On the Interaction Between Overlay Routing and Traffic Engineering

Yong Liu*, Honggang Zhang*, Weibo Gong† and Don Towsley*
*Computer Science Department
University of Massachusetts, Amherst
Amherst, MA, 01003
{yongliu,honggang,towsley}@cs.umass.edu
†Electrical & Computer Engineering Department
University of Massachusetts, Amherst
Amherst, MA, 01003
gong@ecs.umass.edu

***Abstract*— In this paper, we study the interaction between overlay routing and Traffic Engineering (TE) in a single Autonomous System (AS). We formulate this interaction as a two-player non-cooperative non-zero sum game, where the overlay tries to minimize the delay of its traffic and the TE's objective is to minimize network cost. We study a Nash routing game with best-reply dynamics, in which the overlay and TE have equal status, and take turns to compute their optimal strategies based on the response of the other player in the previous round. We prove the existence, uniqueness and global stability of Nash equilibrium point (NEP) for a simple network. For general networks, we show that the selfish behavior of an overlay can cause huge cost increases and oscillations to the whole network. Even worse, we have identified cases, both analytically and experimentally, where the overlay's cost increases as the Nash routing game proceeds even though the overlay plays optimally based on TE's routing at each round. Experiments are performed to verify our analysis.**

I. Introduction

THERE are two recent trends in network routing research. One is overlay routing, and the second is Traffic Engineering (TE). Overlay routing (*e.g.*, Detour [1], RON [2]) allows end hosts to choose routes by themselves. It occurs at the application level, where traffic is routed by application level routers (computers). The logical paths and links of an overlay lie on top of physical paths set by intra-domain (*e.g.*, OSPF, MPLS, IS-IS) and inter-domain routing protocols (*e.g.*, BGP). It has been shown that these overlay routing schemes are effective in dealing with some of the deficiencies in today's IP routing structure ([1], [2], [3]). On the other hand, as pointed out by [4] and [5], Internet Service Providers (ISPs) are using traffic engineering (TE) to provide better and more robust intra-domain routing.

There is a misalignment between the objectives of overlay routing and TE routing algorithms. An overlay is interested in the optimal routes for its own group of users whereas TE is interested in improving the whole network performance by considering all users including both overlay and non-overlay (or underlay) users. It is well-known that counter-intuitive behavior can arise when individual agents conduct selfish optimizations [6]. Conflicts in their objective functions results in system performance degradation. Overlay routing is rapidly being employed by a wide variety of emerging applications. For example, Akamai runs a large overlay network for high volume content distribution. This trend in applications demands a fundamental understanding of the interaction between overlay routing and TE. In this paper, we formally study the interaction between the routing decisions made by an overlay network and MPLS Traffic Engineering within a single ISP. Our work is motivated in part by the work of Qiu *et al* [7], in which the interaction between selfish overlay routing and TE is brought up.

Overlay users view a *logical* network. Two overlay nodes are connected by a *logical link*. An overlay routing algorithm allocate overlay demands on logical links based on current logical link delays. The traffic flow on a logical link between two overlay nodes is interpreted by TE as a traffic demand between these two nodes. TE takes as input the traffic demand matrix (each demand pair includes demand from the underlay traffic and/or demand from overlay traffic), and computes a set of physical level routes using an algorithm such as that described in [4] to minimize overall network cost or to minimize maximum link utilization. Figure 1 shows conceptually how the overlay and TE interact with each other. Since both the overlay and TE optimize their

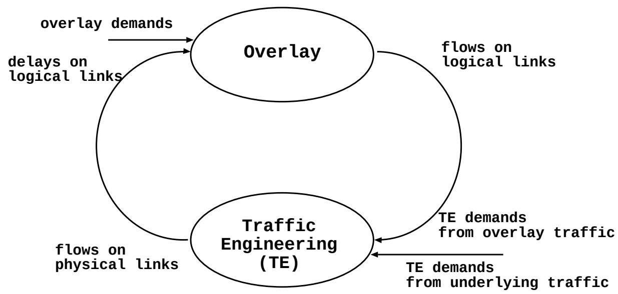


Fig. 1. Interaction between overlay optimizer and Traffic Engineering optimizer.

routes over time, the interaction between their decisions can be understood as an iterative process. Throughout this process, TE and the overlay modify each other's the input in turn. Overlay routing decisions result in the logical link traffic flows, which are in turn interpreted as traffic demands by the TE algorithm. On the other hand, TE changes the delays of logical links by adjusting the actual physical-level routes of the overlay traffic, therefore influence future routing decisions made by the overlay.

In this paper, we focus on the *dynamics of this interaction process*. We formally model this interaction as a *non-cooperative non-zero sum two player game*. The overlay and the TE algorithm are essentially two players with different optimization objectives. From now on, we refer to an overlay routing optimizer as *overlay*, and refer to a TE routing optimizer as *TE*. In the interaction process, or best-reply dynamics, each player adjusts its response optimally based on the other player's decisions during the previous round. Given the misalignment between the objectives of overlay and TE, we ask questions such as: *does a Nash equilibrium exist in this game? If a Nash equilibrium exists, is it unique? How about the stability of Nash equilibriums? Does the interaction process always converge to a Nash equilibrium? What effects on the performance of TE and overlay can be caused by this interaction process?*

The key contributions are summarized as follows:

- We formally define the optimization problems for overlay routing and TE, and formulate their interaction as a non-cooperative non-zero sum two player game. To the best of our knowledge, this is the first formal framework proposed to study the routing interaction between an overlay and a underlay network.
- Based on the framework, we thoroughly study an illustrative example. We prove the existence, uniqueness and global stability of Nash equilibrium. Mathematical analysis enables us to gain a fundamental understanding and subtle insights of the intricate interaction process. In particular, we demonstrate how the misalignment between overlay and TE's objective function triggers oscillations in their routes. We also show that the mapping from logical links to physical paths alleviates the conflict between TE and overlay, and helps them to reach equilibrium.
- For the illustrative example, we identify scenarios where the Nash equilibrium is not Pareto efficient [8]. The overlay's cost increases when it plays a best-reply Nash routing game with TE. We also observed this phenomenon in experiments with a 14-node tier-1 ISP topology. Thus, it may not be wise for an overlay to always optimize its routes each time that TE recalculates its physical routes, because in the long run, overlay's cost may increase. This observation is of practical importance to an overlay routing structure, even though it is not surprising from a game-theoretic point of view.
- Analytical results are verified by experiments on real network topologies. In those experiments, TE's costs are increased a lot in the interaction with overlay, and the cost increase of TE is a function of the percentage of overlay traffic. If overlay traffic corresponds to approximately half of the total traffic, then the overlay's influence on TE performance achieves the largest.

The rest of the paper is organized as follows. Section II describes related work. In Section III, we formally model the interaction process as a two-person non-cooperative non-zero sum game. A Nash routing game on a simple topology is investigated in Section IV. Section V is devoted to experimental results with focus on the effect of overlay routing on the performance of underlay networks. We conclude the paper in Section VI and discuss possible directions for future work.

II. Related Work

Noncooperative games in the context of routing have been studied in the areas of transportation networks for a long time. In that framework, each user controls just an *infinitesimally small portion* of the network flow, and tries to minimize its own delay or cost. Dafermos and

Sparrow [9] show that a simple transformation of the cost function can make the routing game a standard network optimization problem called the *user equilibrium* model. On the other hand, a *system optimum* model has the objective to minimize the overall delay of the whole network. In the area of computer networks, the *user equilibrium* model is called selfish routing ([7], [10], [11]) Orda [12] and Korilis [13] studied a model in which users control a *non-negligible portion* of flow. Orda [12] investigates the existence and uniqueness of a Nash equilibrium in a routing game in which each user attempts to optimize its own performance by controlling its own portion of traffic. In [13], a central manager is introduced into the model. Other related work can be seen in [14] and [15].

Our work differs from much of previous selfish routing studies ([7], [16], [10]) in that each user or player controls a *non-negligible portion* of flow in our work. Our work also differs from [12] and [13] in the following ways. In our work, the two players (overlay and TE) have different views of the network. Overlay has a logical view of the network, whereas TE has a physical view of the network. In addition, each player's decision can change the input for the other player. The interaction between selfish overlay routing and TE is first brought up in [7]. Two experimental studies were given to show the interaction between overlay and MPLS TE, and the interaction between overlay and OSPF TE. Our work starts from this base to formally study this interaction as a noncooperative game.

III. Models of Interaction

In this section, we formally define the interaction between TE and overlay routing.

