Cooperative Inter-Domain Traffic Engineering Using Nash Bargaining and Decomposition

Gireesh Shrimali† Aditya Akella⋆ Almir Mutapcic†
Stanford University†, University of Wisconsin-Madison⋆

***Abstract*— We present a new inter-domain traffic engineering protocol based on the concepts of *Nash bargaining* and *dual decomposition*. Under this scheme, ISPs use an iterative procedure to jointly optimize a social cost function, referred to as the Nash product. We show that the global optimization problem can be separated into sub-problems by introducing appropriate shadow prices on the inter-domain flows. These sub-problems can then be solved independently and in a decentralized manner by the individual ISPs. Our approach does not require the ISPs to share any sensitive internal information (such as network topology or link weights). More importantly, our approach is *provably Pareto-efficient and fair*. Therefore, we believe that our approach is highly amenable to adoption by ISPs when compared to past naive approaches.**

We conduct simulation studies of our approach over several real ISP topologies. Our evaluation shows that the approach converges quickly, offers equitable performance improvements to ISPs, is significantly better than unilateral approaches (e.g. hot potato routing) and offers the same performance as a centralized solution with full knowledge.

I. Introduction

A key component of operating and managing any ISP network is the ability to control how traffic enters or leaves the network. This is critical to ensuring that the ISP can offer good performance and reliability even in the face of internal or external failures and overload.

BGP provides networks with a limited set of mechanisms to achieve this control (e.g. local prefs for outbound control, MEDs and AS path pre-pending for inbound control). However, these mechanisms only offer ISPs *unilateral* control over traffic. Unfortunately, unilateral decisions of neighboring networks may have undesirable interactions, and may result in unstable routing [1], poor performance [2], and huge, unpredictable shifts in network traffic volumes [3].

Recently, it has been argued that supporting dynamic control over inter-domain traffic in a stable, efficient and predictable manner requires a new inter-domain traffic engineering architecture that is based on *explicit coordination* between ISPs [4], [5], [6]. In this approach, neighboring ISPs exchange information about inter-domain traffic volumes and routes, and participate in a simple "negotiation protocol" to arrive at mutually acceptable routes for the traffic between them (see Section VI for more details). It has been shown that such explicit coordination can simultaneously help *both* networks [5], [6].

These seminal studies establish the potential benefits of coordinated inter-domain traffic engineering (TE). Unfortunately, realizing co-operation among ISPs in practice is not straight-forward, since ISPs also *compete* against each other, and their competitive concerns must be explicitly accounted for. As such, any naive approach for inter-domain TE – such as the negotiation protocol above – is unlikely to be adopted by ISPs. In particular, we note that co-operative inter-domain TE approaches must satisfy the following criteria for ISPs to adopt them[1]:

1. Minimum information revealed: ISPs regard their network structure, link capacities, and link weights as "sensitive internal information", crucial to maintaining their competitive edge. Therefore, ISPs must be able to perform cooperative TE *without* having to reveal their sensitive information.

2. Efficiency: Cooperative approaches must ideally result in *Pareto-efficient* operating points. By this, we mean that the resulting allocation of traffic across inter-domain routes must lie on the boundary of the feasible outcomes – on this boundary, we cannot make one ISP better off without disadvantaging the other. Pareto-efficiency ensures that network resources are used in the most efficient manner by both ISPs. Note that efficient network usage is also the driving goal of intra-domain traffic engineering.

3. Fairness: Any inter-domain TE approach should yield an operating point that is *provably fair*. By fair we mean that cooperation should yield equitable performance gains to the participants when compared to their default TE strategies (e.g. both ISPs employing naive unilateral control). Approaches that yield disproportionate benefits are likely to be spurned by the ISP that gets the short end of the stick.

Another desirable property is that of *incentive compatibility*, which means that the participating ISPs have no incentive to lie or cheat. Without this guarantee, ISP's can "game" any inter-domain TE protocol to gain unfair advantage. It is a well-known fact that achieving fairness, efficiency and incentive compatibility *together* is impossible [7]. However, it is possible to achieve two out of these three criteria. As a first step, in this paper, we focus on fairness and efficiency. We assume that ISPs are willing to co-operate with each other, and will not resort to lying, as long as cooperation can yield better performance than the default un-cooperative mode of operation. We leave incentive compatibility for future work (further details in Section IV).

To the best of our knowledge, no single approach for inter-domain traffic engineering can provide information hiding along with fairness and efficiency (i.e. criteria 1–3 listed above). Existing approaches [5], [6] at best satisfy the first criteria, but not the other two. In this paper, we present a new cooperative inter-domain TE approach that can provably offer these three desirable properties. Therefore, we believe that our approach is highly amenable to adoption by ISPs.

Our approach uses ideas from multi-criteria optimization [8] and axiomatic bargaining [9]. Like past studies, we

[1]Unless otherwise specified, our focus in this paper is on a pair of neighboring ISPs.

assume that ISPs can improve their local performance by bargaining (or negotiating) about the traffic flow distribution on their peering links. Our first insight is that we can use the well-known concept of *Nash bargaining* [10],[11] to do so. Under this scheme, the ISPs agree to jointly optimize a social cost function, known as the Nash product, which is essentially the product of the utility functions of the two ISPs. The key advantage of using this approach is that the solution is guaranteed to provide Pareto efficiency and fairness. When the ISPs' utilities (measures of performance, such as average delay or maximum load on a link) are directly comparable, this solution is min-max fair, i.e., the gains from cooperation are equal. However, when the utilities are not comparable, it still provides a Pareto efficient solution that is *proportionally fair* [12]. By this, we mean that the gains from cooperation to individual ISPs are equal after some (automatic) suitable scaling of the utilities. This scaling is endogenous to the solution and, therefore, is highly desirable.

This leaves us with the issue of not revealing critical internal information. Our insight here is that we can use *dual decomposition* [13] to transform the joint optimization of the Nash product into a procedure with precisely this property, as follows. The global optimization problem can be decomposed into two independent sub-problems by recognizing the coupling flows (these are the flows crossing between ISP domains) and introducing appropriate Lagrange multipliers (or shadow prices) [13]. These sub-problems can now be assigned to the ISPs to be solved in a decentralized manner. These have the critical feature that they are completely local – an ISP's assigned sub-problem depends only on its own network – thus, the ISPs don't have to share critical internal information. Relying on this insight, we develop an iterative procedure based on the sub-gradient method [14] where, given Lagrange multipliers, the ISPs independently optimize their local sub-problems to come up with their required coupling flows. The Lagrange multipliers are then updated using the sub-gradient method, which uses the difference in the two sets of required flows to determine the magnitude of the update. The update can be done in a decentralized manner. After the update, the ISPs again try to optimize their local sub-problems. We show that this process converges in finite time to a fair and Pareto-efficient allocation.

