Cooperative Content Distribution and Traffic Engineering in an ISP Network

Wenjie Jiang[†], Rui Zhang-Shen[†], Jennifer Rexford[†], Mung Chiang[*]
[†]Department of Computer Science, and [*]Department of Electrical Engineering
Princeton University
{wenjiej, rz, jrex, chiangm}@princeton.edu

ABSTRACT

Traditionally, Internet Service Providers (ISPs) make profit by providing Internet connectivity, while content providers (CPs) play the more lucrative role of delivering content to users. As network connectivity is increasingly a commodity, ISPs have a strong incentive to offer content to their subscribers by deploying their own content distribution infrastructure. Providing content services in an ISP network presents new opportunities for coordination between traffic engineering (to select efficient routes for the traffic) and server selection (to match servers with subscribers). In this work, we develop a mathematical framework that considers three models with an increasing amount of cooperation between the ISP and the CP. We show that separating server selection and traffic engineering leads to sub-optimal equilibria, even when the CP is given accurate and timely information about the ISP's network in a partial cooperation. More surprisingly, extra visibility may result in a less efficient outcome and such performance degradation can be unbounded. Leveraging ideas from cooperative game theory, we propose an architecture based on the concept of Nash bargaining solution. Simulations on realistic backbone topologies are performed to quantify the performance differences among the three models. Our results apply both when a network provider attempts to provide content, and when separate ISP and CP entities wish to cooperate. This study is a step toward a systematic understanding of the interactions between those who provide and operate networks and those who generate and distribute content.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: [Performance attributes]

General Terms

Design, Economics, Performance

1. INTRODUCTION

Internet Service Providers (ISPs) and content providers (CPs) are traditionally independent entities. ISPs only provide connectivity, or the "pipes" to transport content. As in most transportation businesses, connectivity and bandwidth are becoming commodities and ISPs find their profit margin shrinking [1]. At the same time, content providers generate revenue by utilizing existing connectivity to deliver content to ISPs' customers. This motivates ISPs to host and distribute content to their customers. Content can be enterprise-oriented, like web-based services, or residential-based, like triple play as in AT&T's U-Verse [2] and Verizon FiOS [3] deployments. When ISPs and CPs operate independently, they optimize their performance without much cooperation, even though they influence each other indirectly. When ISPs deploy content services or seek cooperation with CP, they face the question of how much can be gained from such cooperation and what kind of cooperation should be pursued.

A traditional service provider's primary role is to deploy infrastructure, manage connectivity, and balance traffic load inside its network. In particular, an ISP solves the *traffic engineering* (TE) problem, i.e., adjusting the routing configuration to the prevailing traffic. The goal of TE is to ensure efficient routing to minimize congestion, so that users experience low packet loss, high throughput, and low latency, and that the network can gracefully absorb flash crowds.

To offer its own content service, an ISP replicates content over a number of strategically-placed servers and directs requests to different servers. The CP, whether as a separate business entity or as a new part of an ISP, solves the *server selection* (SS) problem, i.e., determining which servers should deliver content to each end user. The goal of SS is to meet user demand, minimize network latency to reduce user waiting time, and balance server load to increase throughput.

To offer both network connectivity and content delivery, an ISP is faced with coupled TE and SS problems, as shown in Figure 1. TE and SS interact because TE affects the routes that carry the CP's traffic, and SS affects the offered load seen by the network. Actually, the degrees of freedom are also the "mirror-image" of each other: the ISP controls routing matrix, which is the constant parameter in the SS problem, while the CP controls traffic matrix, which is the constant parameter in the TE problem.

In this paper, we study several approaches an ISP could take in managing traffic engineering and server selection, ranging from running the two systems independently to designing a joint system. We refer to CP as the part of the system that manages server selection, whether it is performed directly by the ISP or by a separate company that cooperates with the ISP. This study allows us to explore a migration path from the status-quo to different models of synergistic traffic management. In particular, we consider three scenarios with increasing amounts of cooperation between traffic engineering and server selection:

SIGMETRICS/Performance'09, June 15–19, 2009, Seattle, WA, USA.
Copyright 2009 ACM 978-1-60558-511-6/09/06 ...$5.00.

	Optimality Gap	Information Exchange	Fairness	Architectural Change
Model I	Large, not Pareto-optimal	No exchange Measurement only	No	Current practice
Model II	Improved, not Pareto-optimal Social-optimal in special case More info. may hurt the CP	Topology, routes, capacity Background traffic level	No	Minor CP changes Better SS algorithm
Model III	Pareto-optimal 5-30% performance improvement	Topology Link prices	Yes	Clean-slate design Incrementally deployable CP given more control

Table 1: Summary of results and engineering implications.

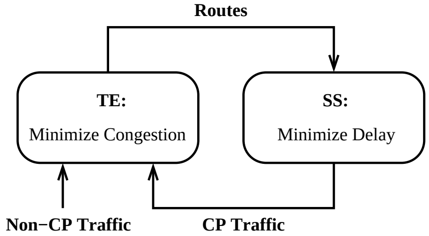


Figure 1: The interaction between traffic engineering (TE) and server selection (SS).

- **Model I:** no cooperation (current practice).
- **Model II:** improved visibility (sharing information).
- **Model III:** a joint design (sharing control).

Model I. Content services could be provided by a CDN that runs independently on the ISP network. However, the CP has limited visibility into the underlying network topology and routing, and therefore has limited ability to predict user performance in a timely and accurate manner. We model a scenario where the CP measures the end-to-end latency of the network and greedily assigns each user to the servers with the lowest latency to the user, a strategy some CPs employ today [4]. We call this *SS with e2e info*. In addition, TE assumes the offered traffic is unaffected by its routing decisions, despite the fact that routing changes can affect path latencies and therefore the CP's traffic. When the TE problem and the SS problem are solved separately, their interaction can be modeled as a game in which they take turns to optimize their own networks and settle in a Nash equilibrium, which may not be *Pareto optimal*.

Not surprisingly, performing TE and SS independently is often sub-optimal because (i) server selection is based on incomplete (and perhaps inaccurate) information about network conditions and (ii) the two systems, acting alone, may miss opportunities for a joint selection of servers and routes. Models II and III capture these two issues, allowing us to understand which factor is more important in practice.

Model II. Greater visibility into network conditions should enable the CP to make better decisions. There are, in general, four types of information that could be shared: (i) physical topology information [5, 6, 7], (ii) logical connectivity information, e.g., routing in the ISP network, (iii) dynamic properties of links, e.g., OSPF link weights, background traffic, and congestion level, and (iv) dynamic properties of nodes, e.g., bandwidth and processing power that can be shared. Our work focuses on a combination of these types of information, i.e., (i)-(iii), so that the CP is able to solve the SS problem more efficiently, i.e., to find the *optimal server selection*.

Sharing information requires minimal extensions to existing solutions for TE and SS, making it amenable to incremental deployment. Similar to the results in the parallel work [8], we observe and prove that TE and SS separately optimizing over their own variables is able to converge to a global optimal solution, when the two systems share the same objectives with the absence of background traffic. However, when the two systems have different or even conflicting performance objectives (e.g., SS minimizes end-to-end latency and TE minimizes congestion), the equilibrium is *not* optimal. In addition, we find that model II sometimes performs *worse* than model I—that is, extra visibility into network conditions sometimes leads to a *less efficient* outcome—and the CP's latency degradation can be unbounded. The facts that both Model I and Model II in general do not achieve optimality, and that extra information (Model II) sometimes hurts the performance, motivate us to consider a clean-slate joint design for selecting servers and routes next.

Model III. A joint design should achieve *Pareto optimality* for TE and SS. In particular, our joint design's objective function gives rise to *Nash Bargaining Solution* [9]. The solution not only guarantees *efficiency*, but also *fairness* between synergistic or even conflicting objectives of two players. It is a point on the Pareto optimal curve where both TE and SS have better performance compared to the Nash equilibrium. We then apply the optimization decomposition technique [10] so that the joint design can be implemented in a distributed fashion with a limited amount of information exchange.

The analytical and numerical evaluation of these three models allows us to gain insights for designing a cooperative TE and SS system, summarized in Table 1. The conventional approach of Model I requires minimum information passing, but suffers from sub-optimality and unfairness. Model II requires only minor changes to the CP's server selection algorithm, but the result is still not Pareto optimal and performance is not guaranteed to improve, even possibly degrading in some cases. Model III ensures optimality and fairness through a distributed protocol, requires a moderate increase in information exchange, and is incrementally deployable. Our results show that letting CP have some control over network routing is the key to effective TE and SS cooperation.

We perform numerical simulations on realistic ISP topologies, which allow us to observe the performance gains and loss over a wide range of traffic conditions. The joint design shows significant improvement for both the ISP and the CP. The simulation results further reveal the impact of topologies on the efficiencies and fairness of the three system models.