A. Physical v.s. Logical Network View

On top of an underlay network, a group of nodes forms an overlay network and all the nodes forward traffic for each other. At the application level, the overlay nodes view a *logical* network. We use $G = (V, E)$ to denote an underlay network and $G' = (V', E')$ to denote an overlay network on top of G. In G', we use i' to represent the overlay node built upon node i in the underlay physical graph G. Overlay node i' is connected to j' by a *logical link* (i', j'), which corresponds to a physical path from i to j in G. [1] In G', a *logical path* p is an ordered set of logical links connecting one overlay node to another overlay node. We introduce the mapping coefficient, $\delta_p^{(s',t')}$, which assumes value 1 if logical link (s', t') is on logical path p, and 0 otherwise. In Figure 2, four nodes

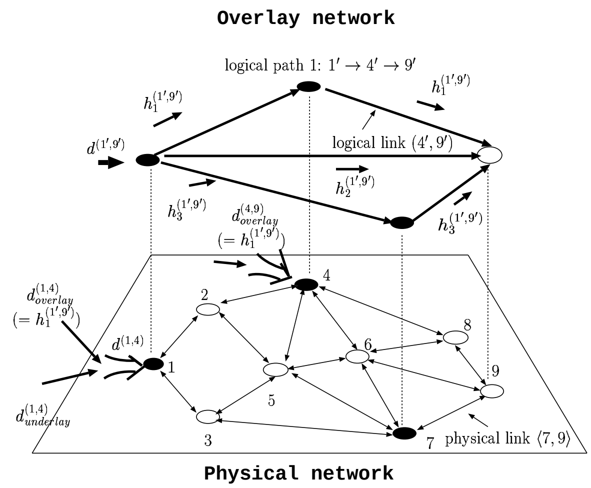


Fig. 2. An example overlay network.

$1, 4, 7, 9$ form an overlay network. There are three logical paths connecting overlay node $1'$ to $9'$: $1' \to 4' \to 9'$; $1' \to 9'$; $1' \to 7' \to 9'$.

In the logical graph G', an overlay routing algorithm allocates traffic demand between overlay node pairs onto different logical paths. Let $d^{(s',t')}$ be the traffic demand from s' to t', $P^{(s',t')}$ be the set of logical paths from s' to t', and $h_p^{(s',t')}$ be the flow rate on logical path $p \in P^{(s',t')}$. Given $\{d^{(s',t')}, P^{(s',t')}, \forall s', t' \in V'\}$, an overlay routing decision can be represented by $\{h_p^{(s',t')}, \forall s', t' \in V', \forall p \in P^{(s',t')}\}$. For the previous example, the overlay routing algorithm finds $\{h_1^{(1',9')}, h_2^{(1',9')}, h_3^{(1',9')}\}$ to implement the single demand pair $d^{(1',9')}$.

Overlay traffic on logical link (i', j') will be physically routed from node i to j by TE. At the same time, TE also accounts for traffic demands from normal underlay users. Therefore, the TE demand on a physical node pair (s, t) can be expressed as

$$d^{(s,t)} = d_{\text{overlay}}^{(s,t)} + d_{\text{under}}^{(s,t)},$$

where $d_{\text{overlay}}^{(s,t)}$ is the demand due to the overlay traffic on logical link (s', t'), and $d_{\text{under}}^{(s,t)}$ is the demand from the underlay users. In Figure 2, overlay traffic on logical path $1' \to 4' \to 9'$ will generate two demands for TE: $d_{\text{overlay}}^{(1,4)} = d_{\text{overlay}}^{(4,9)} = h_1^{(1',9')}$.

In the physical graph G, TE allocates all physical traffic demands $\{d^{(s,t)}, \forall s, t \in V\}$ to all of the physical links $\{a \in E\}$. Let $f_a^{(s,t)}$ denote the fraction of TE

[1] If TE allows a single demand pair to use multiple paths, traffic on a logical link will be distributed among multiple physical paths

(i', j')	:	logical link.
$P^{(s',t')}$	:	Set of logical paths from s' to t'.
$\delta_p^{(s',t')}$	:	Path mapping coefficient.
$d^{(s',t')}$	:	Overlay demand on pair (s', t').
$h_p^{(s',t')}$	:	Overlay flow on logical path p.
$d^{(s,t)}$	:	TE Demand on physical node pair (s, t).
$d_{\text{overlay}}^{(s,t)}$	:	TE demand due to overlay flow on logical link (s', t')
$d_{\text{under}}^{(s,t)}$	:	TE demand due to underlay traffic.
$d^{(\cdot,t)}$	:	Demand of TE to destination t.
$a = \langle i, j \rangle$	:	physical link a connecting node i to j.
C_a	:	Capacity of a physical link a.
l_a	:	Link traffic on a physical link a.
$f_a^{(s,t)}$	:	Fraction of TE demand $d^{(s,t)}$ on link a.
$v_a^{(s,t)}$	:	Flow of $d^{(s,t)}$ on link a.
v_a^t	:	Flow destined to t on link a.

TABLE I

NOTATIONS

demand $d^{(s,t)}$ on physical link a. [2] Then the traffic rate of $d^{(s,t)}$ assigned to physical link a is $v_a^{(s,t)} = f_a^{(s,t)} d^{(s,t)}$. The total traffic rate on link a is

$$l_a = \sum_{(s,t)} v_a^{(s,t)} = \sum_{(s,t)} \{f_a^{(s,t)} \cdot (d_{\text{overlay}}^{(s,t)} + d_{\text{under}}^{(s,t)})\}$$

All notations used in our formulations are summarized in Table I.

B. Traffic Engineering v.s. Overlay Routing

We now formulate the operations of TE and overlay routing. TE and overlay routing each individually optimize some performance metric, such as delay. TE cares about network-wide performance, while overlay only cares about the performance of overlay users. We will observe in the following sections that this objective misalignment between optimizations at logical and physical level leads to bad interactions between TE and overlay routing.

We adopt congestion delay as the network performance metric and use the $M/M/1$ delay formula to calculate link cost [4]. [3] For a physical link with capacity C_a, if its traffic rate is l_a, the mean delay experienced by a single packet is $\frac{1}{C_a - l_a}$. Therefore, the total cost seen by TE on this link is $\frac{l_a}{C_a - l_a}$. Given the demand matrix $\{d^{(s,t)}, \forall s, t \in V\}$, the goal of TE is to choose a physical link flow allocation $\{v_a^{(s,t)}\}$, or equivalently $\{f_a^{(s,t)}\}$, that minimizes network cost:

$$\min_{v_a^{(s,t)}} J^{\text{TE}} = \sum_{a \in E} \frac{\sum_{(s,t)} v_a^{(s,t)}}{C_a - \sum_{(s,t)} v_a^{(s,t)}} \tag{1}$$

subject to physical level flow conservation constraints

$$\sum_{a:d(a)=y} v_a^{(s,t)} - \sum_{a:s(a)=y} v_a^{(s,t)} = \begin{cases} d^{(s,t)} & y = t, \\ -d^{(s,t)} & y = s, \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

$\forall y \in V, \forall (s,t) \in V \times V$, where $s(a)$ and $d(a)$ denote the source node and destination node of link a respectively.

On the other hand, the overlay routing algorithm determines a logical path flow allocation $\{h_p^{(s',t')}\}$ that minimizes the average delay experienced by overlay users. Overlay users can choose their routes independently by probing the underlay network, or a centralized entity can calculate routes for all overlay users. Ideally, if the centralized overlay routing entity knows exactly the physical network topology, traffic demand and TE's routing, optimal overlay routing can be obtained by solving the following non-linear optimization problem:

$$\min_{h_p^{(s',t')}} J^{\text{overlay}} = \sum_a \frac{\sum_{(s,t)} f_a^{(s,t)} d_{\text{overlay}}^{(s,t)}}{C_a - \sum_{(s,t)} \{f_a^{(s,t)} (d_{\text{overlay}}^{(s,t)} + d_{\text{under}}^{(s,t)})\}} \tag{3}$$

subject to logical level flow conservation constraints

$$\sum_{p \in P^{(s',t')}} h_p^{(s',t')} = d^{(s',t')}, h_p^{(s',t')} \geq 0, \forall (s', t') \in V' \times V'.$$

The overlay demand $d_{\text{overlay}}^{(s,t)}$ in (3) can be calculated as

$$d_{\text{overlay}}^{(s,t)} = \sum_{s',t',p} \delta_p^{(s,t)} h_p^{(s',t')}$$

Overlay source routing can be formulated similarly, interested users are referred to a technical report for details [17].

C. Non-cooperative Non-zero Sum Two-player Game

Based on the above formulations, TE and overlay routing are coupled through the mapping from the logical level path to physical level links. They both optimize their objective functions using their own strategies. We can formulate their interaction as a non-cooperative non-zero sum two-player game.

The strategy used by overlay is represented by a vector of logical link flows. A strategy of overlay is one flow configuration on logical links for all overlay demand pairs:

$$\mathbf{d}_{\text{overlay}}^{(\mathbf{s},\mathbf{t})} = (..., d_{\text{overlay}}^{(s,t)}, ...) \tag{4}$$

Recall that $d^{(s',t')}$ denotes overlay demand, but $d_{\text{overlay}}^{(s,t)}$ denotes overlay traffic flow, which are interpreted as demands by TE. As a comparison, the demand seen by

[2] TE doesn't distinguish between overlay and underlay traffic, and uses the same routing fraction $f_a^{(s,t)}$ for all traffic from s to t

[3] In order to focus on the relationships between sets of routes and network congestion (or costs), we do not consider propagation delays in these examples. The case with propagation delays are also of interests to our future research.