We evaluate the effectiveness of our approach using simulations over real ISP topologies. Through simulations, we show that our approach significantly out-performs unilateral approaches such as the commonly used hot potato or shortest path routing (where ISPs route to nearest peering location in terms of link weights) as well as the Nash equilibrium setting (where ISPs optimize local objectives while playing best responses to each other [9]). For the case where ISPs employ similar utilities, we compare our solution against centralized optimal routing (where a central arbitrator optimizes the common objective across both ISP networks). We also confirm the proportional fairness guarantees of our solution via simulation experiments.

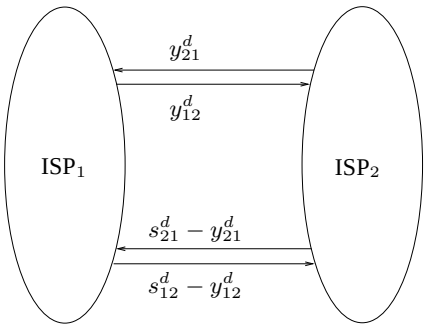

Fig. 1. The Model

II. Inter-domain Traffic Engineering using Nash Bargaining and Decomposition

A. The Model

We model the interaction between two ISPs: ISP$_1$ and ISP$_2$ as shown in Figure 1. These ISPs are optimizing utilities u_1 and u_2, respectively. As mentioned in Section I, these utilities are related to some measure of performance. These utilities could mean different things to the ISPs. For example, for one ISP the utility could be related to the average delay in the network, and for the other ISP the utility could be related to the maximum load on a link in the network. The ISPs optimize these utilities under the flow conservation constraints, i.e., flows from all sources to all destinations must be routed. To simplify exposition, in the following description, we assume that the ISPs employ MPLS-like routing. We believe the approach we describe below can be easily modified to yield a mechanism for setting link weights for ISPs using OSPF in a way similar to [15].

We make the common assumption that the utilities are either convex or concave functions, and that the ISPs are respectively minimizing or maximizing these utilities. For example, some convex utilities are the maximum load on a link and the average delay using convex link per unit delay functions (e.g., the per unit delay in an $M/M/1$ queue).

We assume that ISP$_1$ needs to send flows s^d_{12} to ISP$_2$ on a per destination basis: s^d_{12} is a vector of flows to be sent to each of the destinations in ISP$_2$ from ISP$_1$.[2] Similarly, ISP$_2$ needs to send flows s^d_{21} to ISP$_1$ on a per destination basis: s^d_{21} is a vector of flows to be sent to each of the destinations in ISP$_1$ from ISP$_2$. Even though the ISPs may have multiple peering links, to facilitate easier understanding, we explain our model using two bi-directional peering links (Figure 1). The model generalizes readily to multiple peering links. We assume that ISP$_1$ splits s^d_{12} so that y^d_{12} goes on the upper link and $(s^d_{12} - y^d_{12})$ goes on the lower link. Similarly ISP$_2$ splits s^d_{21} so that y^d_{21} goes on the upper link and $(s^d_{21} - y^d_{21})$ goes on the lower link.

Optimizing for the utilities would be a no-brainer if the ISPs had no interaction. Then they could optimize their respective utilities independently. What makes it complicated is the interaction of the ISPs through flows sent between each other. These flows make the ISPs utilities inter-

[2]We make the notation precise in Section II-C.1.

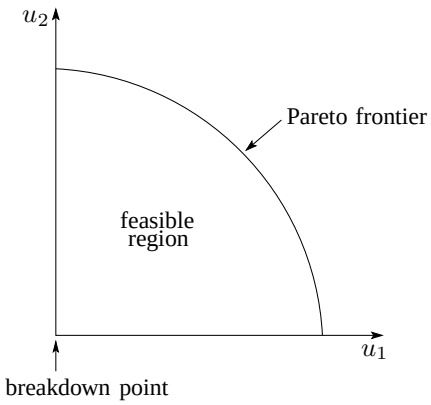

Fig. 2. The feasible region with Pareto efficient frontier.

dependent. These flows between the ISPs are sometimes referred to as *coupling* flows since they cause the ISPs optimization problems to be coupled.

If the ISPs are myopic, i.e., they employ unilateral approaches towards inter-domain TE, they would optimize without paying attention to how the coupling flows affect the other ISPs optimization problem. For example, y_{12}^d is an output of ISP$_1$'s optimization problem and is thus under its control. However, y_{12}^d is an input to ISP$_2$'s optimization problem and thus affects its outcome. Now, if ISP$_1$ is myopic, it will optimize without paying attention to how much it may be hurting ISP$_2$ by determining y_{12}^d myopically. Similarly, ISP$_2$ could determine y_{21}^d myopically. Thus, in this process, both ISPs may end up hurting each other.

When both ISPs route myopically, we denote the ISPs utilities as u_1^{myopic} and u_2^{myopic}, respectively. The question then arises is: Is there any way that the ISPs could somehow cooperate on determining the coupling flows and improve their performance, i.e., achieve $u_1 \geq u_1^{myopic}$ and $u_2 \geq u_2^{myopic}$ such that

1) The gains from cooperation are equitable (or fair) while operating at a Pareto efficient operation point?
2) ISPs don't have to divulge any critical information about their networks?

The answer to the first question lies in the idea of Nash Bargaining [11]. The answer to the second question lies in the idea of decomposition [13]. We explain both of these ideas next.

B. Nash Bargaining

The basic idea behind Nash bargaining is as follows. We assume that the ISP utilities are inter-dependent, concave and cardinal, where by cardinal we mean that the actual values of utilities matter – as opposed to ordinal utilities where only the relative ordering of outcomes matters. Figure 2 shows the feasible region for the two utilities, where the feasible region is defined as the region where both ISPs would do better off compared to the myopic outcome. The myopic outcome is also referred to as the *breakdown point*.

A fair and Pareto efficient outcome, also referred to as the *Nash solution* can be obtained by maximizing the *Nash product* given by $u_1 u_2$. Using the axiomatic theory of cooperative games, it can be shown that when two players (ISPs for us) with equal market power bargain, using threat strategies, they should arrive at the Nash solution. Referring to Figure 2, these threat strategies correspond to the breakdown point, which is the outcome if the ISPs are unable to reach an agreement.

