This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

Reverse Engineering TCP/IP-like Networks using Delay-Sensitive Utility Functions

John Pongsajapan and Steven H. Low
Engineering & Applied Science, Caltech

***Abstract*— TCP/IP can be interpreted as a distributed primal-dual algorithm to maximize aggregate utility over source rates. It has recently been shown that an equilibrium of TCP/IP, if it exists, maximizes the same delay-insensitive utility over both source rates and routes, provided pure congestion prices are used as link costs in the shortest-path calculation of IP. In practice, however, pure dynamic routing is never used and link costs are weighted sums of both static as well as dynamic components. In this paper, we introduce delay-sensitive utility functions and identify a class of utility functions that such a TCP/IP equilibrium optimizes. We exhibit some counter-intuitive properties that any class of delay-sensitive utility functions optimized by TCP/IP necessarily possess. We prove a sufficient condition for global stability of routing updates for general networks. We construct example networks that defy conventional wisdom on the effect of link cost parameters on network stability and utility.**

I. Introduction and Summary

Any TCP congestion control algorithm can be interpreted as carrying out a distributed primal-dual algorithm over the Internet to maximize aggregate utility, see e.g. [12], [13], [17]–[21], [25] for unicast and [4], [10], [25] for multicast. All of these works assume that routing is given and fixed at the timescale of interest, and TCP, together with active queue management (AQM), attempt to maximize aggregate utility over source rates. The paper [26] studies cross-layer utility maximization at the timescale of route changes, mainly for the special case of *pure* dynamic routing. In this paper, we extend the results of [26] in several ways.

As in [26], we focus on the situation where a *single* minimum-cost route (shortest path) is selected for each source-destination pair (Section II). This models IP routing in the current Internet within an Autonomous Systems using common routing protocols such as OSPF [22][1] or RIP [8]. For joint congestion control and routing optimization using multiple paths, see, e.g., [2], [5]–[7], [9], [11], [12], [14], [15], [23]. Routing is typically updated at a much slower timescale than TCP–AQM. We model this by assuming that TCP and AQM converge instantly to equilibrium after each route update to produce source rates and "congestion prices" for that update period. These congestion prices may represent delays or loss probabilities across network links. They determine the next routing update in the case of dynamic routing. Thus TCP–AQM/IP form a feedback system where routing interacts with congestion control in an iterative process. We are interested in the equilibrium and stability properties of this iterative process.

[1]Even though OSPF implements a shortest-path algorithm, it allows multiple equal-cost paths to be utilized. Our model ignores this feature.

To simplify notation, we will henceforth use TCP–AQM/IP and TCP/IP interchangeably.

We assume routing is chosen to minimize the weighted sum $ap_l + b\tau_l$ of congestion prices p_l and propagation delays τ_l along the path. When $b = 0$ (pure dynamic routing), [26] characterizes the exact condition under which an equilibrium of TCP/IP exists, and proves that such an equilibrium maximizes delay-insensitive utility over both rates and routes. In practice, however, pure dynamic routing is never used because of its instability. Instead, both weights a and b are typically nonzero, a case for which no result is available.

To reverse engineer TCP/IP networks with nonzero weights a and b, we introduce in Section III delay-*sensitive* utility functions that depend on not only source rates but also (propagation) delays. We identify a class $\mathcal{C}$ of delay-sensitive utility functions that is implicitly optimized by TCP/IP. As for the $b = 0$ case, we characterize the exact condition under which TCP/IP has an equilibrium and prove that such an equilibrium maximize utility functions in $\mathcal{C}$ over both rates and routes. As the relative weight $a/b \to \infty$, the utility functions in $\mathcal{C}$ become delay-insensitive and these results reduce to those proved in [26] for pure dynamic routing. We exhibit some counter-intuitive properties of class $\mathcal{C}$ utility functions, and prove that any (other) class of utility functions that TCP/IP optimizes necessarily possess some "strange" properties.

We prove in Section IV that, for general networks, if the weight a is small enough, only minimum-propagation-delay paths are selected. This implies that if all source-destination pairs have unique minimum-propagation-delay paths, then equilibrium of TCP/IP exists and is (globally) asymptotically stable. It is often believed that decreasing a helps ensure routing stability. We prove that this may not be the case if not all source-destination pairs have unique minimum-propagation-delay paths. Indeed, for a general network, its equilibrium and stability properties are the same as a modified network whose routing is based on *pure* congestion prices p_l, a network that is prone to routing instability. More surprisingly, there exists networks where reducing the weight a can destabilize an originally stable equilibrium.

It is conjectured in [26] that there is generally an inevitable tradeoff between utility maximization and stability in TCP/IP networks. In particular, as the weight a increases, the routing is conjectured to become more unstable but the achievable utility higher. We show however how to construct a network that has *any given* utility profile as a function of the weight a.

0743-166X/07/$25.00 ©2007 IEEE

II. MODEL

We use the same model as in [26]. In general, we use small letters to denote vectors, e.g., x with x_i as its ith component; capital letters to denote matrices, e.g., H, W, R, or constants, e.g., L, N, K^i; and script letters to denote sets of vectors or matrices, e.g., $\mathcal{W}_s$, $\mathcal{W}_m$, $\mathcal{R}_s$, $\mathcal{R}_m$. Superscript is used to denote vectors, matrices, or constants pertaining to source i, e.g., y^i, w^i, H^i, K^i.

A. Network

A network is modeled as a set of L unidirectional links shared by a set of N source-destination pairs, indexed by i (we will also refer to the pair simply as "source i"). Each link l has a finite capacity $c_l > 0$ and a delay $\tau_l > 0$ across the link, i.e., it takes τ_l to process and propagate a packet from one end of the link to the other, excluding queueing delay. Let $c = (c_l, l = 1, \ldots, L)$ and $\tau = (\tau_l, l = 1, \ldots, L)$.