TE that comes from underlay traffic is denoted as $d^{(s,t)}_{\text{under}}$. The strategy space Γ^{overlay} of an overlay network is the set of all feasible flow configurations on logical links or paths.

A strategy of TE is represented by one feasible flow configuration on the physical links for all TE demand pairs:

$$\mathbf{f}^{\text{TE}} = (..., f^{(s,t)}_{\langle i,j\rangle}, ...) \quad (5)$$

The strategy space Γ^{TE} of TE is the set of all feasible flow configurations on physical links.

A strategy profile is $\bar{\gamma} = (\mathbf{f}^{\text{TE}}, \mathbf{d}^{(\mathbf{s},\mathbf{t})}_{\text{overlay}})$. The cost function of TE is $J^{\text{TE}}(\mathbf{f}^{\text{TE}}, \mathbf{d}^{(\mathbf{s},\mathbf{t})}_{\text{overlay}})$ and the cost function of overlay is $J^{\text{overlay}}(\mathbf{f}^{\text{TE}}, \mathbf{d}^{(\mathbf{s},\mathbf{t})}_{\text{overlay}})$. We have the following definition of *Nash equilibrium* for this routing game.

Nash Equilibrium A strategy profile $\bar{\gamma}^*$ is a Nash equilibrium if, for both players, TE and overlay,

$$J^{\text{TE}}(\mathbf{f}^{\text{TE}^*}, \mathbf{d}^{(\mathbf{s},\mathbf{t})*}_{\text{overlay}}) \leq J^{\text{TE}}(\mathbf{f}^{\text{TE}}, \mathbf{d}^{(\mathbf{s},\mathbf{t})*}_{\text{overlay}}), \quad \forall \mathbf{f}^{\text{TE}} \in \Gamma^{\text{TE}} \quad (6)$$

$$J^{\text{overlay}}(\mathbf{f}^{\text{TE}^*}, \mathbf{d}^{(\mathbf{s},\mathbf{t})*}_{\text{overlay}}) \leq J^{\text{overlay}}(\mathbf{f}^{\text{TE}^*}, \mathbf{d}^{(\mathbf{s},\mathbf{t})}_{\text{overlay}}), \quad \forall \mathbf{d}^{(\mathbf{s},\mathbf{t})}_{\text{overlay}} \in \Gamma^{\text{overlay}} \quad (7)$$

For a TE optimizer, overlay's response is observed as part of the demand matrix. Since TE knows the physical network's topology and all link capacities, and if we assume TE can estimate its demand matrix accurately ([18]), then TE can compute its optimal strategy. As for the implementation, if TE uses MPLS, it can exactly realize its strategy; if TE uses OSPF, it can only approximately realize its optimal strategy. However, an overlay optimizer may *not* be able to compute its optimal strategy because it may not know all of the necessary information as mentioned in last section. To gain insights into the interaction between TE and overlay routing, in the following sections, we first assume that overlay has the necessary information to compute its optimal routes, and model this interaction as a *Nash routing game* [19]. This assumption is relaxed when we study overlay source routing. the situation where overlay only has limited information will be discussed in Section VI

Our Nash routing game model is a *discrete* time model. One basic assumption is that, during its turn, one player completes its optimization before the other player starts. It could be true of course that a player starts its turn even when the other player has not yet finished. We are not concerned with such an interaction process in this paper. A similar process is studied in [20], [21], [22], and [23].

IV. Nash Routing Game

In this section, we explore the structure of the the previously defined Nash routing game through an illustrative example. Mathematical analysis enables us to gain a fundamental understanding and subtle insights of the intricate interaction process. Two types of overlay routing schemes are considered, centralized optimal routing and selfish source routing. We analytically study the best-reply dynamics of the game-playing process. In particular, the existence, convergence and stability of Nash equilibrium (NEP) are established for this example. Interestingly, even for this simple topology, we identify cases where the Nash game converges to an inefficient NEP for overlay. In other words, overlay's performance degrades as the game proceeds. The best-reply strategy is *not* the best strategy for overlay to use when interacting with TE.

We use the three node topology illustrated in Figure 3 and assume that the bandwidth on two physical links, between node 2 and 3, is large enough such that delay on both links negligible. Without loss of generality, we assume link $\langle 1,2\rangle$ has a higher capacity than link $\langle 1,3\rangle$. Note, TE maintains a physical view of the network and overlay has a logical view of the network. Therefore, logical link $(1',2')$ is actually mapped onto two physical paths $1 \rightarrow 2$ and $1 \rightarrow 3 \rightarrow 2$. We assume the only overlay demand is from node $1'$ to $3'$. We first

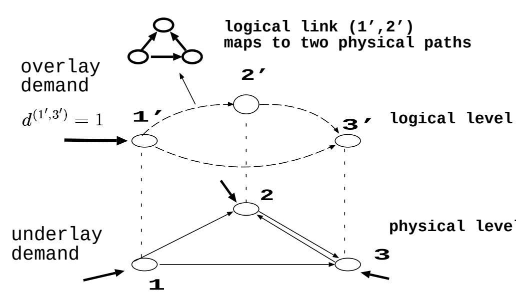


Fig. 3. Topology of a three-node network.

assume that a centralized entity has all of the information necessary to calculate the optimal overlay routes. This assumption is relaxed when we study selfish overlay source routing, where overlay users probe the underlay network and choose the shortest logical paths to send their traffic.

A. Centralized Optimal Overlay Routing

Since link costs between nodes 2 and 3 are negligible, TE's task is to allocate physical traffic demand $d^{(1,2)}$ and $d^{(1,3)}$ onto link $\langle 1,2\rangle$ and $\langle 1,3\rangle$. For this example, the optimal link flow allocation $\{v^{(1,2)}_{\langle 1,2\rangle}, v^{(1,3)}_{\langle 1,2\rangle}\}$ is not unique. However, the optimal link rate vector $\{l_{\langle 1,2\rangle}, l_{\langle 1,3\rangle}\}$ is unique and is the solution to:

$$\min_{\{l_{\langle 1,2\rangle}, l_{\langle 1,3\rangle}\}} J^{TE} = \frac{l_{\langle 1,2\rangle}}{C_{\langle 1,2\rangle} - l_{\langle 1,2\rangle}} + \frac{l_{\langle 1,3\rangle}}{C_{\langle 1,3\rangle} - l_{\langle 1,3\rangle}}$$

subject to

$$l_{\langle 1,2\rangle} + l_{\langle 1,3\rangle} = d^{(1,2)}_{under} + d^{(1,3)}_{under} + d^{(1',3')}, \quad (8)$$

The constraint is due to the fact that TE can arbitrarily allocate $d^{(1,2)}$ and $d^{(1,3)}$ onto links $\langle 1,2\rangle$ and $\langle 1,3\rangle$. This formulation also suggests that for this example, TE's optimal cost is invariant to overlay's routing. This is because we assume the link costs between node 2 and 3 are negligible.

The necessary condition for the TE optimum is $\frac{\partial}{\partial l_{\langle 1,2\rangle}} J^{TE} = \frac{\partial}{\partial l_{\langle 1,3\rangle}} J^{TE}$, which translates into

$$\frac{C_{\langle 1,2\rangle}}{(C_{\langle 1,2\rangle} - l_{\langle 1,2\rangle})^2} = \frac{C_{\langle 1,3\rangle}}{(C_{\langle 1,3\rangle} - l_{\langle 1,3\rangle})^2} \quad (9)$$

Based on (8) and (9), TE can calculate its optimal traffic assignment $\{l^*_{\langle 1,2\rangle}, l^*_{\langle 1,3\rangle}\}$. To avoid ambiguity in the link flow allocation, we force TE to route traffic directly as much as possible. This is consistent with actual practice where the bandwidth on links between node 2 and 3 is always finite. So we have:

$$f^{(1,2)}_{\langle 1,2\rangle} = \begin{cases} 1 & d^{(1,2)} \le l^*_{\langle 1,2\rangle} \\ \frac{l^*_{\langle 1,2\rangle}}{d^{(1,2)}} & d^{(1,2)} > l^*_{\langle 1,2\rangle} \end{cases} \quad (10)$$

and

$$f^{(1,3)}_{\langle 1,2\rangle} = max\{0, (l^*_{\langle 1,2\rangle} - f^{(1,2)}_{\langle 1,2\rangle} \times d^{(1,2)})/d^{(1,3)}\}. \quad (11)$$