Our results apply both when a network provider attempts to provide content, and when separate ISP and CP entities wish to cooperate. For instance, an ISP playing both roles would find the optimality analysis useful such that a low efficiency operating region can be avoided. And cooperative ISP and CP would appreciate the

Notation	Description
$G:$	Network graph $G=(V,E)$. V set of nodes, E set of links
$S:$	$S\subset V$, the set of CP servers
$T:$	$T\subset V$, the set of users
$C_l:$	Capacity of link l
$r_l^{ij}:$	Proportion of flow $i\rightarrow j$ traversing link l
$R:$	The routing matrix $R:\{r_l^{ij}\}$, TE variable
R_{bg}	Background routing matrix $R:\{r_l^{ij}\}_{(i,j)\notin S\times T}$
$X:$	Traffic matrix of all communication pairs $X=\{x_{ij}\}_{(i,j)\in V\times V}$
$x_{st}:$	Traffic rate from server s to user t
$X_{cp}:$	$X_{cp}=\{x_{st}\}_{(s,t)\in S\times T}$, SS variable
$M_t:$	User t's demand rate for content
$B_s:$	Service capacity of server s
$x_l^{st}:$	The amount of traffic for (s,t) pair on link l
$\hat{X}_{cp}:$	$\hat{X}_{cp}=\{x_l^{st}\}_{(s,t)\in S\times T}$, the generalized SS variable
$f_l^{cp}:$	CP's traffic on link l
$f_l^{bg}:$	Background traffic on link l
$f_l:$	$f_l=f_l^{cp}+f_l^{bg}$, total traffic on link l. $\vec{f}=\{f_l\}_{l\in E}$
$D_p:$	Delay of path p
$D_l:$	Delay of link l
$g(\cdot):$	Cost function used in ISP traffic engineering
$h(\cdot):$	Cost function used in CP server selection

Table 2: Summary of key notation.

distributed implementation of Nash bargaining solution that allows for an incremental deployment.

The rest of the paper is organized as follows. Section 2 presents a standard model for traffic engineering. Section 3 presents our two models for server selection, when given minimal information (i.e., Model I) and more information (i.e., Model II) about the underlying network. Section 4 studies the interaction between TE and SS as a game and shows that they reach a Nash equilibrium. Section 5 analyzes the efficiency loss of Model I and Model II in general. We show that the Nash equilibria achieved in both models are not Pareto optimal. In particular, we show that more information is not always helpful. Section 6 discusses how to jointly optimize TE and SS by implementing a Nash bargaining solution. We propose an algorithm that allows practical and incremental implementation. We perform large-scale numerical simulations on realistic ISP topologies in Section 7. Finally, Section 8 presents related work, and Section 9 concludes the paper and discusses our future work.

2. TRAFFIC ENGINEERING (TE) MODEL

In this section, we describe the network model and formulate the optimization problem that the standard TE model solves. We also start introducing the notation used in this paper, which is summarized in Table 2.

Consider a network represented by graph $G=(V,E)$, where V denotes the set of nodes and E denotes the set of directed physical links. A node can be a router, a host, or a server. Let x_{ij} denote the rate of flow (i,j), from node i to node j, where $i,j\in V$. Flows are carried on end-to-end paths consisting of some links. One way of modeling routing is $W=\{w_{pl}\}$, i.e., $w_{pl}=1$ if link l is on path p, and 0 otherwise. We do not limit the number of paths so W can include *all* possible paths, but in practice it is often pruned to include only paths that actually carry traffic. The capacity of a link $l\in E$ is $C_l>0$.

Given the traffic demand, traffic engineering changes routing to minimize network congestion. In practice, network operators control routing either by changing OSPF link weights [11] or by establishing MPLS label-switched paths [12]. In this paper we use the multi-commodity flow solution to route traffic, because a) it is optimal, i.e., it gives the routing with minimum congestion cost, and b) it can be realized by routing protocols that use MPLS tunneling, or as recently shown, in a distributed fashion by a new link-state routing protocol PEFT [13]. Let $r_l^{ij}\in[0,1]$ denote the proportion of traffic of flow (i,j) that traverses link l. To realize the multi-commodity flow solution, the network splits each flow over a number of paths. Let $R=\{r_l^{ij}\}$ be the routing matrix.

Let f_l denote the total traffic traversing link l, and we have $f_l=\sum_{(i,j)}x_{ij}\cdot r_l^{ij}$. Now traffic engineering can be formulated as the following optimization problem:

TE($R|X$):

$$
\begin{aligned}
\text{minimize} \quad & TE=\sum_l g_l(f_l) && (1)\\
\text{subject to} \quad & f_l=\sum_{(i,j)}x_{ij}\cdot r_l^{ij}\le C_l,\ \forall l\\
& \sum_{l:l\in\text{In}(v)} r_l^{ij}-\sum_{l:l\in\text{Out}(v)} r_l^{ij}=I_{v=j},\ \forall(i,j),\ \forall v\in V\backslash\{i\}\\
\text{variables} \quad & 0\le r_l^{ij}\le 1,\ \forall(i,j),\ \forall l
\end{aligned}
$$

where $g_l(\cdot)$ represents a link's congestion cost as a function of the load, $I_{v=j}$ is an indicator function which equals 1 if $v=j$ and 0 otherwise, $\text{In}(v)$ denotes the set of incoming links to node v, and $\text{Out}(v)$ denotes the set of outgoing links from node v.

In this model, TE does not differentiate between the CP's traffic and background traffic. In fact, TE assumes a constant traffic matrix X, i.e., the offered load between each pair of nodes, which can either be a point-to-point background traffic flow, or a flow from a CP's server to a user. As we will see later, this common assumption is undermined when the CP performs dynamic server selection.

We only consider the case where cost function $g_l(\cdot)$ is a convex, continuous, and non-decreasing function of f_l. By using such an objective, TE penalizes high link utilization and balances load inside the network. We will discuss later the analytical form of $g_l(\cdot)$ which the ISP may use in practice.

3. SERVER SELECTION (SS) MODELS

While traffic engineering usually assumes that traffic matrix is point-to-point and constant, both assumptions are violated when some or all of the traffic is generated by the CP. A CP usually has many servers that offer the same content, and the servers selected for each user depend on the network conditions. In this section, we present two novel CP models which correspond to models I and II introduced in Section 1. The first one models the current CP operation, where the CP relies on end-to-end *measurement* of the network condition in order to make server selection decisions; the second one models the situation when the CP obtains enough information from the ISP to *calculate* the effect of its actions.

3.1 Server Selection Problem

The CP solves the server selection problem to optimize the perceived performance of all of its users. We first introduce the notation used in modeling server selection. In the ISP's network, let $S\subset V$ denote the set of CP's servers, which are strategically placed at different locations in the network. For simplicity we assume that all content is duplicated at all servers, and our results can be extended to the general case. Let $T\subset V$ denote the set of users who request content from the servers. A user $t \in T$ has a demand for content at rate M_t, which we assume to be constant during the time a CP optimizes its server section. We allow a user to simultaneously download content from multiple servers, because node t can be viewed as an edge router in the ISP's network that aggregates the traffic of many endhosts, which may be served by different servers.

To differentiate the CP's traffic from background traffic, we denote x_{st} as the traffic rate from server s to user t. To satisfy the traffic demand, we need

$$\sum_{s \in S} x_{st} = M_t.$$

In addition, the total amount of traffic aggregated at a server s is limited by its service capacity B_s, i.e.,

$$\sum_{t \in T} x_{st} \le B_s.$$

We denote $X_{cp} = \{x_{st}\}_{s \in S, t \in T}$ as the CP's decision variable.

One of the goals in server selection is to optimize the overall performance of the CP's customers. We use an additive link cost for the CP based on latency models, i.e., each link has a cost, and the end-to-end path cost is the sum of the link costs along the way. As an example, suppose the content is delay-sensitive (e.g., IPTV), and the CP would like to minimize the average or total end-to-end delay of all its users. Let D_p denote the end-to-end latency of a path p, and $D_l(f_l)$ denote the latency of link l, modeled as a convex, non-decreasing, and continuous function of the amount of flow f_l on the link. By definition, $D_p = \sum_{l \in p} D_l(f_l)$. Then the overall latency experienced by all CP's users is

$$\begin{aligned} SS &= \sum_{(s,t)} \sum_{p \in P(s,t)} x_p^{st} \cdot D_p(f) \\ &= \sum_{(s,t)} \sum_{p \in P(s,t)} x_p^{st} \cdot \sum_{l \in p} D_l(f_l) \\ &= \sum_l D_l(f_l) \cdot \sum_{(s,t)} \sum_{p \in P(s,t): l \in p} x_p^{st} \\ &= \sum_l f_l^{cp} \cdot D_l(f_l) \end{aligned} \tag{2}$$

where $P(s,t)$ is the set of paths serving flow (s,t) and x_p^{st} is the amount of flow (s,t) traversing path $p \in P(s,t)$.

Let $h_l(\cdot)$ represent the cost of link l, which we assume is convex, non-decreasing, and continuous. In this example, $h_l(f_l^{cp}, f_l) = f_l^{cp} \cdot D_l(f_l)$. Thus, the link cost $h_l(\cdot)$ is a function of the CP's total traffic f_l^{cp} on the link, as well as the link's total traffic f_l, which also includes background traffic.

Expression (2) provides a simple way to calculate the total user-experienced end-to-end delay—simply sum over all the *links*, but it requires the knowledge of the load on each link, which is possible only in Model II. Without such knowledge (Model I), the CP can rely only on *end-to-end* measurement of delay.