There are K^i acyclic paths for source i represented by a $L \times K^i$ 0-1 matrix H^i where

$$H^i_{lj} \;=\; \begin{cases} 1, & \text{if path } j \text{ of source } i \text{ uses link } l \\ 0, & \text{otherwise} \end{cases}$$

Let $\mathcal{H}^i$ be the set of all columns of H^i that represents all the available paths to i under single-path routing. Define the $L \times K$ matrix H as

$$H \;=\; [H^1 \;\ldots\; H^N]$$

where $K := \sum_i K^i$. H defines the topology of the network.

Let w^i be a $K^i \times 1$ vector where the jth entry represents the fraction of i's flow on its jth path such that

$$w^i_j \geq 0 \;\; \forall j \quad \text{and} \quad \mathbf{1}^T w^i = 1$$

where $\mathbf{1}$ is a vector of an appropriate dimension with the value 1 in every entry. We require $w^i_j \in \{0,1\}$ for single-path routing, and allow $w^i_j \in [0,1]$ for multi-path routing. Collect the vectors w^i, $i = 1, \ldots, N$, into a $K \times N$ block-diagonal matrix W. Let $\mathcal{W}_s$ be the set of all such matrices corresponding to single path routing defined as

$$\{W | W = \text{diag}(w^1, \ldots, w^N) \in \{0,1\}^{K\times N}, \mathbf{1}^T w^i = 1\}$$

Define the corresponding set $\mathcal{W}_m$ for multi-path routing as:

$$\{W |\; W = \text{diag}(w^1, \ldots, w^N) \in [0,1]^{K\times N},\; \mathbf{1}^T w^i = 1\}$$

As mentioned above, H defines the set of acyclic paths available to each source, and represents the network topology. W defines how the sources load balance across these paths. Their product defines a $L \times N$ routing matrix $R = HW$ that specifies the fraction of i's flow at each link l. The set of all single-path routing matrices is

$$\mathcal{R}_s \;=\; \{\; R \mid R = HW, W \in \mathcal{W}_s \;\} \tag{1}$$

and the set of all multi-path routing matrices is

$$\mathcal{R}_m \;=\; \{\; R \mid R = HW, W \in \mathcal{W}_m \;\} \tag{2}$$

The difference between single-path routing and multi-path routing is the integer constraint on W and R. A single-path routing matrix in $\mathcal{R}_s$ is an 0-1 matrix:

$$R_{li} \;=\; \begin{cases} 1, & \text{if link } l \text{ is in a path of source } i \\ 0, & \text{otherwise} \end{cases}$$

A multi-path routing matrix in $\mathcal{R}_m$ is one whose entries are in the range $[0,1]$:

$$R_{li} \begin{cases} > 0, & \text{if link } l \text{ is in a path of source } i \\ = 0, & \text{otherwise} \end{cases}$$

The path of source i is denoted by $r^i = [R_{1i} \;\; \ldots \;\; R_{Li}]^T$, the ith column of the routing matrix R.

B. TCP–AQM/IP

We consider the situation where TCP–AQM operates at a faster timescale than routing updates. We assume a *single* path is selected for each source-destination pair that minimizes the sum of the link costs in the path, for some appropriate definition of link cost. In particular, traffic is not split across multiple paths from the source to the destination even if they are available. This models, e.g., IP routing within an Autonomous System. We focus on the timescale of the route changes, and assume TCP–AQM is stable and converges instantly to equilibrium after a route change. As in [17], we will interpret the equilibria of various TCP and AQM algorithms as solutions of a utility maximization problem defined in [12]. Different TCP algorithms solve the same prototypical problem (3) with different utility functions; see e.g. [17], [19], [25] for the utility functions for various popular TCP proposals.

Specifically, suppose each source i has a utility function $U_i(x_i, d_i)$ which depends on both its (total transmission) rate x_i and the end-to-end propagation delay d_i. Given a routing matrix R, we assume

$$d_i \;:=\; d_i(R) \;=\; \sum_{l=1}^{L} R_{li}\tau_l$$

Hence the delay d_i depends only on routing R and not on congestion in the path. The routing matrix R is in $\mathcal{R}_s$ for single-path routing and in $\mathcal{R}_m$ for multi-path routing. Note that in the multi-path case, d_i is the *traffic-weighted average* of propagation delays along its paths. We assume that utility functions are strictly concave for fixed d_i. The special case where the utility function $U_i(x_i) = U_i(x_i, d_i)$ depends on its rate x_i but not on the delay d_i is studied in [26]. Here, we focus on the delay-sensitive case.

Given a routing matrix R, $U_i(x_i, d_i)$ is a function only of rate x_i. Let $R(t) \in \mathcal{R}_s$ be the (single-path) routing in period t. Given a $R(t)$, let the equilibrium rates $x(t) = x(R(t))$ and prices $p(t) = p(R(t))$ generated by TCP–AQM in period t, respectively, be the optimal solutions of the constrained maximization problem

$$\max_{x\geq 0} \;\; \sum_i U_i(x_i, d_i) \qquad \text{s. t. } \;\; R(t)x \;\leq\; c \tag{3}$$

and its Lagrangian dual

$$\min_{p\geq 0} \sum_i \max_{x_i\geq 0} \left(U_i(x_i, d_i) - x_i \sum_l R_{li}(t)p_l \right) + \sum_l c_l p_l \tag{4}$$

The prices $p_l(t)$, $l = 1, \ldots, L$, are measures of congestion, such as queueing delays or loss probabilities [17], [19]. We assume that the link costs in period t are

$$z_l(t) \quad = \quad a p_l(t) + b\tau_l \tag{5}$$

where $a \geq 0$, $b \geq 0$, and $\tau_l > 0$ are constants. Based on these costs, each source computes its new route $r^i(t+1) \in \mathcal{H}^i$ individually that minimizes the sum of link cost in its path:

$$r^i(t+1) \quad = \quad \arg\min_{r^i \in \mathcal{H}^i} \sum_l z_l(t) r_l^i \tag{6}$$

Recall that τ_l in (5) are propagation delays across links l. If $p_l(t)$ represents the queueing delays at links l and $a = b = 1$, then $z_l(t)$ represent total delays across links l. The protocol parameters a and b determine the responsiveness of routing to network traffic: $a = 0$ corresponds to static routing, $b = 0$ corresponds to purely dynamic routing, and the larger the ratio of a/b, the more responsive routing is to network traffic. They determine whether an equilibrium exists, whether it is stable, and the achievable utility at equilibrium. The paper [26] focuses on the case of $b = 0$; we study the general case here.

