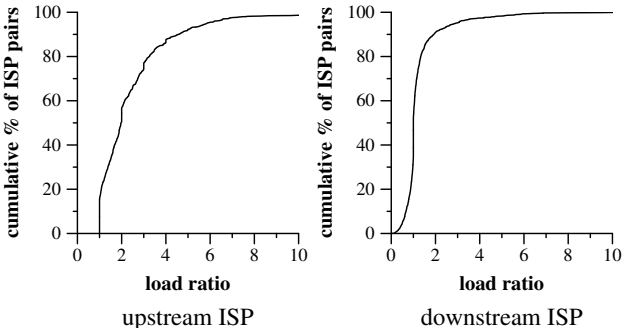

Figure 13: Comparison of the default and optimal routing after a failure. The x-axis is the ratio of the MELs for the default and optimal routing.

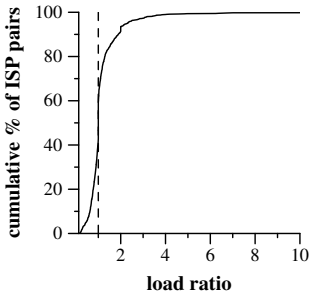

Figure 14: The impact on downstream ISP of unilateral routing optimization by the upstream ISP. The x-axis is the ratio of the MELs for the upstream-optimized and default routing; values more than one imply that upstream-centric optimization was harmful for the downstream ISP.

Figure 13 shows a different view of the same data as Figure 12. It plots the MEL with default routing normalized by the MEL with optimal routing. There is a significant difference between the default and optimal routing. For the upstream ISP, the ratio of the two MELs is more than two for half of the cases, and more than five for 10% of the cases. This implies that the default routing tends to overload certain links in the topology even when this overloading is avoidable.

Both Figures 12 and 13 show that the overload in the upstream ISP is more than that in the downstream ISP. Early-exit routing implies that an upstream source picks exactly one peering point for all the destinations. When a peering link fails, all sources that were using it migrate to another peering point. In this process a number of sources can start traversing an internal link in the upstream ISP that they were not using before the failure, potentially leading to congestion. Contrast this with what happens in the downstream ISP. When a peering link fails, the excess load on the remaining peering links is bounded by the traffic carried by the failed link. Since all peering links send traffic to all destinations even before the failure, no new paths are explored, which bounds the excess load that an internal link in the downstream ISP will have to carry.

We next consider a different ISP routing scheme. Since the upstream ISPs suffers more due to peering link failures, a natural question is what happens if instead of negotiating with the downstream, the upstream unilaterally adjusts the traffic flow to suit itself. It is possible that these actions, which are an attempt to load balance traffic in its own network, do not hurt or may even end up benefiting the downstream ISP. We evaluate this hypothesis by simulating the upstream ISP optimizing the routing for its own network.

Figure 14 shows that the impact of upstream-centric optimization on the downstream ISP. It shows the ratio of MELs in the downstream ISP with upstream-centric optimization versus early-exit routing. We see that the result is unpredictable. While in some cases, upstream-centric optimization helps the downstream (left end of the graph), in others the downstream ISP heavily suffers (right end of the graph). In 10% of the cases, the MEL for upstream-centric optimization is more than twice of that for the default routing. Thus, the upstream unilaterally making routing changes is undesirable because it may end up causing congestion in the downstream. This is similar to the example in Section 1.

5.2.3 Negotiated Solution

So far we have shown that the default routing is highly suboptimal compared to the optimal and that unilateral rerouting by the upstream is undesirable. The next question is whether negotiation can help in this situation. To answer it, we simulated the two ISPs negotiating using NP. Flow options are put in preference classes in the integral range [-10, 10] based on the maximum excess load (MEL) they cause on the links they traverse. Unlike the case with latency, bandwidth-based preferences interfere with one another because the routing of traffic along some path reduces its ability to accommodate further traffic. To handle this, preferences are reassigned after flows representing 1% of the traffic over the failed peering link are rerouted.

Figures 15 and 16 show the results of this experiment. They plot the MELs of the default and negotiated routing normalized by the MEL of globally optimal routing. Figure 15 shows the experiment in which all previously unused links were assigned the median load, and Figure 16 shows the one in which all links below the median load were upgraded to the median load. Both sets of results are very similar, implying that they are not dominated by links that were carrying no or too little traffic in the pre-failure scenario. We show results only for the early termination case; the full termination results were almost identical.

The graphs show that for almost all the ISP-pairs, negotiated routing is very close to the globally optimal routing even

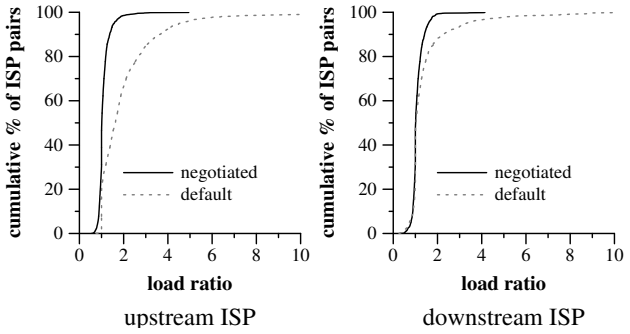

Figure 15: The quality of negotiated routing with median load assignment for previously unused links. The x-axis is the MEL relative to the MEL of optimal routing.