3.2 Server Selection with E2E Info: Model I

In today's Internet architecture, a CP does not have access to an ISP's network information, such as topology, routing, link capacity, or background traffic. Therefore a CP relies on measured or inferred information to optimize its performance. To minimize its users' latencies, for instance, a CP can assign each user to servers with the lowest (measured) end-to-end latency to the user. In practice, content distribution networks like Akamai's server selection algorithm is based on this principle [4]. We call it *SS with e2e info* and use it as our first model.

CP monitors the latencies from all servers to all users, and makes server selection decisions to minimize users' total delay. Since the demand of a user can be arbitrarily divided among the servers, we can think of the CP as greedily assigning each infinitesimal demand to the best server. The placement of this traffic may change the path latency, which is monitored by the CP. Thus, at the equilibrium, the servers which send (nonzero) traffic to a user should have the same end-to-end latency to the user, because otherwise the server with lower latency will be assigned more demand, causing its latency to increase, and the servers not sending traffic to a user should have higher latency than those that serve the user. This is sometimes called the *Wardrop equilibrium* [14]. The SS model with e2e info is very similar to selfish routing [15, 16], where each flow tries to minimize its average latency over multiple paths without coordinating with other flows. It is known that the equilibrium point in selfish routing can be viewed as the solution to a global convex optimization problem [15]. Therefore, SS with e2e info has a unique equilibrium point under mild assumptions.

Although the equilibrium point is well-defined and is the solution to a convex optimization problem, in general it is hard to compute the solution analytically. Thus we leverage the idea of Q-learning [17] to implement a distributed iterative algorithm to find the equilibrium of SS with e2e info. The algorithm is guaranteed to converge even under dynamic network environments with cross traffic and link failures, and hence can be used in practice by the CPs. The detailed description and implementation can be found in [18]. As we will show, SS with e2e info is not optimal. We use it as a baseline for how well a CP can do with only the end-to-end latency measurements.

3.3 Server Selection with Improved Visibility: Model II

We now describe how a CP can optimize server selection given *complete* visibility into the underlying network, but not into the ISP objective. That is, this is the best the CP can do without *changing* the routing in the network. We also present an optimization formulation that allows us to analytically study its performance.

Suppose that content providers are able to either obtain information on network conditions directly from the ISP, or infer it by its measurement infrastructure. In the best case, the CP is able to obtain the complete information about the network, i.e., routing decision and link latency. This situation is characterized by problem (3). To optimize the overall user experience, the CP solves the following cost minimization problem:

SS$(X_{cp}|R)$**:**

$$\begin{aligned} \text{minimize} \quad & SS = \sum_l h_l(f_l^{cp}, f_l) && (3) \\ \text{subject to} \quad & f_l^{cp} = \sum_{(s,t)} x_{st} \cdot r_l^{st}, \ \forall l \\ & f_l = f_l^{cp} + f_l^{bg} \le C_l, \ \forall l \\ & \sum_{s \in S} x_{st} = M_t, \ \forall t \\ & \sum_{t \in T} x_{st} \le B_s, \ \forall s \\ \text{variables} \quad & x_{st} \ge 0, \ \forall (s,t) \end{aligned}$$

where we denote $f_l^{bg} = \sum_{(i,j) \ne (s,t)} x_{ij} \cdot r_l^{ij}$ as the non-CP traffic on link l, which is a parameter to the optimization problem. If the cost function $h_l(\cdot)$ is increasing and convex on the variable f_l^{cp}, one can verify that (3) is a convex optimization problem, hence has a unique global optimal value. In the remainder of this paper, we ignore the server capacity constraint by assuming sufficiently large server capacities, since we are more interested in the impact of SS on the network than the decision of load balancing among congested servers.

SS with improved visibility (3) is amenable to an efficient implementation. The problem can either be solved centrally, e.g., at the CP's central coordinator, or via a distributed algorithm similar to that used for Model I. We solve (3) centrally in our simulations, since we are more interested in the performance improvement brought by complete information than any particular algorithm for implementing it.

4. ANALYZING TE-SS INTERACTION

In this section, we study the interaction between the ISP and the CP when they operate independently without coordination in both Model I and Model II, using a game-theoretic model. The game formulation allows us to analyze the stability condition, i.e., we show that alternating TE and SS optimizations will reach an equilibrium point. In addition, we find that when the ISP and the CP optimize the same system objective, their interaction achieves *global optimality* under Model II. Results in this section are also found in a parallel work [8].

4.1 TE-SS Game and Nash Equilibrium

We start with the formulation of a two-player non-cooperative Nash game that characterizes the TE-SS interaction.

Definition 1. *The* TE-SS game *consists of a tuple* $[N, A, U]$. *The player set* $N = \{isp, cp\}$. *The action set* $A_{isp} = \{R\}$ *and* $A_{cp} = \{X_{cp}\}$, *where the feasible set of* R *and* X_{cp} *are defined by the constraints in (1) and (3) respectively. The utility functions are* $U_{isp} = -TE$ *and* $U_{cp} = -SS$.

Figure 1 shows the interaction between SS and TE. In both Model I and Model II, the ISP plays the best response strategy, i.e., the ISP always optimizes (3) given the CP's strategy X_{cp}. Similarly, the CP plays the best response strategy in Model II when given full information. However, the CP's strategy in Model I is not the best response, since it does not optimize (3) due to the lack of network visibility. Indeed, the utility the CP implicitly optimizes in SS with e2e info is [15]

$$U_{cp} = -\sum_{l \in E} \int_0^{f_l} D_l(u)du$$

This later helps us understand the stability conditions of the game.

Consider a particular game procedure in which the ISP and the CP take turns to optimize their own objectives by varying their own decision variables, treating that of the other player as constant. Specifically, in the $(k+1)$-th iteration, we have

$$\begin{aligned} R^{(k+1)} &= \underset{R}{\text{argmin}}\ TE(X_{cp}^{(k)}) \\ X_{cp}^{(k+1)} &= \underset{X_{cp}}{\text{argmin}}\ SS(R^{(k+1)}) \end{aligned} \qquad (4)$$

Note that the two optimization problems may be solved on different timescales. The ISP runs traffic engineering at the timescale of hours, although it could run on a much smaller timescale. Depending on the CP's design choices, server selection is optimized a few times a day, or at a smaller timescale like seconds or minutes of a typical content transfer duration. We assume that each player has fully solved its optimization problem before the other one starts.

Next we prove the existence of Nash equilibrium of the TE-SS game. We establish the stability condition when two players use general cost functions $g_l(\cdot)$ and $h_l(\cdot)$ that are continuous, non-decreasing, and convex. While TE's formulation is the same in Model I and Model II, we consider the two SS models, i.e., SS with e2e info and SS with improved visibility.

Theorem 1. *The TE-SS game has a Nash equilibrium for both Model I and Model II.*

Proof Sketch: It suffices to show that (i) each player's strategy space is a nonempty compact convex subset, and (ii) each player's utility function is continuous and quasi-concave on its strategy space, and follow the standard proof in [19]. The ISP's strategy space is defined by the constraint set of (1), which are affine equalities and inequalities, hence a convex compact set. Since $g_l(\cdot)$ is continuous and convex, we can easily verify that the objective of (1) is quasi-convex on $R = \{r_l^{ij}\}$. CP's strategy space is defined by the constraint set of (3), which is also convex and compact. Similarly, if $h_l(f_l^{cp})$ is continuous and convex, the objective of (3) is quasi-convex on X_{cp}. In particular, consider the special case in which CP minimizes latency (2). When CP solves SS with e2e info, $h_l(f_l) = \int_0^{f_l} D_l(u)du$. When CP is solves SS with improved visibility, $h_l(f_l^{cp}) = f_l^{cp} D_l(f_l)$. In both cases, if $D_l(\cdot)$ is continuous, non-decreasing, and convex, so is $h_l(\cdot)$. One can again verify the quasi-convexity of the objective in (3). ■

The existence of a Nash equilibrium does not guarantee that the trajectory (4) leads to one. In Section 7 we demonstrate the convergence of iterative player optimization in simulation. In general, the Nash equilibrium may not be unique, in terms of both decision variables and objective values. Next, we discuss a special case where the Nash equilibrium is unique and can be attained by alternating player moves (4).

4.2 Global Optimality under Same Objective and Absence of Background Traffic

In the following, we consider a special case of the TE-SS game, in which the ISP and the CP optimize the *same* objective function, i.e., $g_l(\cdot) = h_l(\cdot)$, so

$$TE = SS = \sum_l \Phi_l(f_l), \qquad (5)$$

when there is *no background traffic*. One example is when the network carries *only* the CP traffic, and both the ISP and the CP aim to minimize the average traffic latency, i.e., $\Phi_l(f_l) = f_l \cdot D_l(f_l)$. An interesting question that naturally arises is whether the two players' alternating best response to each other's decision can lead to a socially optimal point.

Define a notion of *global optimum*, which is the optimal point to the following optimization problem.