An equivalent way to specify the TCP–AQM/IP system as a dynamical system, at the timescale of route changes, is to replace (3)–(4) by their optimality conditions. The routing is updated according to (combining (5) and (6))

$$r^i(t+1) = \arg\min_{r^i \in \mathcal{H}^i} \sum_l (a p_l(t) + b\tau_l) r_l^i, \quad \text{for all } i \tag{7}$$

where $p(t)$ and $x(t)$ are given by

$$\sum_l R_{li}(t) p_l(t) \quad = \quad \left[\frac{\partial U_i}{\partial x_i}(x_i(t), d_i)\right]^+ \text{ for all } i \tag{8}$$

$$\sum_i R_{li}(t) x_i(t) \begin{cases} \leq c_l & \text{if } p_l(t) \geq 0 \\ = c_l & \text{if } p_l(t) > 0 \end{cases} \text{ for all } l \tag{9}$$

$$x(t) \geq 0, \quad p(t) \geq 0 \tag{10}$$

This set of equations describe how the routing $R(t)$, rates $x(t)$, and prices $p(t)$ evolve. Note that $x(t)$ and $p(t)$ depend on $R(t)$ only through (8)–(10), implicitly assuming that TCP–AQM converges instantly to an equilibrium given the new routing $R(t)$.

We say that (R^*, x^*, p^*) is an *equilibrium of TCP/IP* if it is a fixed point of (3)–(6), or equivalently, (7)–(10), i.e., starting from routing R^* and associated (x^*, p^*), the above iterations yield (R^*, x^*, p^*) in the subsequent periods.

C. The joint optimization problem

Definition 1: A *delay-sensitive* utility function is a continuously differentiable function $U(x, d)$ from $[0, \infty) \times [0, \infty)$ to $[-\infty, \infty)$, that satisfies the following properties:

1) $\forall$ fixed $d > 0$, $U(x, d)$ is strictly concave in x.
2) $\forall d > 0, x > 0$, $U(x, d)$ and $\frac{\partial U}{\partial x}(x, d)$ are finite.
3) $(\forall x > 0), (\forall d_1, d_2 \text{ s.t. } 0 < d_1 < d_2) : U(x, d_1) > U(x, d_2)$.
4) $(\exists D > 0), (\forall 0 < d < D), (\exists X(d) > 0), (\forall x < X(d)) : \frac{\partial U}{\partial x}(x, d) > 0$.

Essentially, a delay-sensitive utility function is defined so that the source always gains utility from reducing propagation delay. If propagation delay is too high, the source can choose not to transmit. Otherwise, for fixed delay, the source's utility increases with transmission rate, possibly up to some limit. We assume all sources on the network have delay-sensitive utility functions $U_i(x_i, d_i)$.

We adapt the single-path delay-insensitive network optimization problem from [26] to a delay-sensitive network optimization problem:

$$\max_{R \in \mathcal{R}_s, x \geq 0} \sum_{i=1}^{N} U_i \left(x_i, \sum_{l=1}^{L} R_{li} \tau_l \right) \qquad \text{s.t. } Rx \leq c \tag{11}$$

Its Lagrangian dual is:

$$\min_{p \geq 0} \sum_{i=1}^{N} \max_{x_i \geq 0} \max_{r^i \in \mathcal{H}^i} \left(U_i(x_i, d_i) - x_i \sum_{l=1}^{L} R_{li} p_l \right) + \sum_{l=1}^{L} c_l p_l \tag{12}$$

where r^i is the ith column of R with $r_l^i = R_{li}$. This problem maximizes utility over both rates and routes.

Define the Lagrangian [1]:

$$L(R, x, p) = \sum_{i=1}^{N} \left(U_i(x_i, d_i) - x_i \sum_{l=1}^{L} R_{li} p_l \right) + \sum_{l=1}^{L} c_l p_l$$

Then we can express the primal and dual problems respectively as:

$$V_{sp} = \max_{R \in \mathcal{R}_s} \max_{x \geq 0} \min_{p \geq 0} L(R, x, p)$$
$$V_{sd} = \min_{p \geq 0} \max_{R \in \mathcal{R}_s} \max_{x \geq 0} L(R, x, p)$$

If we allow sources to use multiple paths, the corresponding problems are:

$$V_{mp} = \max_{R \in \mathcal{R}_m} \max_{x \geq 0} \min_{p \geq 0} L(R, x, p)$$
$$V_{md} = \min_{p \geq 0} \max_{R \in \mathcal{R}_m} \max_{x \geq 0} L(R, x, p)$$

The TCP/IP dynamical system is described by (3)–(6), or equivalently, (7)–(10).

D. Review: delay-insensitive utility functions

In this subsection, we consider the special case where the utility functions $U_i(x_i, d_i) = U_i(x_i)$ depend only on rates x_i but not on propagation delays d_i. There are three sub-cases: *i)* $a > 0, b = 0$; *ii)* $a = 0, b > 0$; *iii)* $a > 0, b > 0$.

For the first case where $a > 0, b = 0$ in (5), i.e., IP uses only congestion prices p_l generated by TCP–AQM as link costs, it is shown in [26] that TCP/IP maximizes aggregate utility over both rates and routing when an equilibrium exists.