In what follows, we provide a brief summary of the properties of the Nash solution, using the axiomatic bargaining approach. The idea here is that a good bargaining solution should satisfy the following four axioms, which we simply state as follows (see [11] for a detailed discussion):

Pareto efficiency. This is obviously desirable since ISPs prefer more to less.

Symmetry. This says that the solution should provide equal gains from cooperation when the feasible region is symmetric, where by symmetric we mean that the feasible region is agnostic of the player's identities and that it would look the same even if the ISPs utility axis were swapped.

Independence of affine transformations. This requires that the solution should be agnostic of any affine transformations (that is, shifts and scalings) applied to any of the two utilities. So, if the solution is given by (u_1^{NB}, u_2^{NB}) for some utilities (u_1, u_2), and u_1 is scaled and shifted to $\alpha_1 u_1 + \beta_1$, then the solution should change to $(\alpha_1 u_1^{NB} + \beta_1, u_2^{NB})$.

Independence of irrelevant alternatives. This basically says that addition of irrelevant alternatives should not change the solution. That is, for feasible regions F and G, if $(u_1^{NB}, u_2^{NB}) \in solution(F)$, $G \subset F$, and $(u_1^{NB}, u_2^{NB}) \in G$ then $(u_1^{NB}, u_2^{NB}) \in solution(G)$.

It turns out that the Nash solution is the only solution that satisfies these four axioms [10]. In fact, the Nash solution is the only solution that satisfies the following problem that is simultaneously *utilitarian* (Pareto efficient) and *egalitarian* (fair) [11]. That is, the Nash solution solves

$$\begin{array}{ll} \text{maximize} & \alpha_1 u_1 + \alpha_2 u_2 \\ \text{subject to} & \alpha_1 u_1 = \alpha_2 u_2 \\ & (u_1, u_2) \in \mathcal{U} \end{array}$$

for some $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$, where the optimization is over the bounded set $\mathcal{U}$. Note that this scaling by α's does not change the Pareto efficient frontier in Figure 2, i.e., the values of the choice variables resulting in Pareto efficient points remain the same. These α's bring the usually uncomparable u_1 and u_2 on a common footing so that we can talk about fairness in the first place. In particular, the Nash solution is *proportionally fair* [12]. This means that moving away from the Nash solution causes a negative cumulative percentage change in utilities. That is, if (u_1^{NB}, u_2^{NB}) is the Nash solution, and we move to another point (u_1^*, u_2^*), then

$$\sum_{i=1}^{2} \frac{(u_i^* - u_i^{NB})}{u_i^{NB}} \leq 0 \tag{1}$$

We next describe how decomposition can be used to jointly optimize the Nash product without revealing any sensitive information.

C. Decomposition

The idea of decomposition is not new. It has been successfully used to solve large scale optimization problems [13]

and to solve separable problems in a decentralized manner. Moreover, in our case, decomposition allows separate entities in the optimization problem to hide their internal critical information. In what follows, we first develop a precise optimization framework, and then use this framework to explain decomposition.

1) Optimization Formulation: We denote the network topology of ISP$_i$, $i \in \{1,2\}$, by a directed graph $\mathcal{G}_i = (\mathcal{N}_i, \mathcal{L}_i)$ with $n_i = |\mathcal{N}_i|$ nodes and $l_i = |\mathcal{L}_i|$ internal links. We also denote by $\mathcal{P}$ the set of p directed peering links. We then define the incidence matrix for ISP$_i$ as matrix $A_i \in \mathbf{R}^{n_i \times (l_i+p)}$, with $A_{i,jk} = +1$ if link k leaves node j, $A_{i,jk} = -1$ if link k enters node j, and 0 otherwise.

We consider *aggregate* data flows through the network, where we identify each flow by its destination node. We denote by $\mathcal{D}$ the set of all possible destination nodes. For ISP$_i$, we denote the nonnegative amount of flow originating at node j and destined to node $d \in \mathcal{D}$ by $s^d_{i,j}$ ($j \neq d$). When $j = d$, $s^d_{i,d}$ is the negative sum of the flows destined to the node d, thus ensuring flow conservation. We refer to $s^d_i \in \mathbf{R}^n$ as the *source-sink vector*. Note that $s^d_i,\ i \in \{1,2\}$ include s^d_{12} and s^d_{21}, as described in Section II-A. Similarly, for ISP$_i$, we denote the amount of nonnegative flow destined to node d on each internal link $k \in \mathcal{L}_i$ by $x^d_{i,k}$. We call $x^d_i \in \mathbf{R}^l$ the *internal flow vector* for destination d. Finally, we denote the amount of nonnegative flow destined to node d on each peering link $k \in \mathcal{P}$ by y^d_k. We call $y^d \in \mathbf{R}^p$ the *peering flow vector* for destination d. This y^d includes y^d_{12} and y^d_{21}, as described in Section II-A.

Now, we are ready to define the optimization problems in various scenarios. We first present the *Nash product* problem, where ISP would jointly solve

$$\begin{array}{ll} \text{maximize} & u_1 u_2 \\ \text{subject to} & A_1 \left[\begin{array}{c} x^d_1 \\ y^d \end{array} \right] = s^d_1, \quad A_2 \left[\begin{array}{c} x^d_2 \\ y^d \end{array} \right] = s^d_2 \\ & x^d_1 \geq 0, \quad x^d_2 \geq 0, \quad y^d \geq 0, \end{array} \tag{2}$$

for all $d \in \mathcal{D}$, where the optimization variables are x^d_1, x^d_2, and y^d. Here the two equality constraints are the flow conservation constraints for ISP$_1$ and ISP$_2$, respectively, and the last set of inequality constraints ensures that the choice variables are non-negative.[3]

A related problem to (2) is when both ISPs route myopically. The myopic routing schemes that we are particularly interested in are:

1. *Hot potato routing*: Under this approach, each ISP routes inter-domain traffic originating in its network to the closest peering point (i.e., least OSPF-cost). In a way, this attempts to minimize the network resources consumed by inter-domain traffic within the source ISP network. This form of inter-domain traffic exchange is commonly used today.
2. *Nash equilibrium*: Under this approach, ISPs myopically optimize local objectives while iteratively playing best response to each other. Each ISP finds the optimal way to split inter-domain traffic across peering links, given the traffic splits of its neighbor, until no better traffic split can be found. This dynamic eventually finds an equilibrium, also known as the Nash equilibrium [9], from which no ISP has an incentive to deviate.