Overlay divides the demand $d^{(1',3')}$ among the two logical paths. To simplify notation without causing confusion, we use $h^{(1,2)}$ and $h^{(1,3)}$ to denote overlay traffic on logical path $1' \to 2' \to 3'$ and $1' \to 3'$ respectively. The overlay optimization can be formulated as:

$$\min_{\{h^{(1,2)}, h^{(1,3)}\}} J^{overlay} = \frac{f^{(1,2)}_{\langle 1,2\rangle} h^{(1,2)} + f^{(1,3)}_{\langle 1,2\rangle} h^{(1,3)}}{\tilde{C}_{\langle 1,2\rangle} - f^{(1,2)}_{\langle 1,2\rangle} h^{(1,2)} - f^{(1,3)}_{\langle 1,2\rangle} h^{(1,3)}} + \frac{f^{(1,2)}_{\langle 1,3\rangle} h^{(1,2)} + f^{(1,3)}_{\langle 1,3\rangle} h^{(1,3)}}{\tilde{C}_{\langle 1,3\rangle} - f^{(1,2)}_{\langle 1,3\rangle} h^{(1,2)} - f^{(1,3)}_{\langle 1,3\rangle} h^{(1,3)}} \quad (12)$$

subject to $h^{(1,2)} + h^{(1,3)} = d^{(1',3')}$, where $\tilde{C}_{\langle 1,2\rangle}$ and $\tilde{C}_{\langle 1,3\rangle}$ are available bandwidth for overlay on link $\langle 1,2\rangle$ and $\langle 1,3\rangle$:

$$\tilde{C}_{\langle 1,2\rangle} = C_{\langle 1,2\rangle} - f^{(1,2)}_{\langle 1,2\rangle} d^{(1,2)}_{under} - f^{(1,3)}_{\langle 1,2\rangle} d^{(1,3)}_{under} \quad (13)$$

$$\tilde{C}_{\langle 1,3\rangle} = C_{\langle 1,3\rangle} - f^{(1,2)}_{\langle 1,3\rangle} d^{(1,2)}_{under} - f^{(1,3)}_{\langle 1,3\rangle} d^{(1,3)}_{under} \quad (14)$$

The necessary condition for an overlay optimum with $h^{(1,2)} > 0$ and $h^{(1,3)} > 0$ is :

$$\frac{\partial}{\partial h^{(1,2)}} J^{overlay} = \frac{\partial}{\partial h^{(1,3)}} J^{overlay} \quad (15)$$

which is equivalent to

$$\frac{\tilde{C}_{\langle 1,2\rangle}}{(\tilde{C}_{\langle 1,2\rangle} - f^{(1,2)}_{\langle 1,2\rangle} h^{(1,2)} - f^{(1,3)}_{\langle 1,2\rangle} h^{(1,3)})^2} = \frac{\tilde{C}_{\langle 1,3\rangle}}{(\tilde{C}_{\langle 1,3\rangle} - f^{(1,2)}_{\langle 1,3\rangle} h^{(1,2)} - f^{(1,3)}_{\langle 1,3\rangle} h^{(1,3)})^2} \quad (16)$$

together with (13), (14), the necessary condition in terms of link rate is:

$$\frac{\tilde{C}_{\langle 1,2\rangle}}{(C_{\langle 1,2\rangle} - l_{\langle 1,2\rangle})^2} = \frac{\tilde{C}_{\langle 1,3\rangle}}{(C_{\langle 1,3\rangle} - l_{\langle 1,3\rangle})^2} \quad (17)$$

Existence of NEP Equation (17) and (9) demonstrate the misalignment of the objectives of TE and overlay. It results in oscillations of routes at both logical and physical levels. To reach any NEP with $h^{(1,2)} \cdot h^{(1,3)} > 0$, we must have $\frac{\tilde{C}_{\langle 1,2\rangle}}{C_{\langle 1,2\rangle}} = \frac{\tilde{C}_{\langle 1,3\rangle}}{C_{\langle 1,3\rangle}}$. It is easily satisfied for the trivial case when there is no underlay traffic. We have shown that if $d^{(1,2)}_{under} = 0$, $d^{(1,3)}_{under} > 0$ and $d^{(1,3)}_{overlay} < l^*_{\langle 1,2\rangle}$, there exists one unique NEP with $h^{(1,2)} \cdot h^{(1,3)} > 0$. On the other hand, if $d^{(1,2)}_{under} > 0$ and $d^{(1,3)}_{under} = 0$, $\frac{\tilde{C}_{\langle 1,2\rangle}}{C_{\langle 1,2\rangle}} \neq \frac{\tilde{C}_{\langle 1,3\rangle}}{C_{\langle 1,3\rangle}}$, the possible NEPs are on the boundary, i.e., $h^{(1,2)} = 0$ or $h^{(1,3)} = 0$. Since the delay on logical path $(1,2)$ is always smaller than on logical path $(1,3)$, $\{h^{(1,2)} = d^{(1,3)}_{overlay}, h^{(1,3)} = 0\}$ is the only NEP.

Characteristics of NEP We are interested in the characteristics of NEP, namely, stability and efficiency (for either overlay or TE), which are of practical importance. It is easy to show that those NEPs occurring on the boundary are stable and give overlay lower cost compared with initial cost. One interesting NEP is identified in the following theorem:

Theorem 1: If $d^{(1,2)}_{under} = 0$ and $d^{(1',3')} < l^*_{\langle 1,2\rangle}$, the NEP between TE and overlay is unique and globally stable, i.e. overlay routing always converges to the NEP regardless of its initial routing.

Proof: See Appendix A. ■

One interesting observation is that this NEP is *inefficient* for overlay for some initial conditions, namely, overlay's cost at NEP is higher that its initial cost at the beginning of the interaction process. To illustrate, we present results from one experiment. We set $C_{\langle 1,2\rangle} = 1$, $C_{\langle 1,3\rangle} = 0.5$, $d^{(1',3')} = d^{(1,3)}_{under} = 0.5$. Overlay takes its turn at even rounds, TE at odd rounds. We use the Matlab optimization toolbox to solve (8) and (12). We performed two experiments with different initial overlay route: $h^{(1,2)}(0) = d^{(1',3')}$; $h^{(1,2)}(0) = 0$. Figure 4(a) shows that in both cases overlay routing converges to the NEP. Figure 4(b) shows overlay cost as the Nash game proceeds.

It is interesting to observe that for the case where $h^{(1,2)}(0) = d^{(1',3')}$, overlay cost actually increases over rounds even though it tries to minimize its cost at each round. This is because after overlay chooses its routes, TE will adjust its routes to minimize the whole network cost. The updated TE routes will increase overlay's cost. The interaction between TE and overlay routing is bad for overlay and overlay cost increases until the game converges to its NEP. We observe the same phenomenon in experiments on a 14-node tier-1 ISP network later. For the current example, the best strategy for overlay is to place all of its traffic on logical path $(1,2)$. This is to say it may *not* be wise for overlay to play a Nash game with TE. This is consistent to the inefficiency property of NEP. A Stackelberg routing strategy for overlay will be discussed in Section VI.

B. Selfish Overlay Source Routing

In practice, it is difficult to have a centralized entity calculate optimal routes for the overlay users. In most cases, overlay users choose their own routes by probing the underlay network. Based on the TE routes, overlay users try to move their traffic to the minimum delay logical path. Just as in normal selfish source routing, an equilibrium between competing selfish users will be reached if delays on all paths utilized by one user are equal. For our current example, selfish overlay users reach an equilibrium either when the delay on the two logical paths are equal or all overlay users shift their traffic to the minimum delay logical path. We assume there is no underlay traffic, and TE and overlay takes turns in doing optimizations. We prove that a NEP exists and can still be reached when overlay employs selfish source routing.

For the clarity of the proof, we approximate the $M/M/1$ link cost function by a convex piece-wise linear function [4]:

$$\Phi(C,l) = \begin{cases} k_1 l + b_1 & l \in [0, \alpha_1 \cdot C) \\ \cdots & \cdots \\ k_m l + b_m & l \in [\alpha_{m-1} \cdot C, C), \end{cases}$$

where the slope k_i increases with i. As described in (1), the objective of TE is to minimize the summation of the costs on physical links $\langle 1,2\rangle$ and $\langle 1,3\rangle$:

$$\textbf{min} \quad J^{\text{TE}} = \Phi(C_{\langle 1,2\rangle}, l_{\langle 1,2\rangle}) + \Phi(C_{\langle 1,3\rangle}, l_{\langle 1,3\rangle}) \quad (18)$$

subject to $l_{\langle 1,2\rangle} + l_{\langle 1,3\rangle} = d^{(1,2)} + d^{(1,3)}$. The necessary condition for the TE's optimal solution is $\frac{d\Phi_{\langle 1,2\rangle}}{dl_{\langle 1,2\rangle}} = \frac{d\Phi_{\langle 1,3\rangle}}{dl_{\langle 1,3\rangle}}$, which means $\frac{l_{\langle 1,2\rangle}}{C_{\langle 1,2\rangle}}$ falls into the same region $[\alpha_{j-1}, \alpha_j)$ as $\frac{l_{\langle 1,3\rangle}}{C_{\langle 1,3\rangle}}$. To illustrate, we plot the two-

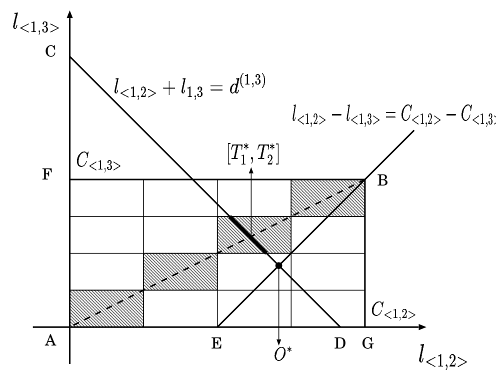


Fig. 5. Traffic allocation on two physical links $\langle 1,2\rangle$ and $\langle 1,3\rangle$