Theorem 1 ([26]): Suppose $a > 0$ and $b = 0$ in (5). Then:

1) An equilibrium (R^*, x^*, p^*) of TCP/IP exists if and only if there is no duality gap between (11) and (12).
2) In this case, the equilibrium (R^*, x^*, p^*) is a solution of (11) and (12).

Moreover, in that case, there is no penalty in not splitting the traffic among multiple paths.

Theorem 2 ([26]): $V_{sp} \leq V_{sd} = V_{mp} = V_{md}$.

One of the open questions raised in [26] is the characterization of TCP/IP equilibrium in the other two cases where $b > 0$ in (5), i.e., when IP uses propagation delay, exclusively or not, as link costs. It is shown in [24] that for any delay-*insensitive* utility function $U(x)$, there exists a network with sources using this utility function, where TCP/IP equilibrium exists but does not solve (11) and (12). See [24] for explicit construction of such networks.

We now show that TCP/IP turns out to maximize a class of delay-*sensitive* utility functions when $b > 0$.

III. Delay-Sensitive Network Optimization

In this section, we consider the case where the utility functions $U_i(x_i, d_i)$ depend both on rates x_i and on propagation delays d_i. We identify a class $\mathcal{C}$ of utility functions for which TCP/IP, with $a > 0$ and $b = 1$ in (5), does maximize aggregate utility at equilibrium, when equilibrium exists. We analyze the properties of $\mathcal{C}$, and then derive properties that any class of utility functions that TCP/IP implicitly maximizes at equilibrium must possess.

We start with the case where $a = 0$ or $b = 0$ in (5), i.e., if all links use only the propagation delays τ_l as link costs, or if all links use only the congestion prices p_l generated by TCP–AQM as link costs. In this case, it can be shown that for every delay-sensitive utility function $U(x, d)$, there exists a network with sources using this utility function, where TCP/IP equilibrium exists but does not solve (11) and (12). See [24] for explicit construction of such networks.

Hence we consider the case where both $a > 0$ and $b > 0$.

A. Reverse engineering for link cost $ap_l + \tau_l$

In this subsection, we assume that IP uses $ap_l + \tau_l$ as link cost, i.e., $a > 0$ and $b = 1$ in (5). Consider the class $\mathcal{C}$ of functions $U(x, d)$ that can be written as:

$$U(x, d) = V(x) - a^{-1}xd$$

where $V(x)$ is a continuously differentiable function from $[0, \infty)$ to $[-\infty, \infty)$ so that $V(x)$ is strictly concave increasing, and $\forall x > 0$, $V(x)$ and $V'(x)$ are finite. In [24] we show that functions in $\mathcal{C}$ are delay-sensitive utility functions defined in Definition 1. Every utility function in $\mathcal{C}$ has two components: $V(x)$ measures the benefit of throughput x; $a^{-1}xd$ measures the penalty of delay d weighted by the throughput x. Note that a larger throughput x increases both the benefit of throughput and the penalty due to delay. The relative importance is determined by the (relative) weight a on congestion price in the link cost used in routing decisions: the larger the weight a, the more important the throughput benefit and the less important the delay penalty. In the limiting case as $a \to \infty$, corresponding to pure dynamic routing (i.e., $b = 0$), the utility function become delay-insensitive and our results below reduce to those established in [26] for the $b = 0$ case.

Specializing to utility functions in $\mathcal{C}$, the optimization problem (11) reduces to:

$$\max_{R \in \mathcal{R}_s, x \geq 0} \sum_{i=1}^{N} \left(V_i(x_i) - a^{-1}x_i \sum_{l=1}^{L} R_{li}\tau_l \right) \quad \text{s.t. } Rx \leq c$$

Consider the Lagrangian

$$L(R, x, p) = \sum_{i}^{N} (V_i(x_i) - x_i \sum_{l=1}^{L} R_{li}(p_l + a^{-1}\tau_l)) + \sum_{l=1}^{L} p_l c_l$$

and the dual problem $D(p) := \max_{R \in \mathcal{R}_s, x \geq 0} L(R, x, p)$:

$$D(p) = \max_{x \geq 0} \sum_{i=1}^{N} [V_i(x_i) - x_i \min_{r^i \in \mathcal{H}^i} \sum_{l=1}^{L} R_{li}(p_l + a^{-1}\tau_l)] + \sum_{l=1}^{L} p_l c_l$$

The minimization over R in the dual problem appears to involve minimal-cost routing using $p_l + a^{-1}\tau_l$ as route cost. But this is the same as minimal-cost routing using $ap_l + \tau_l$ as route cost. This suggests that TCP/IP might solve the joint optimization problem with utility functions in $\mathcal{C}$. This is indeed the case.

Our first main results, and their proofs, are analogous to Theorem 1 and Theorem 2 but for $b > 0$ and delay-sensitive utility functions. They say that the equilibrium of TCP/IP, when exists, solves the joint utility maximization over routes and rates and its dual problem. Moreover, in that case, there is no penalty in not splitting traffic among multiple paths. Their proofs are in [24].

Theorem 3: Suppose all utility functions are in $\mathcal{C}$, $a > 0$ and $b = 1$.

1) An equilibrium (R^*, x^*, p^*) of TCP/IP exists if and only if there is no duality gap between (11) and (12).
2) In this case, the equilibrium (R^*, x^*, p^*) is a solution of (11) and (12).

Theorem 4: Suppose utility functions are in $\mathcal{C}$. Then $V_{sp} \leq V_{sd} = V_{mp} = V_{md}$.

B. Counter-intuitive properties of class $\mathcal{C}$

In this subsection, we exhibit some counter-intuitive properties of the class $\mathcal{C}$ of utility functions. Specifically, we present example networks where, because the penalty term $a^{-1}xd$ is proportional to throughput x, these utility functions can underutilize link capacities or available network paths.

The first example illustrates a network equilibrium which is in the *strict* interior of the feasible set $Rx \leq c$, contrary to what the traditional TCP model with delay-insensitive utility functions would predict.