TE-SS-Special($\hat{X}_{cp}$):

$$\begin{aligned} \text{minimize} \quad & \sum_l \Phi_l(f_l) \qquad (6) \\ \text{subject to} \quad & f_l = \sum_{(s,t)} x_l^{st} \le C_l,\ \forall l \\ & \sum_{s \in S} \left(\sum_{l:l \in \text{In}(v)} x_l^{st} - \sum_{l:l \in \text{Out}(v)} x_l^{st} \right) = M_t \cdot I_{v=t},\ \forall v \notin S,\ \forall t \in T \\ \text{variables} \quad & x_l^{st} \ge 0,\ \forall (s,t), \forall l \end{aligned}$$

where x_l^{st} denotes the traffic rate for flow (s,t) delivered on link l. The variable x_l^{st} allows a global coordinator to route a user's demand from any server in any way it wants, thus problem (6) establishes an upper-bound on how well one can do to minimize the traffic latency. Note that x_l^{st} captures both R and X_{cp}, which offers more degrees of freedom for a joint routing and server-selection

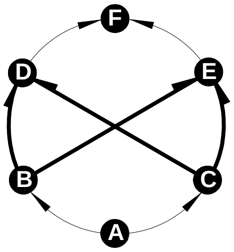


Figure 2: An Example of the Paradox of Extra Information

problem. Its mathematical properties will be further discussed in Section 5.2.

The special case TE-SS game (5) has a Nash equilibrium, as shown in Theorem 1. Nash equilibrium may not be unique in general. This is because when there is no traffic between a server-user pair, the TE routing decision for this pair can be arbitrary without affecting its utility. In the worst case, a Nash equilibrium can be arbitrarily suboptimal to the global optimum. Now assume that there exists non-zero traffic demand between any server-user pair as in [8]. Then the alternating player moves in (4) reach a *unique* Nash equilibrium, which is also global optimum to (6). TE-SS interaction does not sacrifice any efficiency in this special case, and the optimal operating point can be achieved by alternating best response unilaterally, without the need of a global coordination. This result is shown in [8] where the idea is to prove the equivalence of Nash equilibrium and the global optimum. An alternative proof is to first transform problem (6) and consider alternating projections of variables onto a convex set [18].

The special case analysis establishes a lower bound on the efficiency loss of TE-SS interaction. In general, there are three sources of mis-alignment between the optimization problems of TE and SS: (1) different shapes of the cost functions, (2) different types of delay modeled by the cost functions, and (3) existence of background traffic in the TE problem. The above special case illustrates what might happen if differences (1) and (3) are avoided, while the evaluation results in Section 7 highlight the impact of difference (2). As we will show next, difference in objective functions and presence of background traffic can lead to significant efficiency loss.

5. EFFICIENCY LOSS

We next study the efficiency loss in the general case of the TE-SS game, which may be caused by incomplete information, or unilateral actions that miss the opportunity to achieve a jointly attainable optimal point. We present two case studies that illustrate these two sources of suboptimal performance. We first present a toy network and show that under certain conditions the CP performs even worse in Model II than Model I, despite having more information about underlying network conditions. We next propose the notion of Pareto-optimality as the performance benchmark, and quantify the efficiency loss in both Model I and Model II.

5.1 The Paradox of Extra Information

Consider an ISP network illustrated in Figure 2. We designate an end user node, $T = \{F\}$, and two CP servers, $S = \{B, C\}$. The end user has a content demand of $M_F = 2$. We also allow two background traffic flows, $A \to D$ and $A \to E$, each of which has one unit of traffic demand. Edge directions are noted on the figure, so one can figure out the possible routes, i.e., there are two paths for each traffic flow (clockwise and counter-clockwise). To simplify the analysis and deliver the most essential message from this example, suppose that both TE and SS costs on the four thin links are negligible so the four bold links constitute the *bottleneck* of the network. In Table 3, we list the link capacities, ISP's cost function $g_l(\cdot)$, and link latency function $D_l(\cdot)$. Suppose the CP aims to minimize the average latency of its traffic. We compare the Nash equilibrium of two situations when the CP optimizes its network by SS with e2e info and SS with improved visibility.

link	$l_1 : BD$	$l_2 : BE$	$l_3 : CD$	$l_4 : CE$
C_l	$1+\varepsilon$	$1+\varepsilon$	$1+\varepsilon$	$1+\varepsilon$
$D_l(f_l)$	f_1	$\frac{1}{1+\varepsilon-f_2}$	$\frac{1}{1+\varepsilon-f_3}$	f_4
$g_l(x)$	$g_1(\cdot) = g_2(\cdot) = g_3(\cdot) = g_4(\cdot)$			

Table 3: Link capacities, ISP's and CP's link cost functions in the example of Paradox of Extra Information.

The stability condition for the ISP at Nash equilibrium is $g_1'(f_1) = g_2'(f_2) = g_3'(f_3) = g_4'(f_4)$. Since the ISP's link cost functions are identical, the total traffic on each link must be identical. On the other hand, the stability condition for the CP at Nash equilibrium is that (B, F) and (C, F) have the same marginal latency. Based on the observations, we can derive two Nash equilibrium points.

When the CP takes the strategy of SS with e2e info, let

$$\text{Model I:} \begin{cases} X_{CP} : \left\{ x_{BF} = 1,\ x_{CF} = 1 \right\} \\ R : \left\{ r_1^{BF} = 1-\alpha,\ r_2^{BF} = \alpha,\ r_3^{CF} = \alpha,\ r_4^{CF} = 1-\alpha, \right. \\ \qquad \left. r_1^{AD} = \alpha,\ r_3^{AD} = 1-\alpha,\ r_2^{AE} = 1-\alpha,\ r_4^{AE} = \alpha \right\} \end{cases}$$

One can check that this is indeed a Nash equilibrium solution, where $f_1 = f_2 = f_3 = f_4 = 1$, and $D_{BF} = D_{CF} = 1 - \alpha + \alpha/\varepsilon$. The CP's objective $SS_{\mathrm{I}} = 2(1 - \alpha + \alpha/\varepsilon)$.

When the CP takes the strategy of SS with improved visibility, let

$$\text{Model II:} \begin{cases} X_{CP} : \left\{ x_{BF} = 1,\ x_{CF} = 1 \right\} \\ R : \left\{ r_1^{BF} = \alpha,\ r_2^{BF} = 1-\alpha,\ r_3^{CF} = 1-\alpha,\ r_4^{CF} = \alpha, \right. \\ \qquad \left. r_1^{AD} = 1-\alpha,\ r_3^{AD} = \alpha,\ r_2^{AE} = \alpha,\ r_4^{AE} = 1-\alpha \right\} \end{cases}$$

This is a Nash equilibrium point, where $f_1 = f_2 = f_3 = f_4 = 1$, and $d_{BF} = d_{CF} = \alpha(1+\alpha) + (1-\alpha)(1/\varepsilon + (1-\alpha)/\varepsilon^2)$. The CP's objective $SS_{\mathrm{II}} = 2(\alpha + (1-\alpha)/\varepsilon)$.

When $0 < \varepsilon < 1, 0 \leq \alpha < 1/2$, we have the counter-intuitive $SS_{\mathrm{I}} < SS_{\mathrm{II}}$: more information may hurt the CP's performance. In the worst case,

$$\lim_{\alpha \to 0, \varepsilon \to 0} \frac{SS_{\mathrm{II}}}{SS_{\mathrm{I}}} = \infty$$

i.e., the performance degradation can be unbounded.

This is not surprising, since the Nash equilibrium is generally non-unique, both in terms of equilibrium solutions and equilibrium objectives. When ISP and CP's objectives are mis-aligned, the ISP's decision may route CP's traffic on bad paths from the CP's perspective. In this example, the paradox happens when the ISP *privileges* the CP traffic in Model I (although SS relies on e2e info only) by assigning it to good paths, and when the ISP misroutes the CP traffic to bad paths in Model II (although SS gains improved visibility). In practice, such a scenario is likely to happen, since the ISP cares about link congestion (link utilization), while the CP cares about latency, which depends not only on link load, but also on propagation delay. Thus ISP and CP's partial collaboration by only passing information is not sufficient to achieve global optimality.