[3]Other constraints such as link capacity constraints can be readily included.

Under these routing schemes, each ISP myopically solves an optimization problem

$$\begin{array}{ll} \text{maximize} & u_i \\ \text{subject to} & A_i \left[\begin{array}{c} x^d_i \\ y^d \end{array} \right] = \hat{s}^d_i \\ & x^d_i \geq 0, \quad y^d \geq 0, \end{array} \tag{3}$$

for $i = \{1,2\}$ and for all $d \in \mathcal{D}$. Here we use $\hat{s}^d_i$ instead of s^d_i to represent the fact that the myopic routing strategies change the original flow vectors. For example, in hot-potato routing, since each ISP routes the inter-domain flows to the nearest exit points, the flow vector reflects the source of flows being on peering points instead of being on internal nodes. In Nash equilibrium, where the ISPs iterate over the myopic problems based on current incoming flows, the flow vector reflects a similar transformation.

2) Decomposition: Now we look into how we can cast problem (2) in separable form, allowing for a decentralized solution. We face two challenges: first, as it stands, the objective is not separable, and second, the ISPs utilities are coupled through y^d.

We first transform problem (2) into an equivalent problem by taking the logarithm of the objective function. Since the logarithm is an increasing and concave function, this transformation does not change the solution to the original problem [14]. We then get the following equivalent problem

$$\begin{array}{ll} \text{maximize} & \ln(u_1) + \ln(u_2) \\ \text{subject to} & A_1 \left[\begin{array}{c} x^d_1 \\ y^d \end{array} \right] = s^d_1, \quad A_2 \left[\begin{array}{c} x^d_2 \\ y^d \end{array} \right] = s^d_2 \\ & x^d_1 \geq 0, \quad x^d_2 \geq 0, \quad y^d \geq 0, \end{array} \tag{4}$$

We next introduce new nonnegative variables y^d_1 and y^d_2, which are local versions of y^d for ISP$_1$ and ISP$_2$, respectively. Problem (4) can then be rewritten as

$$\begin{array}{ll} \text{maximize} & \ln(u_1) + \ln(u_2) \\ \text{subject to} & A_1 \left[\begin{array}{c} x^d_1 \\ y^d_1 \end{array} \right] = s^d_1, \quad A_2 \left[\begin{array}{c} x^d_2 \\ y^d_2 \end{array} \right] = s^d_2 \\ & x^d_1 \geq 0, \quad y^d_1 \geq 0, \quad x^d_2 \geq 0, \quad y^d_2 \geq 0 \\ & y^d_1 = y^d_2. \end{array} \tag{5}$$

We still have a coupling constraint $y^d_1 = y^d_2$, which we deal with using dual (or pricing) decomposition [13], which is outlined next.

We first write the partial Lagrangian of problem (5), with respect to the coupling constraint, as

$$L(x^d_1, y^d_1, x^d_2, y^d_2, \lambda^d) = \ln(u_1) + \ln(u_2) + \sum_d (\lambda^d)^T (y^d_1 - y^d_2),$$

where $\lambda^d \in \mathbf{R}^p$ are *Lagrange multipliers* associated with the coupling constraint. This is a separable function in (x^d_1, y^d_1) and (x^d_2, y^d_2). We now solve the dual problem of problem (5), given by

$$\text{minimize} \quad g_1(\lambda^d) + g_2(\lambda^d), \tag{6}$$

where g_1 is given by the optimization problem

$$\begin{array}{ll} \text{maximize} & \ln(u_1) + \sum_d (\lambda^d)^T y_1^d \\ \text{subject to} & A_1 \left[\begin{array}{c} x_1^d \\ y_1^d \end{array} \right] = s_1^d, \\ & x_1^d \geq 0, \quad y_1^d \geq 0, \end{array} \tag{7}$$

and g_2 is given by the optimization problem

$$\begin{array}{ll} \text{maximize} & \ln(u_2) - \sum_d (\lambda^d)^T y_2^d \\ \text{subject to} & A_2 \left[\begin{array}{c} x_2^d \\ y_2^d \end{array} \right] = s_2^d, \\ & x_2^d \geq 0, \quad y_2^d \geq 0. \end{array} \tag{8}$$

We note that the optimal value of the dual problem (6) will be equal to the optimal value of the primal problem (5), since problem (5) is a convex problem with a strictly feasible point, and strong duality holds by Slater's condition [14]. We can readily recover the optimal flow allocation from the solution of the dual problem by ensuring, using a small regularization term, that the objective functions are strictly convex (or concave) [14, §5.5.5].

The dual problem (6) is also referred to as the *master problem*. We can solve the master problem using various iterative methods. We choose the subgradient method [16] since it requires very little coordination between problems (7) and (8) and allows encapsulation of the internal data.

The subgradient method requires subgradients of g_1 and g_2. A subgradient of g_1 is evaluated as follows. We first find $\bar{x}_1^d$ and $\bar{y}_1^d$ that minimize

$$\ln(u_1) + \sum_d (\lambda^d)^T y_1^d$$

over x_1^d and y_1^d. Then a subgradient of g_1 at λ^d is given by $\bar{y}_1^d$. Similarly, a subgradient of g_2 at λ^d is given by $-\bar{y}_2^d$. Thus, a subgradient of the dual function $g = g_1 + g_2$ is given by $\bar{y}_1^d - \bar{y}_2^d$, which is nothing more than the consistency check for the coupling constraint.

Dual decomposition, using the subgradient method for solving master problem, then gives the following algorithm:

repeat

1. *Solve the subproblems* (7) *and* (8). Obtain $\bar{y}_1^d$, $\bar{y}_2^d$.
2. *Update master* (6) *subgradient:* $g := \bar{y}_1^d - \bar{y}_2^d$.
3. *Update master* (6) *prices:* $\lambda^d := \lambda^d - \alpha_k g$.

Here α^k is the step size at the kth iteration. We use a constant stepsize, which guarantees convergence to an ϵ-ball around the optimal solution, e.g., see [16] for more details. The subgradient method does not have a good stopping criterion, and in practice it is often terminated when there is no additional progress in the minimization.

We note the following about the proposed procedure:

1) The sub-problems in Step 1 are independent and can be solved by the ISPs independently of each other. Thus, we achieve our objective of not revealing critical information about the internal networks.
2) The updating of the Lagrange multipliers in Step 3 can happen in many ways. One way is for the ISPs to announce the local versions of the coupling flows, i.e., y_1^d and y_2^d. Now, they can both calculate the new Lagrange multipliers.
3) In our simulation experiments over real ISP topologies, we find that this process typically converges in 50-100 iterations to well within the optimal solution using a fixed step size.