Remark 1: Given any utility function in $\mathcal{C}$, there exists a network where TCP/IP underutilizes links.

Proof: $\frac{\partial U}{\partial x}(x, d) = V_i'(x_i) - a^{-1}d$. If $V'(x) - a^{-1}d = 0$, then x is the rate that maximizes $U(x, d)$ for fixed d, since $U(x, d)$ is strictly concave for fixed d.

Choose any $c > 0$ and set $\tau = aV'(c)$. Note that $\tau > 0$, since $V(x)$ is strictly increasing. Consider a network with one link, whose capacity is $2c$. A flow whose path is just this link will have rate c at equilibrium, since $\frac{\partial U}{\partial x}(x, d) = V'(c) - a^{-1}\tau = V'(c) - V'(c) = 0$. But this leaves the link underutilized, since the link has capacity $2c$. □

The second example shows that extra paths that would be utilized if the utility functions were delay-insensitive may not

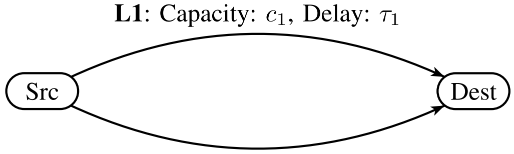

Fig. 1. Network 1

be utilized by utility functions in $\mathcal{C}$. It also illustrates Theorem 3 and Theorem 4.

Consider any $U(x,d) = V(x) - a^{-1}xd$ in $\mathcal{C}$. It can be shown that there exist $\tau > 0$, $c > 0$ so that $\frac{\partial U}{\partial x}(c,\tau) > 0$. Consider Network 1 in Figure 1 with $c_1 = c, \tau_1 = \tau, c_2 = \infty, \tau_2 = \tau + a\frac{\partial U}{\partial x}(c,\tau)$. Suppose there is only one flow, Flow 1, and it can choose between routes **R1**: **L1** and **R2**: **L2**.

Suppose Flow 1 is initially on route **R1**. It achieves rate c with propagation delay τ and utility $U(c,\tau)$. Then **R1** has route cost $\tau + a\frac{\partial U}{\partial x}(c,\tau)$ and **R2** has route cost $\tau + a\frac{\partial U}{\partial x}(c,\tau)$. The initial routing is an equilibrium routing since all flows are using minimal cost routes.

Theorem 3 then implies that there is no duality gap. Theorem 4 then implies that $V_{sp} = V_{mp}$ and there is no benefit in multi-path routing, i.e., there is no benefit in utilizing route **R2**. This seems counter-intuitive – with all delay-insensitive networks, there is always a benefit in utilizing previously unutilized routes. Indeed, we can show directly that utilizing **R2** increases the average propagation delay experienced by the flow, which turns out to be suboptimal regardless of the amount of extra throughput.

Remark 2: Given any utility functions in $\mathcal{C}$, it is suboptimal to use route **R2** in Network 1 with $c_1 = c, \tau_1 = \tau, c_2 = \infty, \tau_2 = \tau + a\frac{\partial U}{\partial x}(c,\tau)$, even when the flow is allowed to distribute its traffic over multiple paths.

Proof: Let kc specify the throughput on link **L2**, so that $\frac{k}{k+1}$ specifies the fraction of traffic sent over link **L2** (where the rest is sent over link **L1**). Then the total throughput is given by $kc + c$, and the weighted average propagation delay is

$$\frac{k\left[\tau + a\frac{\partial U}{\partial x}(c,\tau)\right] + \tau}{k+1}$$

We claim that for every $k > 0$, we must have

$$U(c,\tau) \geq U\left(kc + c, \frac{k\left[\tau + a\frac{\partial U}{\partial x}(c,\tau)\right] + \tau}{k+1}\right)$$

To verify this, plug in our utility function:

$$\begin{aligned} V(x) - a^{-1}c\tau &\geq U\left(kc + c, \frac{k\left(\tau + a\frac{\partial U}{\partial x}(c,\tau)\right) + \tau}{k+1}\right) \\ &\geq V(kc+c) - a^{-1}c\left(kaV'(c) + \tau\right) \\ V'(c) &\geq \frac{V(kc+c) - V(c)}{kc} \end{aligned}$$

which is true since $V(x)$ is concave. $\square$

C. Alternative classes

We have shown in the last subsection some counter-intuitive properties of class $\mathcal{C}$ utility functions. We now show that *any* class of delay-sensitive utility functions that TCP/IP optimizes, using $ap_l + \tau_l$ as link cost, must possess some "strange" properties.

Define

$$M(U,d) := \lim_{c\to\infty} U(c,d), \tag{13}$$

which computes the maximum possible utility at each delay d for any delay-sensitive utility function $U(x,d)$. The next result says that any class $\mathcal{B}$ of utility functions that TCP/IP optimizes either contain functions that are not strictly increasing in throughput (contrary to the traditional delay-insensitive utility functions of TCP models), or are "discontinuous" in throughput in the space of utility functions, or can be discontinuous in delay for large enough delay.

Theorem 5: Suppose $\mathcal{B}$ is any class of delay-sensitive utility functions such that when TCP/IP equilibrium exists, the equilibrium solves (11) and (12) with utility functions in $\mathcal{B}$. Then $\mathcal{B}$ must have at least one of the following three properties:

1) $\exists U(x,d) \in \mathcal{B}, d > 0$ so that $U(x,d)$ is not strictly increasing in x.
2) $\forall U_1(x,d) \in \mathcal{B}$, $\forall \epsilon > 0$, we have $U_2(x,d) := U_1(x+\epsilon, d)$ is not in $\mathcal{B}$.
3) $\exists U(x,d) \in \mathcal{B}$, $D > 0$ such that $f(d) := M(U,d)$ is finite and discontinuous for all $d > D$.

Proof: See [24]. $\square$

In particular, it can be checked that class $\mathcal{C}$ utility functions possess the first two properties.