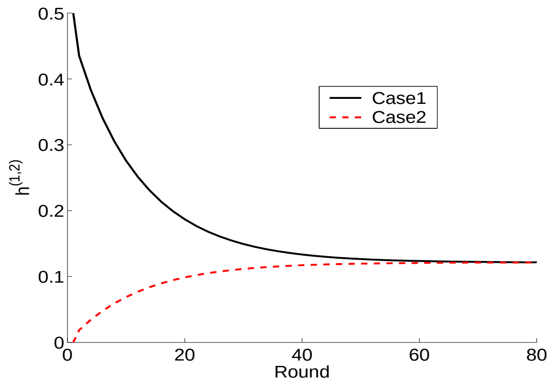
dimension link rate vector $\{l_{\langle 1,2\rangle}, l_{\langle 1,3\rangle}\}$ in Figure 5. We draw vertical lines at $l_{\langle 1,2\rangle} = \alpha_i C_{\langle 1,2\rangle}, 1 \le i \le m$. Between two adjacent vertical lines $j-1$ and j, $\frac{d\Phi_{\langle 1,2\rangle}}{dl_{\langle 1,2\rangle}} = k_j$. Similarly, We draw horizontal lines at $l_{\langle 1,3\rangle} = \alpha_i C_{\langle 1,3\rangle}, 1 \le i \le m$ and between two adjacent horizontal lines $j-1$ and j, $\frac{d\Phi_{\langle 1,3\rangle}}{dl_{\langle 1,3\rangle}} = k_j$. As shown in Figure 5, the plane is partitioned into blocks. Within each block, the link cost derivatives are constant. Since k_j increases in j, two link cost derivatives are equal only when the link rate vector falls into those shaded blocks along the diagonal. Therefore, the TE's optimal solution set is $[T_1^*, T_2^*]$, the intersection between the constraint line CD and the shaded area. The non-uniqueness of the TE optimal solution is due to the piece-wise linear link cost function. As more linear segments used in the piece-wise linear function, the smaller the shaded area. As the size of the linear segments goes to zero, the shaded area degenerates to the diagonal line AB and TE has a unique optimal solution T^*. TE's optimal routes are still calculated as in (10), (11) and $f^{(1,2)}_{\langle 1,2\rangle} > f^{(1,3)}_{\langle 1,2\rangle}$ always. The average delay on logical links can be calculated as:

$$\text{Delay}_{(1,2)} = f^{(1,2)}_{\langle 1,2\rangle}\text{Delay}_{\langle 1,2\rangle} + (1 - f^{(1,2)}_{\langle 1,2\rangle})\text{Delay}_{\langle 1,3\rangle} \quad (19)$$

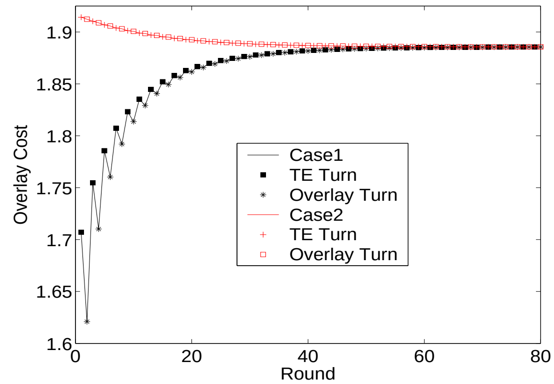
$$\text{Delay}_{(1,3)} = f^{(1,3)}_{\langle 1,2\rangle}\text{Delay}_{\langle 1,2\rangle} + (1 - f^{(1,3)}_{\langle 1,2\rangle})\text{Delay}_{\langle 1,3\rangle} \quad (20)$$

Theorem 2: The Nash routing game defined above always converges to a NEP.

Proof: Because $f^{(1,2)}_{\langle 1,2\rangle} > f^{(1,3)}_{\langle 1,2\rangle}$, the only way to match the average delays on two logical paths is to match the average delays on physical links $\langle 1,2\rangle$ and $\langle 1,3\rangle$. We introduce in Figure 5 the Equal Delay Line EB, where



(a) Overlay routing globally converges to the NEP



(b) Overlay cost decreases/increases with two different initial routings

Fig. 4. Convergence of Routing Game Between Overlay and TE

the rate vector makes the physical delay on link $\langle 1,2\rangle$ and $\langle 1,3\rangle$ equal, or equivalently

$$C_{\langle 1,2\rangle} - l_{\langle 1,2\rangle} = C_{\langle 1,3\rangle} - l_{\langle 1,3\rangle}$$

Then any rate allocation in area $AEBF$ will make $\text{Delay}_{\langle 1,3\rangle} > \text{Delay}_{\langle 1,2\rangle}$ and $\text{Delay}_{\langle 1,3\rangle} < \text{Delay}_{\langle 1,2\rangle}$ in area EGB. Let point O^* be the intersection between line EB and the demand line CD. If O^* falls in TE's optimal solution interval $[T_1^*, T_2^*]$, then O^* achieves optimum for both TE and overlay. Therefore O^* is a NEP and will be reached after one round of TE and overlay optimization.

If EB falls outside of $[T_1^*, T_2^*]$, then after TE's optimization, we have $\text{Delay}_{\langle 1,2\rangle} < \text{Delay}_{\langle 1,3\rangle}$ and $f^{(1,2)}_{\langle 1,2\rangle} > f^{(1,3)}_{\langle 1,2\rangle}$. Therefore, from (19), we will have $\text{Delay}_{(1,2)} < \text{Delay}_{(1,3)}$. When overlay takes its turn, it always try to move some of its demand from logical path $(1,3)$ to logical path $(1,2)$ until either all of its demand has been moved to path $(1,2)$ or the rate vector reach point O^*. For the first case, when TE takes over, it will pull the rate vector back into $[T_0^*, T_1^*]$. And when overlay takes its turn, it still sees $\text{Delay}_{(1,2)} < \text{Delay}_{(1,3)}$. Since all of its demand has already been placed on path $(1,2)$, the game reaches its NEP. For the second case, TE again will pull the rate vector back into $[T_0^*, T_1^*]$. When overlay takes its turn, it will increase $d^{(1,2)}$ and drive the rate vector back to O^*. The interaction continues and $d^{(1,2)}$ keeps increasing until all of the overlay traffic is moved to logical path $(1,2)$. The game converges as in the first case. ■

Again, we see from this proof that the objective misalignment between TE and overlay causes traffic oscillations on network links. In addition, the driving force for this process to converge to a Nash equilibrium is that TE can always map a set of logical link loads required by overlay to a physical flow assignment. This mapping of logical links to physical paths by TE plays a key role in resolving the routing conflicts at logical and physical levels. However, this mapping is network topology dependent, it is difficult to draw general conclusions for arbitrary network topologies.

V. Impact of Overlay Routing on Underlay Network

The existence and stability of Nash equilibria are much more difficult to establish for general network topologies and traffic demand patterns. And, even if a Nash equilibrium exists, the interaction process may not converge to it, as will be shown in our experiments. A more important question we want to answer is how the selfish behavior of overlay routing influences the performance of TE in this game. In this section, we prove that TE's performance will never be improved in this Nash routing game. Various experimental results on a 9-node network given in [7], and a 14-node tier-1 POP network in [18] are presented to demonstrate overlay routing's impact on the underlay network's performance. Similar to the inefficient NEP in Section IV-A, we identify in our experiments some case where the routing interaction is inefficient for overlay, i.e., overlay's cost increases as the iterative process proceeds even though it plays optimally based on TE's routing at each round.

A. Overlay Routing's Impact on the Cost of TE

While overlay routing aims at improving the performance of overlay traffic, the improvement comes at the cost of degrading the performance of underlay traffic. In addition, if we assume TE can perfectly implement the optimal solution, overlay routing cannot improve the overall network performance. In many cases, overlay routing increases the network cost that TE tries to minimize.

Base cost of TE. The *base cost* of TE refers to the optimal cost achieved when overlay simply provides its demand matrix to TE without making any routing decisions on the overlay level, i.e., $d_{\text{overlay}}^{(s,t)} = d^{(s',t')}$.

Theorem 3: Overlay routing never improves TE's performance.

Proof: We compare the network cost with and without overlay routing. Let P_O be the set of source-destination pairs of overlay demands and $\{D_O^{(s',t')}, (s',t') \in P_O\}$ the overlay demand vector. Without overlay routing, TE will take overlay demand and underlay demand directly as its overall physical demand:

$$d^{(s,t)} = \begin{cases} D_O^{(s',t')} + d_{\text{under}}^{(s,t)} & (s',t') \in P_O \\ d_{\text{under}}^{(s,t)} & (s',t') \notin P_O \end{cases} \tag{21}$$

Then TE's optimal set of routes $\{\bar{v}_a^{(s,t)}\}$ produces the minimum cost over all feasible sets of routes under constraint (2), i.e., all $\{v_a^{(s,t)}\}$ which satisfy flow conservation and implement all TE demand (21).