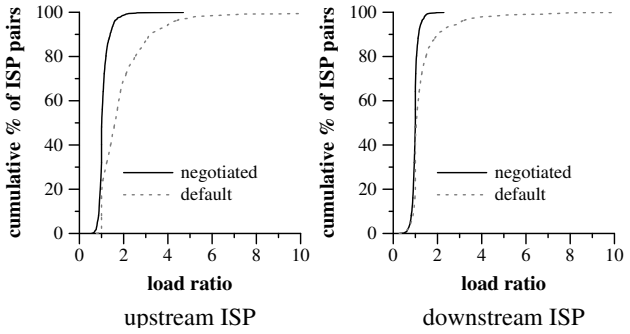

Figure 16: The quality of negotiated routing with median upgrading for all links with load below the median. The x-axis is the MEL relative to the MEL of optimal routing.

though: the amount of information used to compute it is much less; the procedure to compute it is much simpler; and the routing itself is restrictive (while the optimized routing can fractionally divide a flow among peering links, negotiated routing chooses exactly one peering link per flow).

6 Related Work

Compared to intradomain traffic management, interdomain traffic management has received relatively little attention in the research community. Two exceptions, however, are the works of Feamster *et al.* [7] and Winick *et al.* [25].

Feamster *et al.* recommend procedures an ISP can follow to predictably control interdomain traffic (outbound only). Predictability is a key concern because a local configuration change can either adversely impact a neighboring ISP or trigger a change in the way a neighboring ISP routes traffic. Based on common BGP configuration idioms, they outline a set of "safe" changes that are less likely to influence the way the neighboring ISPs route traffic. Our approach is fundamentally different. We argue that interdomain traffic management is an inherently cooperative task. Instead of trying to guess each other's routing policies, ISPs should negotiate to obtain mutually acceptable solutions. Negotiation ensures that the neighboring ISPs are not taken by surprise, as might be the case with unilateral configuration changes.

Winick *et al.* propose a simple method to manage interdomain traffic in consultation with neighboring ISPs. Before making a configuration change, an ISP informs the other ISPs of the impact of that change. For instance, an ISP would tell its neighbors that it is moving a certain set of flows from peering point A to B. The neighbor decides if that change is acceptable. Hence, their proposal is in fact a form of negotiation. The difference is that NP uses preference lists to compute a solution acceptable to both ISPs. Preference lists are better suited to the problem at hand because the solution space is very big – exponential in the number of flows times the number of peering links. Without any input from the second ISP as to its preferences, it is computationally hard for the first ISP to explore this space and propose solutions that are acceptable to both ISPs. Moreover, negotiation should allow both ISPs to influence the final solution. If the set of acceptable solutions is large, then Winick *et al.*'s approach is heavily biased in favor of the proposing ISP.

Our work is another piece in the research theme that examines the "price of anarchy" in the Internet. Roughgarden and Tardos have analyzed the cost of selfish routing in a setting where end users completely control the paths that their packets use [21]. Our setting is different; we analyze the cost of selfish decisions by ISPs. Johari and Tsitsiklis found that early-exit routing can be three times worse than optimal routing [14]. Our results over real ISP topologies show that this cost is much lower in practice. Closely related to our work is that of Qiu *et al.* [20]. They empirically evaluated the cost of selfish routing over measured ISP topologies. Like Roughgarden and Tardos, they do so in a setting where end users control their paths. A distinguishing feature of our work is that we also explore a point between the extremes of absolutely optimal and absolutely selfish. We study how close to optimal one can get through practical solutions acceptable to both parties.

Yet another body of work has examined the impact of current interdomain routing policies on end user performance. The Detour project was perhaps the first to quantify the inefficiencies of Internet routing [22]. Since then many researchers have measured the impact of interdomain routing policies in different ways [24, 12, 23]. We have looked closely at the impact of early-exit routing in particular and unilateral policies in general, with a view to exploring constructive solutions in this domain. An interesting avenue for future work would be to reinvestigate the need for overlays such as RON [3] and Detour in a world where ISPs fully cooperate with each other to remove routing inefficiencies.