IV. Stability and Utility of Routing Policies

In this section, we analyze the effects on stability and utility of dynamic routing.

A. Sufficient condition for stability

Denote the set of paths $\mathcal{G}^i \subseteq \mathcal{H}^i$ with minimal propagation delay for flow i by:

$$\mathcal{G}^i := \left\{ r^i \in \mathcal{H}^i : \tau^T r^i = \min_{s^i \in \mathcal{H}^i} \tau^T s^i \right\}$$

Denote the set of paths $\mathcal{F}^i \subseteq \mathcal{H}^i$ without minimal propagation delay for flow i by:

$$\mathcal{F}^i := \mathcal{H}^i - \mathcal{G}^i$$

Define $q(R)$, a function that computes the equilibrium congestion price vector for a given routing matrix $R \in \mathcal{R}_s$. We assume that it implicitly depends on an arbitrary, fixed network $(L, N, \mathcal{F}^i, \mathcal{G}^i, \mathcal{H}^i, \mathcal{R}_s, K^i, U_i, \tau, c)$:

$$q(R) := \arg\min_{p\geq 0}\max_{x\geq 0}\left(\sum_i^N U_i(x_i,d_i) - p^T Rx\right) + p^T c$$

In this subsection, we show that with sufficiently small a, all flows will choose only minimal propagation delay paths. For notational simplicity, all of the following functions, lemmas,

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2007 proceedings.

and theorems in this subsection implicitly depend on an arbitrary, fixed network $(L, N, \mathcal{F}^i, \mathcal{G}^i, \mathcal{H}^i, \mathcal{R}_s, K^i, U_i, \tau, c)$.

Define $h(x, y)$, a function that will be used to simplify notation:

$$h(x,y) = \begin{cases} \frac{x}{y} & \text{if } y > 0 \\ \infty & \text{if } y \leq 0 \end{cases}$$

Define $a_{\#}$ as follows:

$$a_{\#} := \min_{R \in \mathcal{R}_s} \min_{0<i\leq N} \begin{cases} \max\limits_{m^i \in \mathcal{G}^i} \min\limits_{r^i \in \mathcal{F}^i} b(m^i, r^i) & \text{if } \mathcal{F}^i \neq \emptyset \\ \infty & \text{if } \mathcal{F}^i = \emptyset \end{cases}$$

where $b(m^i, r^i) := h\left(\tau^T(r^i - m^i), q(R)^T(m^i - r^i)\right)$. It can be checked that $a_{\#}$ is strictly positive.

Theorem 6: Suppose $a < a_{\#}$. Then $\forall R \in \mathcal{R}_s, \forall 0 < i \leq N, \exists m^i \in \mathcal{G}^i$ such that

$$(aq(R) + \tau)^T m^i < (aq(R) + \tau)^T r^i, \quad \forall r^i \in \mathcal{F}^i \tag{14}$$

In other words, all flows on the network will choose a path with minimal propagation delay in the next iteration.

Proof: We manipulate the right hand side of (14):

$$(aq(R) + \tau)^T m^i < (aq(R) + \tau)^T r^i$$
$$aq(R)^T(m^i - r^i) < \tau^T(r^i - m^i)$$

But by inspecting the definition of $h(x, y)$, this inequality holds if

$$a < h\left(\tau^T(r^i - m^i), q(R)^T(m^i - r^i)\right)$$

since $\tau^T(r^i - m^i) > 0$ if $m^i \in \mathcal{G}^i$ and $r^i \in \mathcal{F}^i$.

The formal part of the theorem is then easy to see. It implies that for any current routing, and for every flow, a path with minimal propagation delay has strictly lower cost than any path without minimal propagation delay. Therefore no flow will select any path without minimal propagation delay. Therefore every flow will select a path with minimal propagation delay. □

Theorem 7: Suppose all source-destination pairs on a network have unique minimum-propagation-delay paths. Then if $a < a_{\#}$, TCP/IP has an asymptotically stable equilibrium.

Proof: Each flow only has one path with minimal propagation delay. Applying Theorem 6, each flow will always select the same path, and it will always do this from any routing. □

Corollary 1: Suppose every path in a network has different propagation delay. Then if $a < a_{\#}$, TCP/IP has an asymptotically stable equilibrium.

Proof: If every path in the network has different propagation delay, every source-destination pair on a network has a unique minimum-propagation-delay path, so the result of Theorem 7 applies. □

B. Counter-intuitive properties: stability

It is often believed that decreasing the weight a on the traffic sensitive component of link cost can always stabilize dynamic routing; e.g., see [3], [16], [26]. In this subsection, we show that this is generally not true.

It is easy to see that not all networks can be stabilized by decreasing a. For instance, consider Network 1 with $c_1 = c_2 = c > 0$, $\tau_1 = \tau_2 = \tau > 0$. Suppose there is one flow, Flow 1, that can choose between routes **R1**: **L1** and **R2**: **L2**. If it has a delay-insensitive utility function $U(x)$, then for all $a > 0$, this network has no equilibrium.

The next two results are less obvious. The first says that if a is small enough, then a network with routing based on $ap_l + \tau_l$ behaves like a modified network with routing based on p_l, which seems prone to routing instability [26].

Theorem 8: Give a network with $a < a_{\#}$. Consider the modified network obtained by deleting all paths without minimal propagation delay from the original network. Then the original network with routing based on $ap_l + d_l$ has the same equilibrium and stability properties as the modified network with routing based on p_l.