III. Practical Issues

A. Implementation and Deployment

We observe that the easiest path to the adoption of our approach is when individual ISPs employ it in conjunction with centralized routing platforms such as rcp [17] or 4d [18]. In this set-up, the centralized routing controller of an ISP executes the protocol in conjunction with the controllers of its neighbors. The controllers exchange prices, and negotiate flow splits. Each controller then converts the negotiated solution into appropriate forwarding table updates on ISP routers. While it may be possible to implement our approach in a completely distributed manner (e.g. where individual routers participate in negotiation), we believe that the above approach is a simpler alternative.

B. Communication Complexity

The master problem (6) in our protocol is solved using the subgradient method which typically takes 50-100 iterations to converge (see section 10.3.2 in [19]). Denoting the number of subgradient method iterations as I_s, the number of peering links as p, and the maximum number of flows as $d_{\max} = n_1 + n_2$, where $n_i,\ i \in \{1, 2\}$ are the number of nodes in ISP_1 and ISP_2, respectively, we need to communicate $\mathcal{O}(p \times d_{\max})$ real numbers per iteration, or $\mathcal{O}(I_s \times p \times d_{\max})$ real numbers total. These can be converted to bits assuming B (typically 32 or 64) bits per real number. Plugging in values, we see that the protocols requires a total of about 8MB of communication for a pair of ISPs with a total of 500 nodes and 20 peering links.

C. Computational Complexity

The subproblems (7) and (8) in our protocol are solved using interior-point methods [14], [19]. Theoretically, these methods have polynomial complexity in the number of variables , i.e., in $(p+\max(l_1, l_2)) \times d_{\max}$, where p and $d_{\max}$ are as defined in the previous subsection and $l_i,\ i \in \{1, 2\}$ are the number of internal links in ISP_1 and ISP_2, respectively. In practice, however, since we can exploit inherent structure in our problems, these methods can be efficiently implemented to solve the subproblems in constant time even for large ISP networks. For example, our implementation takes about 1 second to solve a subproblem with $p+l = 100$ links and $d = 100$ destinations, which translates to $10,000$ variables.

IV. Protocol Extensions

In this section, we discuss simple extensions to our basic framework that show: (1) how our framework can apply to multiple peering ISPs; (2) how to accommodate single link failures; and (3) how to react quickly under arbitrary failures and changes in traffic demands.

A. Multi-ISP Extension

So far, our discussion has focused on a pair of neighboring ISPs. How do we extend this to multiple ISPs peering in a pairwise manner? We note that this is a non-issue if peers are not used for transit. In that case, our basic framework simply applies pairwise to multiple ISPs.

It becomes more interesting when peers can be used for transit (this can be arranged through explicit agreement). As an example, consider three ISPs – ISP_1, ISP_2 and ISP_3 – with the agreement that ISP_1 can send traffic to destinations in ISP_3 either via direct peering links or through ISP_2 (as transit). The key hurdle in facilitating this setting is that our framework requires all destination demands to be known between any pair of ISPs for computing the shadow prices on peering links.

We outline a simple way to tackle this: ISP_1 determines a-priori the demand splits – between direct and transit routes – for each destination in ISP_3. These direct and transit demands are then used in our protocol as actual demands between ISP_1-ISP_3 and between ISP_2-ISP_3, respectively. As for ISP_1-ISP_2, the total transit demand (that is, the sum of transit demands) is destined to a virtual node that is a-priori agreed upon by ISP_1 and ISP_2. This virtual node is assumed to reside behind the ISP_2-ISP_3 interface and represents all the peering link on this interface - thus, ensuring that all transit traffic destined to ISP_3 is eventually routed to one of these peering links.

We note that this scheme, while practical, is not flexible enough: e.g. ISP_1 cannot dynamically change traffic volumes between transit and direct links for specific destinations. We hope to address this issue in future work.

B. Making the Protocol "Incremental"

In the general case, ISPs would run our protocol at certain times of the day in a somewhat semi-static manner (this might correspond with the time granularity at which demand information is collected). However, traffic demands may change suddenly due to phenomena like denial of service attacks, flash crowds, etc. In addition, links may fail, requiring the ISP to reroute its traffic. How can we extend our protocol to react to incremental changes in internal topologies and in traffic demand? We look at various scenarios below.

1) Single Link Failures: We now show how our framework can deal with single link failures in real time, eliminating the need for dynamic re-negotiation of flow splits. ISPs can identify small lists of links that may fail with high probability (e.g. planned outages, or based on historical data). Say ISP_1 and ISP_2 identify NF_1 and NF_2 number of links, respectively. Assuming the probability of simultaneous multiple link failures to be very small, only a single link would fail at a time in either of the domains. Thus, the ISPs would need to run the basic algorithm NF_1+NF_2+1 times – once for the default (no failure) case and $NF_1 + NF_2$ times to cover each of the single link failures – and store the resulting flow splits for each. These could be indexed using previously agreed upon unique keys. Upon failure of one of these links, the ISP with the failed link can notify the other ISP that a link has failed and that they need to switch to the flow splits corresponding to the failure. They can then switch to the new flow splits as soon as possible, preferably in a coordinated fashion. There is no need for re-negotiation.

2) Dealing with Arbitrary Failures or Changes in Demand: We now show how to accommodate any link failure and significant shifts in traffic demands. Say a link fails in ISP_1. How should the ISP modify its local routing to deal with this, while minimizing its own cost and not impacting the other ISP adversely? A similar issue arises when traffic volumes for certain ingress-egress pairs change suddenly (e.g. due to flash crowds).

The answer lies in using the equilibrium shadow prices arising out of our protocol. These indicate how expensive (or cheap) it would be for the receiving ISP if the corresponding flow splits are changed. When sudden internal changes occur, the sender can use this information to check if local changes may be performed in order to accommodate the change and yet not impact the neighbor in any significant manner. If the impact is likely to be low, the sender can make local changes. If the impact is likely to be high, the sender will have to re-negotiate or bargain the prices.