5.2 Pareto Optimality and Illustration of Sub-Optimality

As in the above example, one of the causes of sub-optimality is that TE and SS's objectives are not necessarily aligned. To measure efficiency in a system with multiple objectives, a common approach is to explore the *Pareto curve*. For points on the Pareto curve, we cannot improve one objective further without hurting the other. The Pareto curve characterizes the tradeoff of potentially conflicting goals of different parties. One way to trace the tradeoff curve is to optimize a weighted sum of the objectives:

$$\begin{aligned} &\text{minimize} \quad TE+\gamma\cdot SS \\ &\text{variables} \quad R\in\mathscr{R},\ X_{cp}\in\mathscr{X}_{cp} \end{aligned} \tag{7}$$

where $\gamma\geq 0$ is a scalar representing the relative weight of the two objectives. $\mathscr{R}$ and $\mathscr{X}_{cp}$ are the feasible regions defined by the constraints in (1) and (3):

$$\mathscr{R}\times\mathscr{X}_{cp}=\Big\{r_l^{ij},x_{st}\mid 0\leq r_l^{ij}\leq 1,\, x_{st}\geq 0, \\ \sum_{l:l\in\mathrm{In}(v)} r_l^{ij}-\sum_{l:l\in\mathrm{Out}(v)} r_l^{ij}=I_{v=j},\forall v\in V\backslash\{i\}, \\ f_l=\sum_{(i,j)} x_{ij}\cdot r_l^{ij}\leq C_l,\ \sum_{s\in S} x_{st}=M_t\Big\}$$

The formulation of problem (7) is not easy to solve. In fact, the objective of (7) is no longer convex in variables $\{r_l^{st},x_{st}\}$, and the feasible region defined by constraints of (7) is not convex. One way to overcome this problem is to consider a relaxed decision space that is a superset of the original solution space. Instead of restricting each player to its own operating domain, i.e., ISP controls routing and CP controls server selection, we introduce a joint routing and content delivery problem. Let x_l^{st} denote the *rate* of traffic carried on link l that belongs to flow (s,t). Such a convexification of the original problem (7) gives more freedom to joint TE and SS problem. Denote the generalized CP decision variable as $\hat{X}_{cp}=\{x_l^{st}\}_{s\in S,t\in T}$, and $R_{bg}=\{r_l^{ij}\}_{(i,j)\notin S\times T}$ as background routing matrix. Consider the following optimization problem:

TE-SS-weighted($\hat{X}_{cp},R_{bg}$)

$$\begin{aligned} &\text{minimize} \quad TE+\gamma\cdot SS \\ &\text{subject to} \quad f_l^{cp}=\sum_{(s,t)} x_l^{st},\ \forall l \\ &\qquad f_l=f_l^{cp}+\sum_{(i,j)\notin S\times T} x_{ij}\cdot r_l^{ij}\leq C_l,\ \forall l \\ &\qquad \sum_{l:l\in\mathrm{In}(v)} r_l^{ij}-\sum_{l:l\in\mathrm{Out}(v)} r_l^{ij}=I_{v=j},\ \forall(i,j)\notin S\times T,\forall v\in V\backslash\{i\} \\ &\qquad \sum_{s\in S}\left(\sum_{l:l\in\mathrm{In}(v)} x_l^{st}-\sum_{l:l\in\mathrm{Out}(v)} x_l^{st}\right)=M_t\cdot I_{v=t},\ \forall v\notin S,\ \forall t\in T \\ &\text{variables} \quad x_l^{st}\geq 0,\ 0\leq r_l^{ij}\leq 1 \end{aligned} \tag{8}$$

Denote the feasible space of the joint variable as $\mathscr{A}=\{\hat{X}_{cp},R_{bg}\}$.

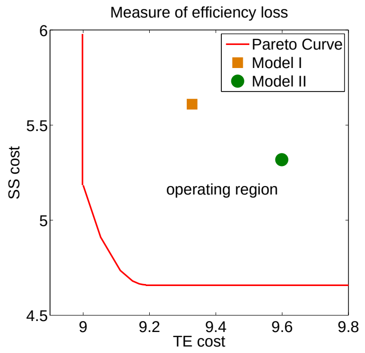


Figure 3: A numerical example illustrating sub-optimality.

If we vary γ and plot the achieved TE objectives versus SS objectives, we obtain the Pareto curve.

To illustrate the Pareto curve and efficiency loss in Model I and Model II, we plot in Figure 3 the Pareto curve and the Nash equilibria in the two-dimensional objective space (TE,SS) for the network shown in Figure 2. The simulation shows that when the CP leverages the complete information to optimize (3), it is able to achieve lower delay, but the TE cost suffers. Though it is not clear which operating point is better, both equilibria are away from the Pareto curve, which shows that there is room for performance improvement in both dimensions.

6. A JOINT DESIGN: MODEL III

Motivated by the need for a joint TE and SS design, we propose the Nash bargaining solution to reduce the efficiency loss observed above. Using the theory of optimization decomposition, we derive a distributed algorithm by which the ISP and the CP can act separately and communicate with a limited amount of information exchange.

6.1 Motivation

An ISP providing content distribution service in its own network has control over both routing and server selection. So the ISP can consider the characteristics of both types of traffic (background and CP) and jointly optimize a carefully chosen objective. The jointly optimized system should meet at least two goals: (i) optimality, i.e., it should achieve Pareto optimality so the network resources are efficiently utilized, and (ii) fairness, i.e., the tradeoff between two non-synergistic objectives should be balanced so both parties benefit from the cooperation.

One natural design choice is to optimize the weighted sum of the traffic engineering goal and server selection goal as shown in (8). However, solving (8) for each γ and adaptively tuning γ in a *trial-and-error* fashion is impractical and inefficient. First, it is not straightforward to weigh the tradeoff between the two objectives. Second, one needs to compute an appropriate weight parameter γ for every combination of background load and CP traffic demand. In addition, the offline computation does not adapt to dynamic changes of network conditions, such as cross traffic or link failures. Last, tuning γ to explore a broad region of system operating points is computationally expensive.

Besides the system considerations above, the economic perspective requires a *fair* solution. Namely, the joint design should benefit both TE and SS. In addition, such a model also applies to a more general case when the ISP and the CP are different business entities. They cooperate only when the cooperation leads to a win-win situation, and the "division" of the benefits should be fair, i.e., one who makes greater contribution to the collaboration should be able to receive more reward, even when their goals are conflicting.

While the joint system is designed from a clean state, it should accept an incremental deployment from the existing infrastructure. In particular, we prefer that the functionalities of routing and server selection be separated, with minor changes to each component. The modularized design allows us to manage each optimization independently, with a judicious amount of information exchange. Designing for scalability and modularity is beneficial to both the ISP and the CP, and allows their cooperation either as a single entity or as different ones.

Based on all the above considerations, we apply the concept of *Nash bargaining solution* [9, 20] from cooperative game theory. It ensures that the joint system achieves an *efficient* and *fair* operating point. The solution structure also allows a modular implementation.

6.2 Nash Bargaining Solution

Consider a Nash bargaining solution which solves the following optimization problem:

$$\begin{array}{ll} \text{maximize} & (TE_0 - TE)(SS_0 - SS) \\ \text{variables} & \{\hat{X}_{cp}, R_{bg}\} \in \mathscr{A} \end{array} \tag{9}$$

where (TE_0, SS_0) is a constant called the *disagreement point*, which represents the starting point of their negotiation. Namely, (TE_0, SS_0) is the status-quo we observe before any cooperation. For instance, one can view the Nash equilibrium in Model I as a disagreement point, since it is the operating point the system would reach without any further optimization. By optimizing the product of performance improvements of TE and SS, the Nash bargaining solution guarantees the joint system is optimal and fair. A Nash bargaining solution is defined by the following axioms, and is the only solution that satisfies all of four axioms [9, 20]:

- *Pareto optimality*. A Pareto optimal solution ensures efficiency.

- *Symmetry*. The two players should get equal share of the gains through cooperation, if the two players' problems are symmetric, i.e., they have the same cost functions, and have the same objective value at the disagreement point.

- *Expected utility axiom*. The Nash bargaining solution is invariant under affine transformations. Intuitively, this axiom suggests that the Nash bargaining solution is insensitive to different units used in the objective and can be efficiently computed by affine projection.

- *Independence of irrelevant alternatives*. This means that adding extra constraints in the feasible operating region does not change the solution, as long as the solution itself is feasible.

The choice of the disagreement point is subject to different economic considerations. For a single network provider who wishes to provide both services, it can optimize the product of improvement ratio by setting the disagreement point to be the origin, i.e., equivalent to $TE \cdot SS/(TE_0 \cdot SS_0)$. For two separate ISP and CP entities who wish to cooperate, the Nash equilibrium of Model I may be a natural choice, since it represents the benchmark performance of current practice, which is the baseline for any future cooperation. It can be obtained from the empirical observations of their average performance. Alternatively, they can choose their preferred performance level as the disagreement point, written into the contract. In this work, we use the Nash equilibrium of Model I as the disagreement point to compare the performances of our three models.

6.3 COST Algorithm

In this section, we show how Nash bargaining solution can be implemented in a modularized manner, i.e., keeping SS and TE functionalities separate. This is important because modularized design increases the re-usability of legacy systems with minor changes, like existing CDNs deployment. In terms of cooperation between two independent financial entities, the modularized structure presents the possibility of cooperation without revealing confidential internal information to each other.

We next develop COST (COoperative Server selection and Traffic engineering), a protocol that implements NBS by separate TE and SS optimizations and communication between them. We apply the theory of optimization decomposition [10] to decompose problem (9) into subproblems. ISP solves a new routing problem, which controls the routing of background traffic only. The CP solves a new server selection problem, given the network topology information. The ISP also passes the routing control of content traffic to the CP, offering more freedom to how content can be delivered on the network. They communicate via underlying *link prices*, which are computed locally using traffic levels on each link.