Proof: Consider the TCP/IP dynamical system on the modified network when link costs are p_l:

$$(p(t))^T r^i(t+1) = \min_{r^i \in \mathcal{G}^i} (p(t))^T r^i, \text{ for all } i$$

$$(p(t), x(t)) = \arg \min_{p \geq 0} \max_{x \geq 0} (\sum_i^N U_i(x_i, d_i) - p^T R(t) x) + p^T c$$

$$\forall i, t : r^i(t) \in \mathcal{G}^i$$

Consider the TCP/IP dynamical system on the original network when link costs are $ap_l + \tau_l$:

$$(ap(t) + \tau)^T r^i(t+1) = \min_{r^i \in \mathcal{H}^i} (ap(t) + \tau)^T r^i, \text{ for all } i.$$

$$(p(t), x(t)) = \arg \min_{p \geq 0} \max_{x \geq 0} (\sum_i^N U_i(x_i, d_i) - p^T R(t) x) + p^T c$$

$$\forall i, t : r^i(t) \in \mathcal{H}^i$$

The next lemma, whose proof is in [24], implies that these dynamical systems are equivalent. Since they are equivalent, they share the same equilibrium and stability properties. □

Lemma 1: Suppose we have some network, and routing policy on this network is such that $a < a_{\#}$. Then for any attainable price vectors p so that $p = q(R)$ for some $R \in \mathcal{R}_s$, and for all $0 < i \leq N$,

$$p^T r^i = \min_{s^i \in \mathcal{G}^i} p^T s^i, \text{ where } r^i \in \mathcal{G}^i \tag{15}$$

if and only if

$$(ap + \tau)^T r^i = \min_{s^i \in \mathcal{H}^i} (ap + \tau)^T s^i, \text{ where } r^i \in \mathcal{H}^i \tag{16}$$

The second counter-intuitive result says that it is possible to destabilize a network by decreasing the static component a in link cost.

Theorem 9: Consider any delay-sensitive or delay-insensitive utility function $U(x, d)$. There exists a network with sources using this utility function, and constants $a_3 > a_2 \geq a_1 > 0$ so that:

1) The network is stable for $a \in (0, a_1)$.

2) The network is unstable for $a \in (a_2, a_3)$.
3) The network is stable for $a \in (a_3, \infty)$.

We will prove the theorem by exhibiting such a network. Suppose $U(x,d)$ is an arbitrary delay-sensitive or delay-insensitive function. It can be shown [24] that it is possible to choose parameters c_1, c_2, τ_1, τ_2 that satisfy the following inequalities:

$$\frac{\partial U}{\partial x}(c_1, \tau_1) > \frac{\partial U}{\partial x}(c_2, \tau_2) > 0 \tag{17}$$

$$c_2 > c_1 > 0 \tag{18}$$

$$\tau_2 > \tau_1 > 0 \tag{19}$$

Consider Network 1 with those parameters. Suppose there are two flows. Suppose the possible routes are **R1**: **L1** and **R2**: **L2**. Suppose that Flow 1 is constrained to **R1**, and Flow 2 can choose between **R1** and **R2**. Denote this instance of Network 1 by Network 2.

Define function $A_3(U, c_1, \tau_1, c_2, \tau_2)$, where U is a delay-insensitive or delay-sensitive utility function and the rest of the parameters are in $\mathbb{R}$, as follows:

$$A_3(U, c_1, \tau_1, c_2, \tau_2) := \frac{\tau_2 - \tau_1}{\frac{\partial U}{\partial x}(c_1, \tau_1) - \frac{\partial U}{\partial x}(c_2, \tau_2)}$$

Lemma 2: Consider any delay-sensitive or delay-insensitive utility function $U(x,d)$. Suppose c_1, τ_1, c_2, τ_2 satisfy (17) through (19) with $U(x,d)$. Network 2 with all sources using $U(x,d)$ is stable for all $a > A_3(U, c_1, \tau_1, c_2, \tau_2)$.

Proof: We show that routing converges from every possible initial condition.

Suppose Flow 2 is on route **R1**. Then Flow 1 and Flow 2 share **L1** equally, and both achieve rate $\frac{c_1}{2}$ with propagation delay τ_1 and utility $U(\frac{c_1}{2}, \tau_1)$. Then **R1** has route cost $a\frac{\partial U}{\partial x}(\frac{c_1}{2}, \tau_1) + \tau_1$ and **R2** has route cost τ_2. It can be verified that $a > A_3(U, c_1, \tau_1, c_2, \tau_2)$ implies that:

$$a\frac{\partial U}{\partial x}\left(\frac{c_1}{2}, \tau_1\right) + \tau_1 > \tau_2$$

So **R2** is a lower cost route than **R1**.

To show asymptotic stability, it is then sufficient to show that the routing where Flow 2 is on **R2** is an equilibrium. Suppose Flow 2 is on **R2**. Then Flow 1 achieves rate c_1 with propagation delay τ_1 and utility $U(c_1, \tau_1)$, and Flow 2 achieves rate c_2 with propagation delay τ_2 and utility $U(c_2, \tau_2)$. Then **R1** has route cost $a\frac{\partial U}{\partial x}(c_1, \tau_1) + \tau_1$ and **R2** has route cost $a\frac{\partial U}{\partial x}(c_2, \tau_2) + \tau_2$.

Consider Flow 2's route choice at the next routing iteration. It can be verified that $a > A_3(U, c_1, \tau_1, c_2, \tau_2)$ implies that

$$a\frac{\partial U}{\partial x}(c_1, \tau_1) + \tau_1 > a\frac{\partial U}{\partial x}(c_2, \tau_2) + \tau_2$$

So **R2** is a lower cost route than **R1**. Therefore **R2** is minimal cost, and so this routing is an equilibrium routing. □

Define function $A_2(U, c_1, \tau_1, c_2, \tau_2)$, where U is a delay-insensitive or delay-sensitive utility function and the rest of the parameters are in $\mathbb{R}$ as follows:

$$A_2(U, c_1, \tau_1, c_2, \tau_2) := \frac{\tau_2 - \tau_1}{\frac{\partial U}{\partial x}(\frac{c_1}{2}, \tau_1)}$$

Lemma 3: Consider any delay-sensitive or delay-insensitive utility function $U(x,d)$. Suppose c_1, τ_1, c_2, τ_2 satisfy (17) through (19) with $U(x,d)$. Then $A_2(U, c_1, \tau_1, c_2, \tau_2) < A_3(U, c_1, \tau_1, c_2, \tau_2)$.