Overlay can assign traffic between any overlay demand pair $\{(s',t') \in P_O\}$ on all associated logical paths $\{p \in P^{(s',t')}\}$. The traffic demand seen by TE can be calculated as:

$$d^{(s,t)} = \begin{cases} d_{\text{under}}^{(s,t)} + \sum\limits_{(i',j',p)} \delta_p^{(s',t')} h_p^{(i',j')} & (s',t') \in E' \\ d_{\text{under}}^{(s,t)} & (s',t') \notin E' \end{cases} \tag{22}$$

Any set of TE routes $\{\tilde{v}_a^{(s,t)}\}$ (and consequently $\{\tilde{f}_a^{(s,t)}\}$) based on any set of overlay routes $\{h_p^{(s',t')}, (s',t') \in P_O, p \in P^{(s',t')}\}$ must implement TE demand as described in (22). At the same time, we calculate the amount of traffic which is from overlay demand pair (s',t') and is placed on each physical link as:

$$\hat{v}_a^{over(s',t')} = \sum_{p \in P^{(s',t')}} h_p^{(s',t')} \cdot \Big(\sum_{(i',j') \in E'} \delta_p^{(i',j')} \tilde{f}_a^{(i,j)} \Big) \tag{23}$$

Based on (23), we construct TE link flow routes

$$\hat{v}_a^{(s,t)} = \begin{cases} \tilde{f}_a^{(s,t)} d_{\text{under}}^{(s,t)} + v_a^{over(s',t')} & (s',t') \in P_O \\ \tilde{f}_a^{(s,t)} d_{\text{under}}^{(s,t)} & (s',t') \notin P_O \end{cases} \tag{24}$$

Here $\{\hat{v}_a^{(s,t)}\}$ accounts for both underlay and overlay traffic demands in the absence of overlay routing as described in (21). Therefore, $J^{TE}(\{\hat{v}_a^{(s,t)}\}) \geq J^{TE}(\{\bar{v}_a^{(s,t)}\})$. At the same time, the aggregate traffic rate vector on all physical links $\{\hat{l}_a\}$ under $\{\hat{v}_a^{(s,t)}\}$ is the same as the link rate vector $\{\tilde{l}_a\}$ under $\{\tilde{v}_a^{(s,t)}\}$ with overlay routing. Since link cost is only a function of its aggregate rate, we have $J^{TE}(\{\tilde{v}_a^{(s,t)}\}) = J^{TE}(\{\hat{v}_a^{(s,t)}\}) \geq J^{TE}(\{\bar{v}_a^{(s,t)}\})$. ∎

B. Experimental Study

To verify our analysis, we performed extensive experiments on different network topologies. Due to page limits, we report two sets of experiments here. In our experiments, we solve overlay and TE's optimization problem numerically. Following a similar approach in [4], we use a piece-wise linear version of the non-linear programming formulations of TE and overlay optimizers defined in (1) and (3) respectively. We then use lp_solve[24] to solve these linear programming problems.

1) TE cost change as a function of the percentage of overlay traffic: We first present experimental results for the 9-node example ([7]) in Figure 2. Three overlay nodes are considered: $1, 4, 7$. They may have demands to each other, or to other nodes not in the overlay. In any case, each overlay node can forward traffic originating from any other overlay node. There are 24 possible overlay demand pairs in this example. We randomly choose 70% of them. We use a bimodal traffic matrix ([18]) generated by a mixture of two Gaussians, one with ($\mu_1 = 1.5, \sigma_1 = 0.2$), and the second with ($\mu_2 = 4, \sigma_2 = 0.2$). These means and standard deviations are proportional to those used in [18]. We set the overlay demands to be 60% of the total traffic demands. Specifically, we divided demand on each Source-Destination pair into two parts: one is overlay; the other one is underlying. Overlay demands takes 60% of the total demand. To prevent flows from exceeding link capacities, we set all of the link capacities to be 18. We also perform experiments when link capacities are randomly distributed. For brevity, we do not present results for the random link capacity case here. Heterogeneous link capacity cases can be seen in experiments with a 14-node tier-1 ISP network.

Initially, overlay demand pairs are given to TE without any forwarding among overlay nodes, achieving the *base cost* of TE. We let TE begin the interaction game. TE takes a turn at every odd step, and overlay takes a turn at every even step. We let this interaction process run for 100 steps. Taking the cost at step 1 as the baseline, we calculate the percentage of deviation from it at the following steps. These percentages of deviation of both TE and overlay are plotted in Figure 6.

We observe from these graphs that there are large oscillations in both players' costs during the observed time interval (this game does not converge). At each even step, overlay's response causes an increase to TE's cost, and then TE reacts optimally to reduce its cost at the following odd step. Overlay does the same thing. On

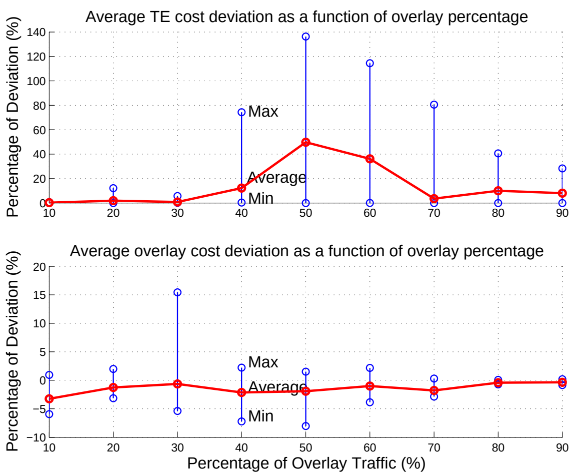


Fig. 7. Cost change of TE and overlay. Percentage of deviation from cost at step 1. Nine-node network. BWs of all links are 18.

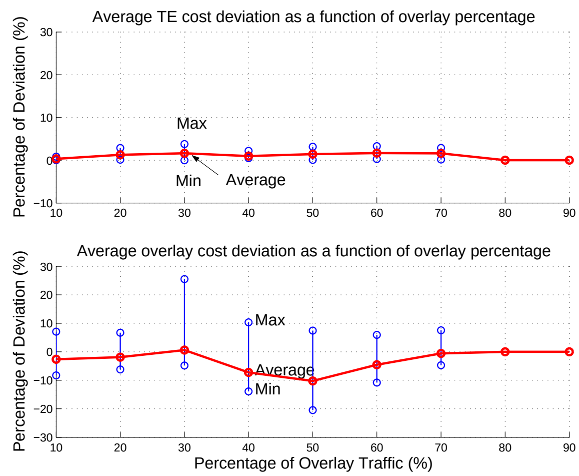


Fig. 8. Cost change of TE and overlay. Percentage of deviation from cost at step 1. Nine-node network. BWs of all links are 20.

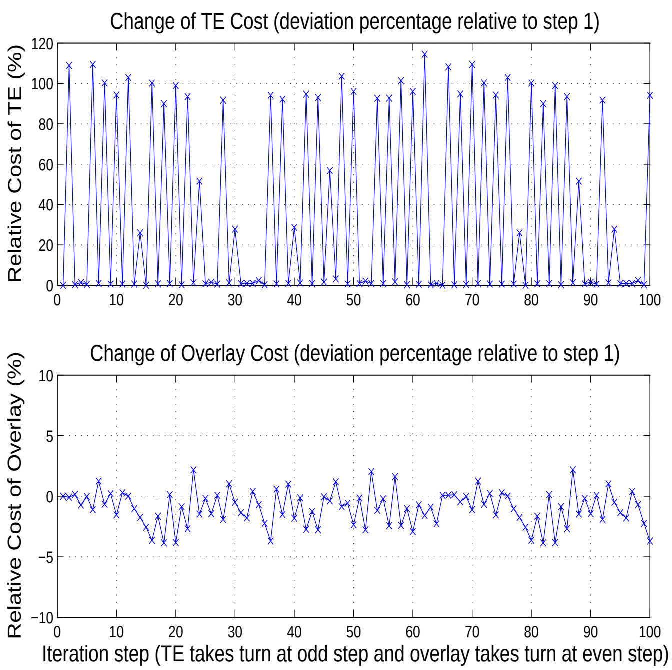


Fig. 6. Cost change of TE and overlay. Percentage of deviation from cost at step 1 at each step in the interaction process.

average, overlay's cost decreases 1% in this interaction process, but TE's cost increases 35.9% as expected.

We are interested in how the cost change of TE varies as the percentage of overlay traffic varies. Our conjecture is as follows. If there is little overlay traffic, then overlay's routing decisions will have little influence on TE's cost. If all traffic consists of overlay traffic, then overlay's routing decision would be the same as that of TE, so, the interaction process will always converge, and TE's cost will not be affected. If there is some significant fraction of overlay traffic, e.g. 50%, TE's cost increase will be maximal.

In Figure 7, we plot the cost deviation percentage for different overlay traffic percentages. Our conjecture is verified through these experiments. In addition, we notice that, when overlay demand is approximately half of total network demand, not only is the average cost increase to TE the largest, but also the variation range is the largest. Larger variations in TE cost reflect greater oscillations in the interaction process, which is clearly harmful to TE.