Consider the objective of (9), which can be converted to

$$\text{maximize} \quad \log(TE_0 - TE) + \log(SS_0 - SS)$$

since the log function is monotonic and the feasible solution space is unaffected. The introduction of the log functions help reveal the decomposition structure of the original problem. Two auxiliary variable $\overline{f_l^{cp}}$ and $\overline{f_l^{bg}}$ are introduced to reflect the preferred CP traffic level from the ISP's perspective and the preferred background traffic level from the CP's perspective. (9) can be rewritten as

$$\text{max.} \quad \log(TE_0 - \sum_l g_l(f_l^{bg} + \overline{f_l^{cp}})) + \log(SS_0 - \sum_l h_l(f_l^{cp} + \overline{f_l^{bg}})) \tag{10}$$

$$\begin{aligned} \text{s.t.} \quad & f_l^{cp} = \sum_{(s,t)} x_l^{st}, \; f_l^{bg} = \sum_{(i,j)\notin S\times T} x_{ij} \cdot r_l^{ij}, \; \forall l \\ & \overline{f_l^{cp}} = f_l^{cp}, \; \overline{f_l^{bg}} = f_l^{bg}, \; f_l^{cp} + f_l^{bg} \le C_l, \; \forall l \\ & \sum_{l:l\in \mathrm{In}(v)} r_l^{ij} - \sum_{l:l\in \mathrm{Out}(v)} r_l^{ij} = I_{v=j}, \; \forall (i,j) \notin S \times T, \forall v \in V \backslash \{i\} \\ & \sum_{s\in S} \left(\sum_{l:l\in \mathrm{In}(v)} x_l^{st} - \sum_{l:l\in \mathrm{Out}(v)} x_l^{st} \right) = M_t \cdot I_{v=t}, \; \forall v \notin S, \; \forall t \in T \\ \text{var.} \quad & x_l^{st} \ge 0, \; 0 \le r_l^{ij} \le 1, \; \forall (i,j) \notin S \times T, \; \overline{f_l^{cp}}, \; \overline{f_l^{bg}} \end{aligned}$$

The consistency constraint on the auxiliary variable and the original variable ensures that the solution equivalent to problem (9). We take a partial Lagrangian of (10) as

$$\begin{aligned} & L(x_l^{st}, r_l^{ij}, \overline{f_l^{cp}}, \overline{f_l^{bg}}, \lambda_l, \mu_l, \nu_l) \\ & = \log(TE_0 - \sum_l g_l(f_l^{bg} + \overline{f_l^{cp}})) + \sum_l \mu_l(f_l^{bg} - \overline{f_l^{bg}}) \\ & + \log(SS_0 - \sum_l h_l(f_l^{cp} + \overline{f_l^{bg}})) + \sum_l \nu_l(f_l^{cp} - \overline{f_l^{cp}}) \\ & + \sum_l \lambda_l(C_l - f_l^{bg} - f_l^{cp}) \end{aligned}$$

λ_l is the *link price*, which reflects the cost of overshooting the link capacity, and μ_l, ν_l are the *consistency prices*, which reflect the cost of disagreement between ISP and CP on the preferred link resource allocation. Observe that f_l^{cp} and f_l^{bg} can be separated in the Lagrangian function. We take a dual decomposition approach, and (10) is decomposed into two subproblems:

SS-NBS($x_l^{st}, \overline{f_l^{bg}}$)**:**

$$\text{max.} \quad \log(SS_0 - \sum_l h_l(f_l^{cp} + \overline{f_l^{bg}})) + \sum_l (\nu_l f_l^{cp} - \mu_l \overline{f_l^{bg}} - \lambda_l f_l^{cp}) \tag{11}$$

$$\text{s.t.} \quad f_l^{cp} = \sum_{(s,t)} x_l^{st}, \ \forall l$$

$$\sum_{s \in S} \left(\sum_{l:l \in \text{In}(v)} x_l^{st} - \sum_{l:l \in \text{Out}(v)} x_l^{st} \right) = M_t \cdot I_{v=t}, \ \forall v \notin S, \ \forall t \in T$$

$$\text{var.} \quad x_l^{st} \geq 0, \forall (s,t) \in S \times T, \ \overline{f_l^{bg}}$$

and

TE-NBS($r_l^{ij}, \overline{f_l^{cp}}$)**:**

$$\text{max.} \quad \log(TE_0 - \sum_l g_l(f_l^{bg} + \overline{f_l^{cp}})) + \sum_l (\mu_l f_l^{bg} - \nu_l \overline{f_l^{cp}} - \lambda_l f_l^{bg}) \tag{12}$$

$$\text{s.t.} \quad f_l^{bg} = \sum_{(i,j) \notin S \times T} x_{ij} \cdot r_l^{ij}, \ \forall l$$

$$\sum_{l:l \in \text{In}(v)} r_l^{ij} - \sum_{l:l \in \text{Out}(v)} r_l^{ij} = I_{v=j}, \ \forall (i,j) \notin S \times T, \forall v \in V \backslash \{i\}$$

$$\text{var.} \quad 0 \leq r_l^{ij} \leq 1, \forall (i,j) \notin S \times T, \ \overline{f_l^{cp}}$$

The optimal solutions of (11) and (12) for a given set of prices μ_l, ν_l, and λ_l define the dual function $\text{Dual}(\mu_l, \nu_l, \lambda_l)$. The dual problem is given as:

$$\begin{aligned} \text{minimize} \quad & \text{Dual}(\mu_l, \nu_l, \lambda_l) \\ \text{variable} \quad & \lambda_l \geq 0, \mu_l, \nu_l \end{aligned} \tag{13}$$

We can solve the dual problem with the following price updates:

$$\lambda_l(t+1) = \left[\lambda_l(t) - \beta_{\lambda l} \left(C_l - f_l^{bg} - f_l^{cp} \right) \right]^+, \ \forall l \tag{14}$$

$$\mu_l(t+1) = \mu_l(t) - \beta_{\mu l}(f_l^{bg} - \overline{f_l^{bg}}), \ \forall l \tag{15}$$

$$\nu_l(t+1) = \nu_l(t) - \beta_{\nu l}(f_l^{cp} - \overline{f_l^{cp}}), \ \forall l \tag{16}$$

where β's are diminishing step sizes or small constant step sizes often used in practice [21]. Table 4 presents the COST algorithm that implements the Nash bargaining solution distributively.

In COST, the ISP solves the new version TE, i.e., TE-NBS, and the CP solves the new version SS, i.e., SS-NBS. In terms of information sharing, the CP learns the network topology from the ISP. They do not directly exchange information with each other. Instead, they report $\overline{f_l^{cp}}$ and $\overline{f_l^{bg}}$ information to underlying links, which pass the computed price information back to TE and SS. It is possible to further implement TE or SS in a distributed manner, such as on the user/server levels.

There are two main challenges on practical implementation of COST. First, TE needs to adapt quickly to network dynamics. Fast timescale TE has recently been proposed in various works. Second, an extra price update component is required on each link, which involves price computation and message passing between TE and SS. This functionality can be potentially implemented in routers.

	COST (COoperative SS and TE)
	ISP: TE algorithm
(i)	Receives link price λ_l and consistency price μ_l, ν_l from physical links $l \in E$
(ii)	ISP solves (12) and computes R_{bg} for background traffic
(iii)	ISP passes f_l^{bg}, $\overline{f_l^{cp}}$ information to each link l
(iv)	Go back to (i)
	CP: SS algorithm
(i)	Receives link price λ_l and consistency price μ_l, ν_l from physical links $l \in E$
(ii)	CP solves (11) and computes X_{cp} for content traffic.
(iii)	CP passes f_l^{cp}, $\overline{f_l^{bg}}$ information to each link l
(iv)	Go back to (i)
	Link: price update algorithm
(i)	Initialization step: set $\lambda_l \geq 0$, and μ_l, ν_l arbitrarily
(ii)	Updates link price λ_l according to (14)
(iii)	Updates consistency prices μ_l, ν_l according to (15)(16)
(iv)	Passes λ_l, μ_l, ν_l information to TE and SS
(v)	Go back to (ii)

Table 4: Distributed algorithm for solving problem (9).

The COST algorithm is precisely captured by the decomposition method described above. Certain choice of step sizes, such as $\beta(t) = \beta_0/t$, where $\beta_0 > 0$, guarantees that the algorithm converges to a global optimum [22].

Theorem 2. *The distributed algorithm COST converges to the optimum of (9) for sufficiently small step sizes $\beta_{\lambda l}, \beta_{\mu l}$ and $\beta_{\nu l}$.*

7. PERFORMANCE EVALUATION

In this section, we use simulations to demonstrate the efficiency loss that may occur for real network topologies and traffic models. We also compare the performance of the three models. We solve the Nash bargaining solution centrally, without using the COST algorithm, since we are primarily interested in its performance. Complementary to the theoretical analysis, the simulation results allow us to gain a better understanding of the efficiency loss under realistic network environments. These simulation results also provide guidance to network operators who need to decide which approach to take, sharing information or sharing control.

7.1 Simulation Setup

We evaluate our models under ISP topologies obtained from Rocketfuel [23]. We use the backbone topology of the research network Abilene [24] and several major tier-1 ISPs in north America. The choice of these topologies also reflects different geometric properties of the graph. For instance, Abilene is the simplest graph with two bottleneck paths horizontally. The backbones of AT&T and Exodus have a hub-and-spoke structure with some shortcuts between nodes pairs. The topology of Level 3 is almost a complete mesh, while Sprint is in between these two kinds. We simulate the traffic demand using a gravity model [25], which reflects the pairwise communication pattern on the Internet. The content demand of a CP user is assumed to be proportional to the node population.