C. A Note on Incentive Compatibility

It is well known that Nash Bargaining is not incentive compatible [20]. Therefore our approach is susceptible to cheating. However, we believe that this will not hinder ISPs from adopting our approach, for two reasons. First, we believe that the real world is more cooperative than often depicted in the non-cooperative game theory setting, and ISPs are honestly trying to improve their performance. Second, since our approach guarantees that ISPs see non-negative improvement when compared to their default strategies (e.g. hot-potato routing or Nash equilibrium), a cheating ISP may be able to gain unfair advantage but it *cannot degrade* the performance of an honest neighbor compared to the neighbor's default (that is, the breakdown point). Nevertheless, we hope to address incentive compatibility in future work.

V. Experimental Results

We conduct simulation experiments to evaluate our protocol. We use Rocketfuel ISP maps which contain PoP-level connectivity information [21]. The links in each map are annotated with the propagation latency, as well as the inferred OSPF link weights employed by the ISP for its internal routing [22]. The maps also include information on the peering locations of neighboring ISPs.

Two key components which are missing from the maps are the traffic demand matrix (both intra, and inter-domain), and the link capacities. For the former, we use gravity-based models [23] where the demand between a pair of cities (or PoPs) is proportional to the product of their populations (the populations can be obtained from public databases). Also, we assume that the demand matrix is symmetric. This model applies to both intra- and inter-domain traffic[4].

For the latter, we assume that all links in an ISP have the same capacity, where the capacity is computed as follows:

[4]We use a lower proportionality constant for inter-domain traffic.

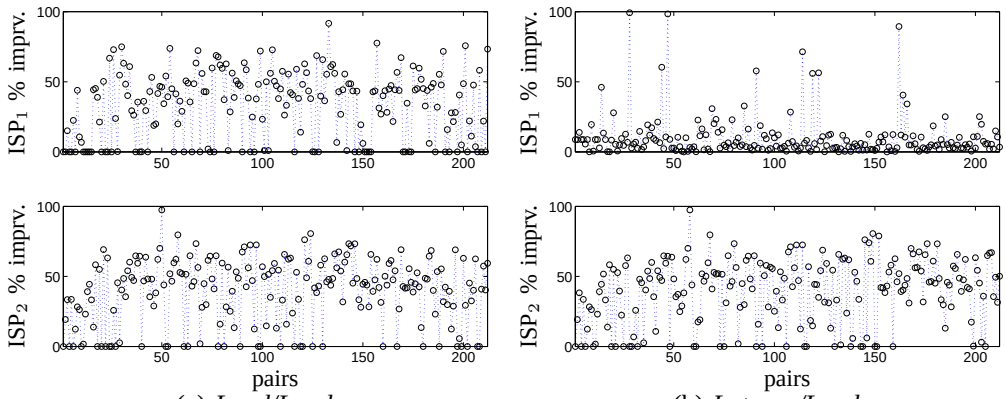

Fig. 3. Nash bargaining compared with hot-potato routing. The top (bottom) graph plots the *percentage improvement* in the optimization objective for ISP_1 (ISP_2) on the y-axis for each ISP pair. The ISP pairs are shown on the x-axis.

we use the ISP link weights to compute the best routes between different PoP pairs (breaking ties randomly). We then identify the most heavily loaded link and set its capacity to be twice the total demand carried by it. We use the same capacity for all other links. We tried with other link capacity assignments (such as a bi-modal distribution), but found the results are qualitatively unchanged. In all, we conduct simulations over 212 ISP pairs.

We simulate two different ISP optimization objectives: (1) *Load*: the goal of the ISP is to minimize the maximum link load in its network. Here, the load of a link is the traffic volume it carries divided by its capacity; (2) *Latency*: the goal of the ISP is to minimize the maximum latency incurred by the traffic it carries. When a traffic demand is split between multiple paths, we compute the weighted latency for the demand-split, where the weight is simply the fraction of the demand routed on a path.

Our simulations compare the Nash bargaining protocol with three other approaches: hot-potato and Nash equilibrium are myopic routing approaches described in Section II-C.1. The third one is *Global optimum* routing. Under this approach, we assume that both the domains are under the control of a central arbitrator who optimizes for a "global" objective. This applies to the specific case where the neighboring ISPs have similar optimization objectives. As an example, if both ISPs want to minimize the maximum load on their internal links, the central arbitrator minimizes the maximum load on links in either ISP. Note that in this approach, one ISP may be penalized while the other ISP benefits.

A. Nash Bargaining vs Hot-Potato Routing

In Figure 3, we compare the performance of our approach against the case where the ISPs employ hot-potato routing. Here, we consider two situations: one where both ISPs employ the same utility - *Load* - shown in (a), and the other where ISP_1 employs *Latency* while ISP_2 employs *Load*, shown in (b).

We make the following observations: First, the objective of either ISP always improves, irrespective of whether the ISPs are optimizing similar objectives or not. In some cases, the value of the objective for one of the ISPs improves two-fold – this can be observed in both 3(a) and (b). In other cases both ISPs see $> 50\%$ improvement each. These results show that, in practice, peering ISPs can both gain significantly from shedding their unilateral TE approaches and adopting the cooperative Nash bargaining-based approach we propose.

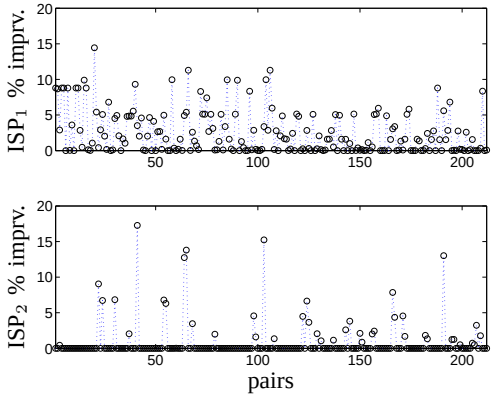

Fig. 4. Nash bargaining compared with Nash equilibrium. The optimization objective for ISP_1 is *Latency* and for ISP_2 is *Load*.

Second, we note that the percentage improvements for the two ISPs are not necessarily equal. In some cases, one ISP gains significantly while the other sees no improvement at all – see 3(a) around ISP-pair 150. This effect is especially pronounced for 3(b) where the ISP utilities are different. The asymmetry in the gains arises due to two reasons: (1) The default strategy (hot potato) may already be offering fairly good performance to one of the participants. Nash bargaining offers incremental benefits. This is in agreement with the observations in [5]. (2) When utilities are dissimilar and therefore not directly comparable, we cannot expect identical percentage improvements anyway.

Nevertheless, as mentioned in Section II, our approach offers *proportional fairness*, a highly desirable property. We illustrate this in Section V-C.