7 Conclusions

In this paper, we have explored an alternative approach to interdomain routing. In our approach, ISPs explicitly negotiate the paths for the traffic they exchange. Using simulation with over sixty measured ISP topologies, we explored the benefits of negotiation in the context of two neighboring ISPs. We found that compared to default routing in which the ISPs make independent decisions, the two ISPs can find win-win solutions for both latency (resource usage) and bandwidth (avoiding hotspots), i.e., routing configurations beneficial for both of them. The benefits for latency are small on average, with half the ISP pairs gaining only 4% over the default, but can be significant for a small fraction of flows whose default paths are very circuitous. For bandwidth, negotiation can significantly reduce the situations in which a failure causes an overload. More generally, we found that the negotiated solutions have the potential to be very close to the optimal in terms of quality. This is interesting because the optimal solution itself turns out to be an undesirable operating point because one ISP loses roughly a third of the time. We also observe that benefits can only be obtained across a large set of flows routed by the ISP, rather than for a single bi-directional flow.

We also presented NP, a practical negotiation protocol designed for inter-ISP negotiation. NP is flexible enough to compute a wide variety of solutions, supports negotiations even when the participants have incompatible optimization criteria, and requires ISPs to share only limited information – opaque preference classes rather than transparent metrics. We found that ISPs can cooperate to use NP to find solutions that are almost as good as the case when the two ISPs share complete, detailed information. A key benefit is that this negotiation is automatic – it has the potential to relieve operators from the time-consuming and error-prone task of reacting to one class of short-term performance problems.

Our work is a first step towards a broader vision in which negotiation happens not only between neighboring ISPs but among all the ISPs traversed by a flow. For the stability and performance of interdomain routing, it is important that ISPs have a global perspective while making local decisions. NP-like negotiation provides ISPs with that view. By working together, ISPs can achieve better efficiency for themselves and better performance for their users.

References

[1] Center for International Earth and Science Information Network. http://www.ciesin.columbia.edu.

[2] S. Agarwal, C.-N. Chuah, and R. Katz. OPCA: Robust interdomain policy routing and traffic control. In *IEEE Openarch*, Apr. 2003.

[3] D. Anderson, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. In *SOSP*, Oct. 2002.

[4] S. Bhattacharyya, C. Diot, J. Jetcheva, and N. Taft. Pop-level and access-link-level traffic dynamics in a tier-1 POP. In *ACM SIGCOMM Internet Measurement Workshop*, Nov. 2001.

[5] C.-N. Chuah. A tier-1 ISP perspective: Design principles & observations of routing behavior. http://sahara.cs.berkeley.edu/jun2002-retreat/chuah_talk.pdf, 2002 June.

[6] D. Clark, J. Wroclawski, K. Sollins, and R. Braden. Tussle in cyberspace: Defining tomorrow's Internet. In *ACM SIGCOMM*, Aug. 2002.

[7] N. Feamster, J. Borkenhagen, and J. Rexford. Guidelines for interdomain traffic engineering. *ACM Computer Communication Review*, 33(5), Oct. 2003.

[8] J. Feigenbaum, C. Papadimitriou, R. Sami, and S. Shenker. A BGP-based mechanism for lowest-cost routing. In *ACM Principles of Distributed Computing*, July 2002.

[9] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. In *IEEE INFOCOM*, Apr. 2000.

[10] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Aug. 1991.

[11] L. Gao. On inferring autonomous system relationships in the Internet. In *IEEE Global Internet Symposium*, Nov. 2000.

[12] L. Gao and F. Wang. The extent of AS path inflation by routing policies. In *IEEE Global Internet Symposium*, Nov. 2002.

[13] T. Griffin and G. T. Wilfong. An analysis of BGP convergence properties. In *ACM SIGCOMM*, Aug. 1999.

[14] R. Johari and J. N. Tsitsiklis. Routing and peering in a competitive Internet. Technical Report P-2570, MIT Laboratory for Information and Decision Systems, Jan. 2003.

[15] A. Lakhina, J. Byers, M. Crovella, and I. Matta. On the geographic location of Internet resources. *IEEE JSAC*, 2003.

[16] R. Mahajan, D. Wetherall, and T. Anderson. Understanding BGP misconfiguration. In *ACM SIGCOMM*, Aug. 2002.

[17] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: Existing techniques and new directions. In *ACM SIGCOMM*, Aug. 2002.

[18] K. Murty. *Linear Programing*. John Wiley & Sons, 1983.

[19] V. N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for Internet hosts. In *ACM SIGCOMM*, Aug. 2001.

[20] L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker. On selfish routing in Internet-like environments. In *ACM SIGCOMM*, Aug. 2003.

[21] T. Roughgarden and E. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2), Mar. 2002.

[22] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson. The end-to-end effects of Internet path selection. In *ACM SIGCOMM*, Aug. 1999.

[23] N. Spring, R. Mahajan, and T. Anderson. Quantifying the causes of path inflation. In *ACM SIGCOMM*, Aug. 2003.

[24] H. Tangmunarunkit, R. Govindan, and S. Shenker. Internet path inflation due to policy routing. In *SPIE ITCom*, Aug. 2001.

[25] J. Winick, S. Jamin, and J. Rexford. Traffic engineering between neighboring domains. http://www.research.att.com/~jrex/papers/interAS.pdf, July 2002.

[26] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast accurate computation of large-scale IP traffic matrices from link loads. In *ACM SIGMETRICS*, June 2003.