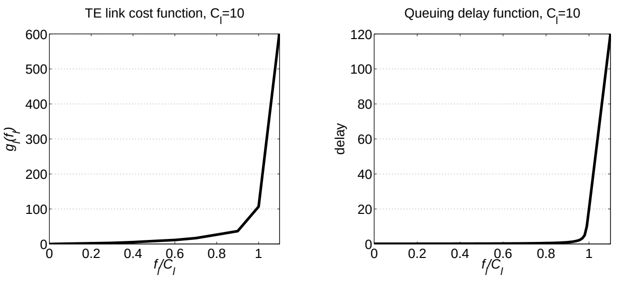
The TE cost function $g(\cdot)$ and the SS cost function $h(\cdot)$ are chosen as follows. ISPs usually model congestion cost with a convex increasing function of the link load. The exact shape of the function $g_l(f_l)$ is not important, and we use the same piecewise linear cost function as in [11], given below:

$$g_l(f_l, C_l) = \begin{cases} f_l & 0 \le f_l/C_l < 1/3 \\ 3f_l - 2/3C_l & 1/3 \le f_l/C_l < 2/3 \\ 10f_l - 16/3C_l & 2/3 \le f_l/C_l < 9/10 \\ 70f_l - 178/3C_l & 9/10 \le f_l/C_l < 1 \\ 500f_l - 1468/3C_l & 1 \le f_l/C_l < 11/10 \\ 5000f_l - 16318/3C_l & 11/10 \le f_l/C_l < \infty \end{cases}$$

The CP's cost function can be the performance cost like latency, financial cost charged by ISPs. We consider the case where latency is the primary performance metric, i.e., the content traffic is delay sensitive like video conferencing or live streaming. So we let the CP's cost function $h_l(\cdot)$ be of the form given by (2), i.e., $h_l(f_l) = f_l^{cp} \cdot D_l(f_l)$. A link's latency $D_l(\cdot)$ consists of queuing delay and propagation delay. The propagation delay is proportional to the geographical distances between nodes. The queuing delay is approximated by the M/M/1 model, i.e.,

$$D_{queue} = \frac{1}{C_l - f_l}, \quad f_l < C_l$$

with a linear approximation when the link utilization is over 99%. We relax the link capacity constraints in both TE and SS and penalize traffic overshooting the link capacity with high costs. The shapes of the TE link cost function and queuing delay function are illustrated in Figure 4. We intensionally choose the cost functions of TE and SS to be similar in shape. This allows us to quantify the efficiency loss of Model I and Model II even when their objectives are relatively well aligned, as well as the improvement brought by Model III.



(a) TE link cost

(b) Link queuing delay

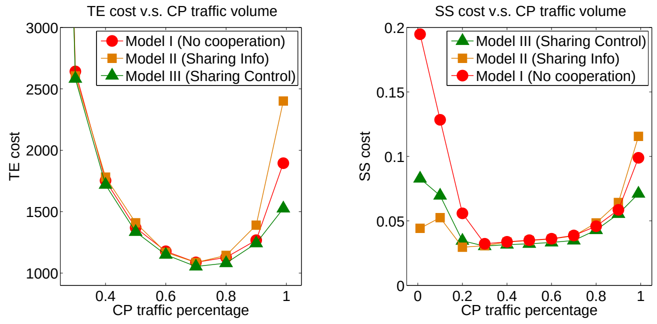
Figure 4: ISP and CP cost functions.

7.2 Evaluation Results

7.2.1 Tussle between background and CP's traffic

We first demonstrate how CP's traffic intensity affects the overall network performance. We fix the total amount of traffic and tune the ratio between background traffic and CP's traffic. We evaluate the performance of different models when CP traffic grows from 1% to 100% of the total traffic. Figure 5 illustrates the results on Abilene topology.

The general trend of both TE and SS objectives for all three models is that the cost first decreases as CP traffic percentage grows, and later increases as CP's traffic dominates the network. The decreasing trend is due to the fact that CP's traffic is self-optimized by selecting servers close to a user, thus offloading the network. The increasing trend is more interesting, suggesting that when a higher percentage of total traffic is CP-generated, the negative effect of TE-SS interaction is amplified, even when the ISP and the CP share similar cost functions. Low link congestion usually means low end-to-end latency, and vice versa. However, they differ in the following: (i) TE might penalize high utilization before queueing delay becomes significant in order to leave as much room as possible to accommodate changes in traffic, and (ii) CP considers both propagation delay and queueing delay so it may choose a moderately-congested short path over a lightly-loaded long path. This explains why the optimization efforts of two players are at odds.



(a)

(b)

Figure 5: The TE-SS tussle v.s. CP's traffic intensity (Abilene topology)

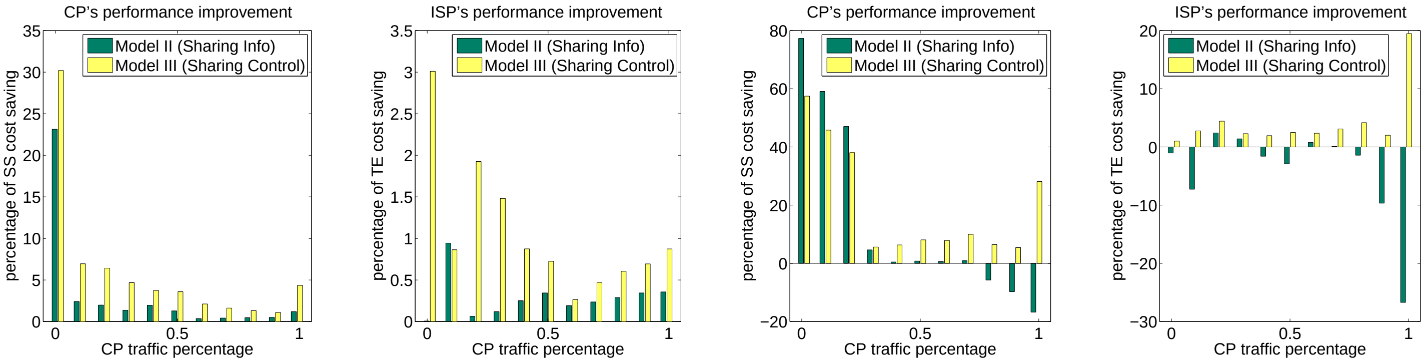
7.2.2 Network congestion v.s. performance improvement

We now study the network conditions under which more performance improvement is possible. We evaluate the three models on the Abilene topology. Again, we fix the total amount of traffic and vary the CP's traffic percentage. Now we change link capacities and evaluate two scenarios: when the network is moderately congested and when the network is highly congested. We show the performance improvement of Model II and Model III over Model I (in percentages) and plot the results in Figure 6. Figures 6(a-b) show the improvement of the ISP and the CP when the network is under low load. Generally, Model II and Model III improve both TE and SS, and Model III outperforms Model II in most cases, with the exception that Model II is biased towards SS sometimes. However, both ISP and CP's improvement are not substantial (note the different scales of y-axes), except when CP traffic is insignificant (1%). This is because when the network is under low load, the slopes of TE and SS cost functions are "flat," thus leaving little space for improvement.

Figure 6(c-d) show the results when the network is under high load. Improvement becomes more significant, especially at the two extremes: when CP's traffic is insignificant or prevalent. This again suggests that when CP traffic is dominant, there is a large space left for improvement even when two objectives are similar in shape. However, observe that while model III always improves TE and SS, Model II could sometimes perform worse than Model I.

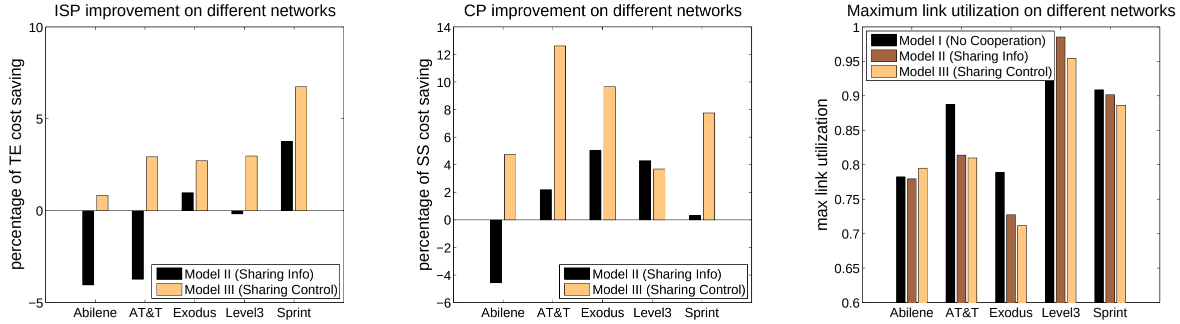
7.2.3 Impact of ISP topologies

We evaluate our three models on different ISP topologies. The topological properties of different graphs are discussed earlier. The CP's traffic is 80% of the total traffic and link capacities are set such that networks are under high traffic load. Our findings are depicted in Figure 7. Note that performance improvement is rela-



(a) (b) (c) (d)

Figure 6: TE and SS performance improvement of Model II and III over Model I. (a-b) Abilene network under low traffic load: moderate improvement; (c-d) Abilene network under high traffic load: more significant improvement, but more information (in Model II) does not necessarily benefit the CP and the ISP (the paradox of extra information).