B. Nash Bargaining vs Nash Equilibrium

In Figure 4, we compare the performance from Nash bargaining against Nash equilibrium when ISP_1 optimizes the *Latency* objective and ISP_2 optimizes the *Load* objective. As explained above, Nash equilibrium arises when each ISPs

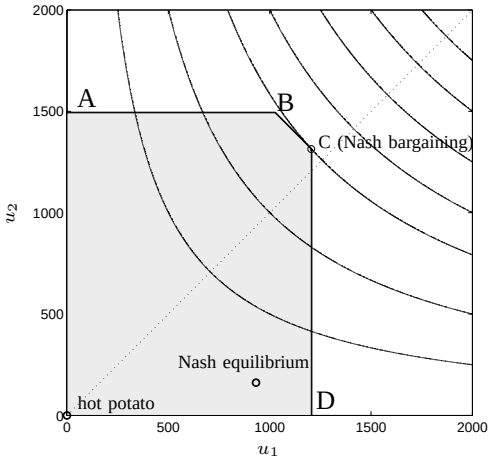

Fig. 5. Feasible region with Pareto efficient frontier. The utilities shown correspond to $C - u$, where C is some large constant, and u is the value of the *Load* objective. This transformation does not change the guarantees offered by our solution.

optimizes its local objective while playing best responses to its neighbor. We note that the Nash equilibrium reflects the behavior of selfish (myopic) and smart ISPs, while hot-potato is a naive greedy strategy. However, unlike hot-potato routing, the Nash equilibrium may be hard to realize in practice, since convergence in finite time is not always guaranteed. In our simulation of Nash equilibrium, we simply ignore cases where the equilibrium is not found after a threshold amount of time.

As with hot-potato routing, we note that Nash bargaining offers superior performance to both ISPs when compared to the performance at the Nash equilibrium. This further establishes the benefit of bi-lateral co-operation in inter-domain traffic engineering.

We note another interesting fact from Figures 3 and 4: the performance of the ISPs at Nash equilibrium seems better than that from hot-potato routing (we did find a negligible fraction of cases where hot-potato performed better than Nash equilibrium). This points to the fact that even among myopic unilateral approaches, the commonly-used hot-potato routing is not the optimal!

C. Efficiency and Fairness

We illustrate the fact that our approach yields a Pareto-efficient and fair solution using a spot-study of a pair of peering ISPs with AS numbers 1 and 5650. ISP_1 has 110 bi-directional links and 42 nodes, whereas ISP_2 has 54 bi-directional links and 22 nodes. In addition, there are 4 bi-directional peering links. Figure 5 shows the feasible region (shaded gray) when both ISPs employ the *Load* utility. It also shows the indifference curves for $u_1 u_2$ as well as the Nash equilibrium and hot-potato points. Our Nash bargaining approach finds the unique solution denoted by point C, with $(u_1^{NB}, u_2^{NB}) = (1205.4, 1314.0)$. This is clearly Pareto efficient since we can't improve the performance of one ISP without hurting the other one.

As a further testament to the quality of the solution found by Nash bargaining, we compare the performance of our approach to that obtained by globally optimal routing, when the objectives of both ISPs are *Load*. Specifically, we compare the load on the maximum loaded link in the global routing case, against the higher among the loads on the most loaded links in the two ISPs when Nash bargaining is used. For all ISP pairs, we found these to be identical!

We next illustrate that the Nash bargaining solution (u_1^{NB}, u_2^{NB}) is proportionally fair, as defined by (1). This is clearly satisfied for (u_1^*, u_2^*) on line segment $[C, D]$ since $u_1^* - u_1^{NB} = 0$ and $(u_2^* - u_2^{NB})/u_2^{NB} \leq 0$. We next show that (1) is satisfied for (u_1^*, u_2^*) on line segment $[B, C]$. These points satisfy $u_2^* - u_2^{NB} = -0.95(u_1^* - u_1^{NB})$. Plugging this in (1), we get

$$\begin{aligned}\frac{(u_1^* - u_1^{NB})}{u_1^{NB}} + \frac{(u_2^* - u_2^{NB})}{u_2^{NB}} &= \frac{(u_1^* - u_1^{NB})}{u_1^{NB}} + \frac{m(u_1^* - u_1^{NB})}{u_2^{NB}} \\ &\approx 0.001(u_1^* - u_1^{NB}) \leq 0.\end{aligned}$$

Similarly, it can be shown that (1) is satisfied for (u_1^*, u_2^*) on line segment $[A, B]$.

VI. Related Work

Inter-domain TE: A single ISP's view point: Several papers on Inter-domain traffic engineering have focused on studying the problem from the point of view of one of the participants (See for example [24], [25], [26]). These papers address issues such as tweaking OSPF weights to achieve fine-grained control over egress points for inter-domain traffic [24], AS-path pre-pending to control the ingress points of inter-domain traffic [26], and best common practices for achieving predictable and stable route selection for inter-domain traffic [25]. These papers differ from our paper in the key aspect that we focus on the benefits of bi-lateral cooperation among ISPs, while the above papers focus on tweaking the unilateral decisions of a single ISP. We do note that our technique can operate in conjunction with the above approaches: once our technique determines the traffic volumes to route via different exit points, the above approaches can be used to tune the configurations of routers in order to achieve the desired effect.

Inter-domain TE based on cooperation: The paper that is perhaps the closest in its goal to our work is Mahajan et al.'s "negotiation-based routing". In [5], [6], Mahajan et al. propose an approach where peering ISPs use a "negotiation protocol" to exchange opaque preference classes for inter-domain flows. Using these opaque preference classes, an ISP can indicate the preferred entry points for traffic arriving from its neighbor. No other internal information is exposed. The negotiation protocol proceeds in iterations, with ISPs taking turns in stating their preference for each inter-domain flow, until they arrive at mutually acceptable mappings of all inter-domain flows to network entry points. Thus, cooperative traffic engineering is achieved.

This approach was shown to work well in practical settings. However, it suffers from the following limitations: it is heuristic-based and, so, does not offer any provable guarantees. First, it does not guarantee that the mutually acceptable outcome lies on the Pareto frontier. Second, it does not make the idea of fairness concrete. For example, if the ISPs are optimizing directly comparable objective functions then the final outcome should satisfy the well-known min-max criteria which guarantees equal gains from

cooperation. Fairness becomes even harder to provide when the ISPs are optimizing different objective functions. Our work directly addresses the above issues.