Another interesting observation is that, if we increase link capacities, the decrease in overlay cost by playing Nash game is not as large as that when link capacities are smaller. This can be seen by comparing Figure 7 with Figure 8. Furthermore, TE's cost is not affected much by overlay's selfish behavior if link capacities increase. Intuitively speaking, this is because TE optimizer has more freedom to allocate traffic to achieve the same minimal cost when link capacities are large.

2) Experiments with a tier-1 ISP network: We also perform extensive experiments on a 14-node tier-1 POP network described in [18]. We invert the weights of links to obtain link capacities. This is based on the assumption that weights are set by turning around capacities as recommended by Cisco. Depending on the traffic matrix used, we multiply these capacities by a certain factor to make sure that, for the traffic matrix we use, no link capacity is exceeded by traffic on that link. Our experimental results confirm our hypotheses presented in previous sections. We present the results of two experiments here.

In these two experiments, we use a bimodal traffic matrix for underlay traffic which is the same as used

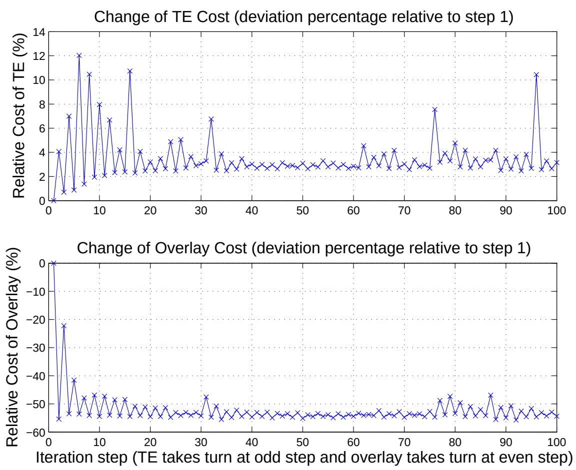


Fig. 9. Cost change of TE and overlay. Percentage of deviation from cost at step 1. A tier-1 ISP network. Experiment setting 1.

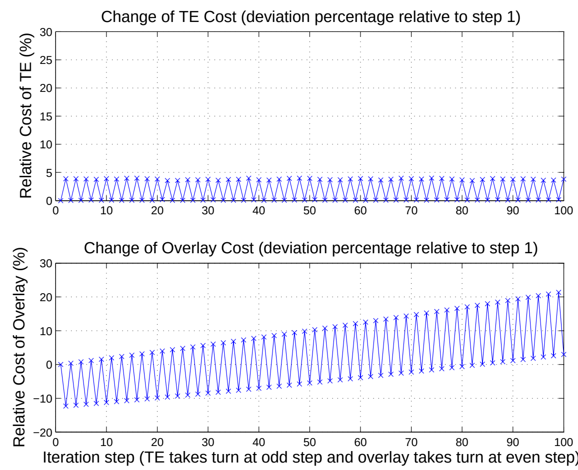


Fig. 10. Cost change of TE and overlay. Percentage of deviation from cost at step 1. A tier-1 ISP network. Experiment setting 2.

in the earlier nine-node experiments. We choose three nodes $6, 10, 11$ as overlay nodes, and randomly choose 32 overlay demand pairs among all possible 39 overlay demand pairs. We add an additional $p\%$ overlay demands. Specifically, for a node pair that is chosen as an overlay demand pair, if the underlay traffic demand is d, we add $d \cdot p\%$ overlay traffic.

In one experiment, we choose $p = 50$. Thus, the total overlay traffic among all network traffic is 8.1%. We run this experiment 100 steps and cost oscillations for TE and overlay during these 100 steps is plotted in Figure 9. The mean cost increase for TE is 3.5%, and the mean cost decrease for overlay is 52.1%. Since the percentage of overlay traffic is small, the cost increase to TE is not big, but still, this small percentage of overlay causes significant oscillations to TE's cost. The highest increase to TE's cost can reach 12%. In another experiment, we choose $p = 68$. Thus, the total overlay traffic among all network traffic is 10.8%. We run this experiment 100 steps. The results are plotted in Figure 10. The mean cost increase for TE is 3%. We observe an increasing trend of overlay cost in these 100 steps. At the final step, even after overlay's optimization, the cost of overlay is 3% higher than the cost in the very first step when overlay does nothing to optimize its routes. The results of this experiment are consistent with our analysis on inefficient NEP in Section IV-A. This experiment verifies our counter-intuitive conclusion that it is *not* always good for an overlay to optimize its routes based on TE's routes at each step. We will briefly discuss a *Stackelberg game* strategy for overlay to address this in the next section.

VI. Conclusions and Future Work

Using game-theoretic models, we provide insights into the fundamental problem on the interaction between the overlay routing optimizer and Traffic Engineering optimizer. Our analytical results for a simple network example provides us with a clear understanding of the existence, uniqueness and stability of Nash equilibrium for this interaction game. We demonstrate, both analytically and experimentally, that the objective misalignment between overlay and TE triggers oscillations in their routes. The selfish behavior of overlay routing optimizer degrades the performance of regular users and the underlay network as a whole. Large oscillations in the TE and overlay costs can be expected in this routing interaction process when overlay accounts for a non-negligible portion of the total network traffic. Even more surprisingly, overlay cost can increase even if overlay optimizes its routing at each iteration, which is not only explained clearly in our analysis of an example network, but also observed in our experiments in a tier-1 ISP network. Even though this observation seems counter-intuitive at first thought, it actually points out the inefficiency of NEP in general.

We believe our work provides a starting point in the search for a complete understanding of the interaction between overlay routing and traffic engineering. Our analytical and experimental studies have identified a rich set of research problems to be investigated. Future work can be pursued in the following directions:
I. In this paper, we assume TE and overlay have equal status and play a Nash game at the same frequency. In current network operation, TE usually happens at a much slower time-scale than that of overlay. This misalignment

of time-scale deserves more investigation. Another situation of interest to us is when one player can predict the other player's response (equivalent to knowing the other player's optimization algorithm.) In this case, the player who has this information and can move faster may choose to play a Stackelberg game ([25], [19]) against the other player (follower.) For example, if an overlay optimizer knows the optimization algorithm used by TE optimizer, then it can predict TE's new physical routings in response to overlay's logical level routing decisions, and then choose an optimal set of logical level routings in consideration of TE's potential responses. We can model this interaction as a *static Stackelberg routing game* [25]. Some preliminary results on this type of game are reported in [17].

II. A natural extension of our work in this paper is the interaction between multiple overlays and TE. This is a much harder problem. For example, we can assume TE does not change its routings during the game playing process between N overlays. We can think of each overlay as a single user who controls a non-negligible amount of traffic and tries to minimize its own group's average cost. Then, this problem is similar to the routing games studied in [12] and [13]. However, there is a significant difference. In [12] and [13], all users work at the same *physical* level, and a link's cost is only a function of load on this link. But in our case, all users work at the *logical* level, and multiple logical links may share the same physical link, so, the cost of a single logical link might be coupled with the cost of other logical links. This logical link load *coupling* makes the existence of Nash equilibrium problem dependent (on network topology, traffic demand patterns.) Furthermore, even if a Nash equilibrium exists for a certain network routing game, the dynamic process of playing a Nash game may not be able to converge to that point.

III. One basic assumption of our models is that TE and overlay have the same frequency and timing of adjusting strategies. But frequency and timing exert an important influence on the structure on this routing game. For example, if overlay knows the starting time of TE's optimization, it can take advantage by doing its optimization immediately after TE's turn. In practice, an overlay most likely will not have all of the necessary information to play with TE. Thus, the estimation of useful information and choice of good strategies for both TE and an overlay optimizer are also important topics. The interaction between overlay routing and the policy based inter-AS routing is another interesting problem to look into.

ACKNOWLEDGMENT

The authors would like to thank reviewers for their valuable comments. This work is supported in part by NSF under grants ITR-0325868, ITR-0085848 and EIA-0080119, and DARPA under contract DOD F30602-00-0554. Any opinions, findings, and conclusions of the authors do not necessarily reflect the views of NSF and DARPA.

REFERENCES

[1] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: a case for informed internet routing and transport," Tech. Rep. TR-98-10-05, 1998.

[2] David Andersen, Hari Balakrishnan, M. Frans Kaashoek, and Robert Morris, "Resilient overlay networks," in *Proc. 18th ACM Symposium on Operating Systems Principles*, Banff, Canada, October 2001.

[3] Stefan Savage, Andy Collins, Eric Hoffman, John Snell, and Thomas E. Anderson, "The end-to-end effects of internet path selection," in *Proceedings of SIGCOMM*, Boston, MA, August-September 1999, pp. 289–299.

[4] Bernard Fortz and Mikkel Thorup, "Internet traffic engineering by optimizing OSPF weights," in *INFOCOM (2)*, 2000, pp. 519–528.

[5] David Applegate and Edith Cohen, "Making intra-domain routing robust to changing and uncertain traffic demands: understanding fundamental tradeoffs," in *Proceedings of ACM SIGCOMM*. 2003, pp. 313–324, ACM Press.