(a) (b) (c)

Figure 7: Performance evaluation over different ISP topologies. Abilene: small cut graph; AT&T, Exodus: hub-and-spoke with shortcuts; Level 3: complete mesh; Sprint: in between.

tively more significant in more complex graphs. Simple topologies with small min-cut sizes are networks where the apparent paradox of more (incomplete) information is likely to happen. Besides the TE and SS objectives, we also plot the maximum link utilization to illustrate the level of congestion in the network. Higher network load shows more space for potential improvement. Also, model III improves this metric generally, which might be another important consideration for network providers.

8. RELATED WORK

This paper is an extension of our earlier workshop paper [26]. Additions in this paper include the following: a more general CP model, analysis of optimality conditions in three cooperation models, paradox of extra information, implementation of Nash bargaining solution, and large scale evaluation.

The most similar work is a parallel work [8], which studied the interaction between content distribution and traffic engineering. It shows the optimality conditions for two separate problems to converge to a socially optimal point, as discussed in Section 4.2. It also provides a theoretical bound on efficiency loss and discusses generalizations to multiple ISPs and overlay networks.

Some earlier work studied the self-interaction within ISPs or CPs themselves. In [16], the authors used simulation to show that selfish routing is close to optimal in Internet-like environments without sacrificing much performance degradation. [27] studied the problem of load balancing by overlay routing, and how to alleviate race conditions among multiple co-existing overlays. [28] studied the resource allocation problem at inter-AS level where ISPs compete to maximize their revenues. [29] applied Nash bargaining solution to solve an inter-domain ISP peering problem.

The need for cooperation between content providers and network providers is raising much discussion in both the research community and industry. [30] used price theory to reconcile the tussle between peer-assisted content distribution and ISP's resource management. [6] proposed a communication portal between ISPs and P2P applications, which P2P applications can consult for ISP-biased network information to reduce network providers' cost without sacrificing their performances. [5] proposed an oracle service run by the ISP, so P2P users can query for the ranked neighbor list according to certain performance metrics. [7] utilized existing network views collected from content distribution networks to drive biased peer selection in BitTorrent, so cross-ISP traffic can be significantly reduced and download-rate improved.

[31] studied the interaction between underlay routing and overlay routing, which can be thought of as a generalization of server selection. The authors studied the equilibrium behaviors when two

	CP no change	**CP change**
ISP no change	current practice	partial collaboration
ISP change	partial collaboration	joint system design

Table 5: To cooperate or not: possible strategies for content provider (CP) and network provider (ISP)

problems have conflicting goals. Our work explores when and why sub-optimality appears, and proposes a cooperative solution to address these issues. [32] studied the economic aspects of traditional transit providers and content providers, and applied cooperative game theory to derive an optimal settlement between these entities.

9. CONCLUSION AND FUTURE WORK

We examine the interplay between traffic engineering and content distribution. While the problem has long existed, the dramatically increased amount of content-centric traffic, e.g., CDN and P2P traffic, makes it more significant. With the strong motivation for ISPs to provide content services, they are faced with the question of whether to stay with the current design or to start sharing information or control. This work sheds light on ways ISPs and CPs can cooperate.

This paper serves as a starting point to better understand the interaction between those that operate networks and those that distribute content. Traditionally, ISPs provide and operate the pipes, while content providers distribute content over the pipes. In terms of what information can be shared between ISPs and CPs and what control can be jointly performed, there are four general categories as summarized in Table 5. The top left corner is the current practice, which may give an undesirable Nash equilibrium. The bottom right corner is the joint design, which achieves optimal operation points. The top right corner is the case where the CP receives extra information and adapts control accordingly, and the bottom left corner is the case of content-aware networking. This paper studies three of the four corners in the table. Starting from the current practice, to move towards the bottom right corner of the table, while the two parties remain separate business entities, requires unilaterally-actionable, backward-compatible, and incrementally-deployable migration paths yet to be discovered.

10. ACKNOWLEDGMENTS

We are grateful for Dan Rubenstein's shepherding the paper and for Ramesh Johari's sharing the preprint [6] that has the results cited in Section 4.2. This work is in part supported by NSF grants CNS-0519880 and CNS-0720570.

11. REFERENCES

[1] W. B. Norton, "Video Internet: The Next Wave of Massive Disruption to the U.S. Peering Ecosystem," Sept 2006. Eqinix white paper.

[2] AT&T, "U-verse." http://uverse.att.com/.

[3] Verizon, "FiOS." http://www.Verizon.com/fios/.

[4] A.-J. Su, D. R. Choffnes, A. Kuzmanovic, and F. E. Bustamante, "Drafting behind Akamai (Travelocity-based detouring)," in *Proc. ACM SIGCOMM*, 2006.

[5] V. Aggarwal, A. Feldmann, and C. Scheideler, "Can ISPs and P2P users cooperate for improved performance?," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 3, pp. 29–40, 2007.

[6] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. Liu, and A. Silberschatz, "P4P: Provider Portal for (P2P) Applications," in *Proc. ACM SIGCOMM*, 2008.

[7] D. R. Choffnes and F. E. Bustamante, "Taming the torrent: a practical approach to reducing cross-ISP traffic in peer-to-peer systems," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4, pp. 363–374, 2008.

[8] D. DiPalantino and R. Johari, "Traffic engineering versus content distribution: A game theoretic perspective," in *Proc. IEEE INFOCOM*, 2009.

[9] J. F. Nash, "The bargaining problem," *Econometrica*, vol. 28, pp. 155–162, 1950.

[10] D. P. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE J. on Selected Areas in Communications*, vol. 24, no. 8, pp. 1439–1451, 2006.

[11] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in *Proc. IEEE INFOCOM*, pp. 519–528, 2000.

[12] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J.McManus, "RFC 2702: Requirements for Traffic Engineering Over MPLS," September 1999.

[13] D. Xu, M. Chiang, and J. Rexford, "Like-state routing with hop-by-hop forwarding can achieve optimal traffic engineering," in *Proc. IEEE INFOCOM*, 2008.

[14] J. Wardrop, "Some theoretical aspects of road traffic research," *the Institute of Civil Engineers*, vol. 1, no. 2, pp. 325–378, 1952.

[15] T. Roughgarden and Éva Tardos, "How bad is selfish routing?," *J. of the ACM*, vol. 49, no. 2, 2002.

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

[17] M. Littman and J. Boyan, "A distributed reinforcement learning scheme for network routing," Tech. Rep. CMU-CS-93-165, Robotics Institute, Carnegie Mellon University, 1993.

[18] W. Jiang, R. Zhang-Shen, J. Rexford, and M. Chiang, "Cooperative content distribution and traffic engineering in a provider network," Tech. Rep. TR-846-08, Department of Computer Science, Princeton University, 2008.

[19] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. MIT Press, 1999.

[20] K. Binmore, A. Rubinstein, and A. Wolinsky, "The Nash bargaining solution in economic modelling," *RAND Journal of Economics*, vol. 17, pp. 176–188, 1986.

[21] J. He, R. Zhang-Shen, Y. Li, C.-Y. Lee, J. Rexford, and M. Chiang, "DaVinci: Dynamically Adaptive Virtual Networks for a Customized Internet," in *Proc. CoNEXT*, Dec 2008.

[22] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[23] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP topologies with Rocketfuel," *IEEE/ACM Trans. Networking*, vol. 12, no. 1, pp. 2–16, 2004.

[24] "Abilene." http://www.internet2.edu.

[25] M. Roughan, M. Thorup, and Y. Zhang, "Performance of estimated traffic matrices in traffic engineering," *SIGMETRICS Perform. Eval. Rev.*, vol. 31, no. 1, pp. 326–327, 2003.

[26] W. Jiang, R. Zhang-Shen, J. Rexford, and M. Chiang, "Cooperative content distribution and traffic engineering," in *NetEcon '08*, August 2008.

[27] W. Jiang, D.-M. Chiu, and J. C. S. Lui, "On the interaction of multiple overlay routing," *Perform. Eval.*, vol. 62, no. 1-4, pp. 229–246, 2005.

[28] S. C. Lee, W. Jiang, D.-M. C. Chiu, and J. C. Lui, "Interaction of ISPs: Distributed resource allocation and revenue maximization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 2, pp. 204–218, 2008.

[29] G. Shrimali, A. Akella, and A. Mutapcic, "Cooperative interdomain traffic engineering using Nash bargaining and decomposition," in *Proc. IEEE INFOCOM*, 2007.

[30] M. J. Freedman, C. Aperjis, and R. Johari, "Prices are right: Managing resources and incentives in peer-assisted content distribution," in *IPTPS 08*, February 2008.

[31] Y. Liu, H. Zhang, W. Gong, and D. Towsley, "On the interaction between overlay routing and underlay routing," in *Proc. IEEE INFOCOM*, pp. 2543–2553, 2005.

[32] R. T. Ma, D. Chiu, J. C. Lui, V. Misra, and D. Rubenstein, "On cooperative settlement between content, transit and eyeball internet service providers," in *Proc. CoNEXT*, December 2008.