Optimization-based TE approaches: A few research studies have explored the applicability of optimization techniques to traffic engineering problems. Representative examples include [15], [27], [28]. The focus of these papers is on intra-domain traffic engineering. [15] shows how to cast network-wide traffic engineering goals as optimization problems, and how to transform the results into OSPF weights. [27], [28] show how to jointly optimize multiple objectives in traffic engineering (such as congestion control and routing, or pricing and routing). Our paper extends this body of work in a new direction: joint-optimization of traffic engineering objectives of multiple ISPs. Our contribution is in showing that this optimization is separable, and therefore, can be performed in a distributed fashion without requiring the participants to reveal any sensitive internal information.

Nash Bargaining in Other Applications: The application of Nash bargaining to multi-criteria optimization is not new. It has been applied to many problems in networking. [29] applied it to ensure fairness in a network flow control problem. [30] applied it to allocate bandwidth fairly. It has also been shown in the influential work of Kelly [12] that Nash bargaining ensures proportional fairness in a TCP setting. To the best of our knowledge, ours is the first work to apply Nash bargaining to inter-domain traffic engineering.

VII. Summary

In this paper, we presented a new inter-domain traffic engineering protocol that is Pareto-efficient, fair and does not require ISPs to reveal internal information. Our approach uses ideas from co-operative game theory (specifically, Nash bargaining) as well as a host of tricks from non-linear optimization (such as, dual decomposition and the sub-gradient method) to achieve the above desirable properties.

We simulated our approach over real ISP topologies and traffic demands. We found that our approach can offer significant improvement both relative to prevalent approaches such as hot-potato routing, as well as more sophisticated selfish inter-domain TE approaches. We also empirically verified the fairness and efficiency properties of our approach.

Our solution provides provable guarantees that are missing from the state-of-the-art in inter-domain traffic engineering. Therefore, our approach is very amenable to adoption by ISPs today.

References

[1] R. Mahajan, D. Wetherall, and T. Anderson, "Towards coordinated interdomain traffic engineering," in *HotNets-III*, 2004.

[2] N. Spring, R. Mahajan, and T. Anderson, "Quantifying the Causes of Internet Path Inflation," in *SIGCOMM*, Aug. 2003.

[3] R. Teixeira, T. Griffin, G. Voelker, and A. Shaikh, "Network sensitivity to hot potato disruptions," in *SIGCOMM*, 2004.

[4] J. Winick, S. Jamin, and J. Rexford, "Traffic Engineering between Neighboring Domains," 2002, manuscript.

[5] R. Mahajan, D. Wetherall, and T. Anderson, "Negotiation-Based Routing Between Neighboring ISPs," in *Proc. Second Networked Systems Design and Implementation*, May 2005.

[6] R. Mahajan, "Practical and efficient internet routing with competing interests," Ph.D. dissertation, University of Washington, 2005.

[7] S. Kumar, R. Randhawa, and T. Yahalom, "Fairness in Capacity Allocation and Scheduling: Non-Monetary Mechanisms," Working paper, 2005.

[8] R. E. Steur, *Multiple Criteria Optimization: Theory, Computation, and Application*. Wiley and Sons, 1986.

[9] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic Theory*. Oxford University Press, 1995.

[10] J. F. Nash, "The bargaining problem," *Econometric*, vol. 28, pp. 155–162, August 1950.

[11] R. B. Myerson, *Game Theory: Analysis of Conflict*. Harvard University Press, 1991.

[12] F. P. Kelly, A. K. H. Mallu, and D. K. H. Tan, "Rate control for communication networks:shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, 1998.

[13] L. Lasdon, *Optimization Theory for Large Systems*. Mcmillan Series in Operations Research, 1970.

[14] S. Boyd and L. Vanderberghe, *Convex Optimization*. Cambridge University Press, 2004.

[15] B. Fortz and M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights," in *Infocom*, 2000.

[16] N. Z. Shor, *Minimization Methods for Non-differentiable Functions*, ser. Springer Series in Computational Mathematics. Springer, 1985.

[17] M. Caesar, D. Caldwell, N. Feamster, J. Rexford, A. Shaikh, and J. van der Merwe, "Design and implementation of a routing control platform," in *NSDI*, 2005.

[18] A. Greenberg, G. Hjalmtysson, D. A. Maltz, A. Meyers, J. Rexford, G. Xie, H. Yan, J. Zhan, and H. Zhang, "A clean slate 4d approach to network control and management," *SIGCOMM Computer Communication Review*, Oct. 2005.

[19] D. P. Bertsekas, *Network Optimization: Continuous and Discrete Models*. Athena Scientific, 1998.

[20] W. Thomson, "Manipulability of Resource Allocation Mechanisms," *Review of Economic Studies*, vol. 51, pp. 447–60, July 1984.

[21] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP Topologies with Rocketfuel," in *SIGCOMM*, Pittsburgh, PA, Aug. 2002.

[22] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson, "Inferring link weights using end-to-end measurements," in *Internet Measurement Workshop*, Nov. 2002.

[23] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale ip traffic matrices from link loads," in *SIGMETRICS*, 2003.

[24] U. Steve and Q. Bruno, "Tweak-it: Bgp-based interdomain traffic engineering for transit ases," in *Next generation Internet networks traffic engineering (NGI)*, 2005, pp. 75–82.

[25] N. Feamster, J. Borkenhagen, and J. Rexford, "Guidelines for Interdomain Traffic Engineering," *ACM SIGCOMM Computer Communication Review*, Oct. 2003.

[26] R. Gao, C. Dovrolis, and E. Zegura, "Interdomain ingress traffic engineering through optimized as-path prepending," in *Proceedings of IFIP Networking*, 2005.

[27] J. He and M. Bessler and M. Chiang and J. Rexford, "Towards Robust Multi-layer Traffic Engineering: Optimization of Congestion Control and Routing," 2006. [Online]. Available: www.princeton.edu/~jhe/research/jsac2.pdf

[28] D. Mitra, K. Ramakrishnan, and Q. Wang, "Combined economic modeling and traffic engineering: Joint optimization of pricing and routing in multi-service networks," in *17th International Teletraffic Congress*, 2001.

[29] R. Mazumdar, L. G. Mason, and C. Doulgligeris, "Fairness in network optimal flow control: Optimality of product forms," *IEEE Trans. Communications*, vol. 39, no. 5, pp. 775–782, 1991.

[30] C. Touati, E. Altman, and J. Galtier, "Utility Based Fair Bandwidth Allocation," Unpublished manuscript, 2002.