[6] J. Murchland, "Braess's paradox of traffic flow," *Transpn. Res.*, vol. 4, pp. 391–394, 1970.

[7] Lili Qiu, Yang Richard Yang, Yin Zhang, and Scott Shenker, "On selfish routing in internet-like environments," in *Proceedings of the ACM SIGCOMM*, Karlsruhe, Germany, august 2003, ACM, All ACM Conferences, pp. 151–162.

[8] P. Dubey, "Inefficiency of nash equilibria," *Math. Oper. Res.*, vol. 11, pp. 1–8, Feb. 1986.

[9] S. C. Dafermos and F. T. Sparrow, "The traffic assignment problem for a general network," *Journal of Research of the National Bureau of Standards*, vol. Series B, 73B(2), pp. 91–118, 1969.

[10] Tim Roughgarden and Eva Tardos, "How bad is selfish routing?," in *IEEE Symposium on Foundations of Computer Science*, 2000, pp. 93–102.

[11] Elias Koutsoupias and Christos Papadimitriou, "Worst-case equilibria," *Lecture Notes in Computer Science*, vol. 1563, pp. 404–413, 1999.

[12] Ariel Orda, Raphael Rom, and Nahum Shimkin, "Competitive routing in multiuser communication networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 5, pp. 510–521, 1993.

[13] Yannis A. Korilis, Aurel A. Lazar, and Ariel Orda, "Achieving network optima using Stackelberg routing strategies," *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 161–173, 1997.

[14] E. Altman, T. Basar, and R. Srikant, "Nash equilibria for combined flow control and routing in networks: Asymptotic behavior for a large number of users," *IEEE Transactions on Automatical Control*, vol. 47, no. 6, pp. 917–930, 2002.

[15] S. H. Rhee and T. Konstantopoulos, "Optimal flow control and capacity allocation in multi-service networks," in *IEEE Conference on Decision and Control*, Tampa, FL, 1998.

[16] M. Florian and D. Hearn., *Network Routing, chapter 6, Network equilibrium models and algorithms, page 485-550.*, Elsevier Science, 1995.
[17] Honggang Zhang, Yong Liu, Weibo Gong, and Don Towsley, "Understanding the interaction between overlay routing and mpls traffic engineering," *Technical Report, Computer Science Department of University of Massachusetts at Amherst*, 2004.
[18] Alberto Medina, Nina Taft, Kave Salamatian, Supratik Bhattacharyya, and Christophe Diot, "Traffic matrix estimation: Existing techniques compared and new directions," in *Proceedings of ACM SIGCOMM 2002*, August 2002.
[19] T. Basar and G.J. Olsder, *Dynamic Noncooperative Game Theory*, Academic Press, New York, 1998.
[20] Tamer Basar, "Relaxation techniques and asynchronous algorithms for on-line computation of noncooperative equilibria," *Journal of Economic Dynamics and Control*, vol. 11, pp. 531–549, 1987.
[21] Shu Li and Tamer Basar, "Distributed algorithms for the computation of noncooperative equilibria," *Automatica*, vol. 23, no. 4, pp. 523–533, 1987.
[22] A. Cournot, *Researches into the Mathematical Principles of the Theory of Wealth*, The MacMillan Company, New York: Kelley, 1927.
[23] J. Friedman and C. Mezzetti, "Learning in games by random sampling," *Journal of Economic Theory*, vol. 98, no. 1, pp. 55–84, 2001.
[24] LP_Solve, "ftp://ftp.ics.ele.tue.nl/pub/lp_solve/," .
[25] H. von Stackelberg, *The Theory of the Market Economy*, Oxford University Press, Oxford, England, 1952.

VII. Appendix

A. Global Stability of the NEP of the Routing game in Section IV-A

Proof: Let x denote overlay demand on the logical path $(1,2)$. When TE takes turn, it updates routing fraction $f^{(1,2)}_{\langle 1,2\rangle}$ and $f^{(1,3)}_{\langle 1,2\rangle}$ as a function of x as described in (10) and (11). Since $d^{(1,2)}_{under} = 0$, and $d^{(1',3')} < l^*_{\langle 1,2\rangle}$, we always have $f^{(1,2)}_{\langle 1,2\rangle} = 1$ and $f^{(1,3)}_{\langle 1,2\rangle}$ decreasing with x. Consequently, the available bandwidth $\{\tilde{C}_{\langle 1,2\rangle}, \tilde{C}_{\langle 1,3\rangle}\}$ can be recalculated according to (13), (14). It is easy to show $\tilde{C}_{\langle 1,2\rangle}$ ($\tilde{C}_{\langle 1,3\rangle}$) is an increasing (decreasing) function of x. Therefore there is only one solution x_0 satisfying

$$\frac{\tilde{C}_{\langle 1,2\rangle}(x)}{C_{\langle 1,2\rangle}} = \frac{\tilde{C}_{\langle 1,3\rangle}(x)}{C_{\langle 1,3\rangle}}.$$

As discussed in Section IV-A, x_0 is the only NEP.

Let $x(k)$ denote overlay demand on the logical path $(1,2)$ after the kth overlay optimization. In order to prove that the NEP x_0 is globally stable, it is sufficient to show that if $x(k) < x_0$, then $x(k) < x(k+1) < x_0$; if $x(k) > x_0$, then $x(k) > x(k+1) > x_0$. We prove here for $x(k) < x_0$ only, the case for $x(k) > x_0$ can be proved similarly.

First we want to show if $x(k) < x_0$, then $x(k+1) > x(k)$. Let's construct a function of $\{x(k), x\}$ as

$$g(x(k),x) = \frac{(\tilde{C}_{\langle 1,2\rangle} - f^{(1,2)}_{\langle 1,2\rangle}x - f^{(1,3)}_{\langle 1,2\rangle}(d^{(1,3)overlay} - x))^2}{(\tilde{C}_{\langle 1,3\rangle} - f^{(1,2)}_{\langle 1,3\rangle}x - f^{(1,3)}_{\langle 1,3\rangle}(d^{(1,3)overlay} - x))^2},$$

where $\{\tilde{C}_{\langle\cdot\rangle}, f^{(\cdot)}_{\langle\cdot\rangle}\}$ are functions of $x(k)$. Since we have $f^{(1,2)}_{\langle 1,2\rangle} = 1 \geq f^{(1,3)}_{\langle 1,2\rangle}$, and $f^{(1,2)}_{\langle 1,3\rangle} = 0$, it is easy to verify that for any fixed $x(k)$, $g(x(k),x)$ is a decreasing function of x. After TE's optimization and before the $k+1$th round overlay optimization, the overlay routing variable is $x(k)$ and the traffic rate vector on physical links is TE's optimal solution. Based on (9), we have $g(x(k),x(k)) = \frac{C_{\langle 1,2\rangle}}{C_{\langle 1,3\rangle}}$. After overlay's optimization, $x(k+1)$ satisfies (16). Therefore, $g(x(k),x(k+1)) = \frac{\tilde{C}_{\langle 1,2\rangle}(x_k)}{\tilde{C}_{\langle 1,3\rangle}(x_k)}$. Since $\frac{\tilde{C}_{\langle 1,2\rangle}(x)}{\tilde{C}_{\langle 1,3\rangle}(x)}$ is an increasing function of x,

$$\frac{\tilde{C}_{\langle 1,2\rangle}(x_k)}{\tilde{C}_{\langle 1,3\rangle}(x_k)} < \frac{\tilde{C}_{\langle 1,2\rangle}(x_0)}{\tilde{C}_{\langle 1,3\rangle}(x_0)} = \frac{C_{\langle 1,2\rangle}}{C_{\langle 1,3\rangle}} = g(x(k),x(k)).$$

Therefore, $g(x(k),x(k+1)) < g(x(k),x(k))$ and $x(k+1) > x(k)$.

Now we have to show if $x(k) < x_0$, then $x(k+1) < x_0$. Since $\tilde{C}_{\langle 1,2\rangle}$ is increasing with x, we have $\tilde{C}_{\langle 1,2\rangle}(x_0) > \tilde{C}_{\langle 1,2\rangle}(x(k))$. Let $l^{overlay}_{\langle 1,3\rangle}$ denote aggregate overlay traffic on physical link $\langle 1,3\rangle$. After overlay's optimization as in (12), $l^{*overlay}_{\langle 1,3\rangle}(x_0) < l^{*overlay}_{\langle 1,3\rangle}(x(k))$. And we have

$$l^{*overlay}_{\langle 1,3\rangle}(x(k)) = (d^{(1,3)overlay} - x(k+1))f^{(1,3)}_{\langle 1,3\rangle}(x(k))$$
$$l^{*overlay}_{\langle 1,3\rangle}(x_0) = (d^{(1,3)overlay} - x_0)f^{(1,3)}_{\langle 1,3\rangle}(x_0).$$

Since $f^{(1,3)}_{\langle 1,3\rangle}(x)$ is increasing with x, $f^{(1,3)}_{\langle 1,3\rangle}(x_0) > f^{(1,3)}_{\langle 1,3\rangle}(x(k))$, therefore we must have $x(k+1) < x_0$. ∎