Proof: See [24]. □

Lemma 4: Consider any delay-sensitive or delay-insensitive utility function $U(x,d)$. Suppose c_1, τ_1, c_2, τ_2 satisfy (17) through (19) with $U(x,d)$. Network 2 with all sources using $U(x,d)$ is unstable for all a satisfying $A_2(U, c_1, \tau_1, c_2, \tau_2) < a < A_3(U, c_1, \tau_1, c_2, \tau_2)$.

Proof: Suppose Flow 2 is on **R2**. Then Flow 1 achieves rate c_1 with propagation delay τ_1 and utility $U(c_1, \tau_1)$, and Flow 2 achieves rate c_2 with propagation delay τ_2 and utility $U(c_2, \tau_2)$. Then **R1** has route cost $a\frac{\partial U}{\partial x}(c_1, \tau_1) + \tau_1$ and **R2** has route cost $a\frac{\partial U}{\partial x}(c_2, \tau_2) + \tau_2$. Consider Flow 2's route decision at the next routing iteration. It can be verified that $a < A_3(U, c_1, \tau_1, c_2, \tau_2)$ implies

$$a\frac{\partial U}{\partial x}(c_1, \tau_1) + \tau_1 < a\frac{\partial U}{\partial x}(c_2, \tau_2) + \tau_2.$$

So Flow 2 next chooses route **R1**. Then Flow 1 and Flow 2 share **L1** equally, and both achieve rate $\frac{c_1}{2}$ with delay τ_1 and utility $U(\frac{c_1}{2}, \tau_1)$. Then **R1** has route cost $a\frac{\partial U}{\partial x}(\frac{c_1}{2}, \tau_1) + \tau_1$ and **R2** has route cost τ_2.

It can be verified that $a > A_2(U, c_1, \tau_1, c_2, \tau_2)$ implies

$$a\frac{\partial U}{\partial x}(\frac{c_1}{2}, \tau_1) + \tau_1 > \tau_2.$$

So Flow 2 next chooses route **R2**.

This implies that Flow 2's routing oscillates between **R1** and **R2**, so the network is unstable. □

Proof (Theorem 9). We choose c_1, τ_1, c_2, τ_2 so that they satisfy (17) through (19). Then consider Network 2 with all sources using $U(x,d)$. Set $a_1 = a_\#$ for this network. Theorem 6 implies that for $a < a_1$, the network is stable. We then set $a_2 = A_2(U, c_1, \tau_1, c_2, \tau_2)$ and $a_3 = A_3(U, c_1, \tau_1, c_2, \tau_2)$. The lemmas in this subsection then establish the desired result. □

C. Counter-intuitive properties: utility

The paper [26] analyzes the effects of increasing a on the time-averaged aggregate utility for a ring network, and a randomly generated network. On the ring network, time-averaged aggregate utility approached the maximum possible time-averaged aggregate utility for any a, as a increased. On the generated network, time-averaged aggregate utility increased until routing stability set in, and then decreased.

In this subsection, we show that the effects of increasing a on time-averaged aggregate utility are network dependent. In particular, we show how to construct a network with *any* given utility profile as a function of the weight a.

For every delay-insensitive $U(x)$, there exist $k^+ > k^* > k^- > 0$ and $g > 0$ so that $g = U(k^+) - U(k^*) = U(k^*) - U(k^-)$. Also, for every $z \in [-1, 1]$, there exists $k \in [k^-, k^+]$ so that $U(k) - U(k^*) = zg$, that is given by $k(z) := U^{-1}(U(k^*) + zg)$. This is easy to see since $U(x)$ is strictly monotone increasing.

Consider the following network $N(j,z)$, parameterized by $j > 0$, and $z \in [-1,1]$, which is defined to be Network 1 with parameters $c_1 = k^*, \tau_1 = jU'(k^*), c_2 = k(z), \tau_2 = 2jU'(k^*)$, with one flow, Flow 1, choosing between routes **R1**: **L1** and **R2**: **L2**.

Denote the time-averaged utility of the network under routing policy $ap_l + d_l$ by $T(N,a)$.

Lemma 5: For every $j > 0$, $z \in [-1,1]$, network $N(j,z)$ has the following properties:

1) $\forall a_1 \in [0,j], a_2 \in [0,j]$: $T(N,a_1) = T(N,a_2)$
2) $\forall a_1 \in (j,\infty), a_2 \in (j,\infty)$: $T(N,a_1) = T(N,a_2)$
3) $\forall a_1 \in (0,j], a_2 \in (j,\infty)$: $T(N,a_2) = T(N,a_1) + \frac{zg}{2}$

Proof: Suppose current routing is **R2**. Then **R1** has route cost $jU'(k^*)$ and **R2** has route cost $2jU'(k^*) + aU'(k(z))$. By inspection, for every $a \geq 0$, **R1** has less cost than **R2**. Therefore routing on the next iteration will be **R1**.

Suppose current routing is **R1**. Then **R1** has route cost $jU'(k^*) + aU'(k^*)$ and **R2** has route cost $2jU'(k^*)$. By inspection, if $a \leq j$ then **R1** is a lower cost route than **R2**, and if $a > j$, then **R2** is a strictly lower cost route than **R1**.

Flow 1's utility on route **R1** between routing changes is $U(k^*)$. Flow 1's utility on route **R2** between routing changes is $U(kz)$.

If $a \leq j$, then the network is stable with Flow 1 on **R1** and the aggregate utility is $U(k^*)$. If $a > j$, then the network oscillates between **R1** and **R2** and achieves time-averaged utility $U(k(z)) + U(k^*)/2$. The excess utility gained by increasing a from less than j to more than j is given by:

$$\frac{U(k(z)) + U(k^*)}{2} - U(k^*) = \frac{U(k(z)) - U(k^*)}{2} = \frac{zg}{2}$$

□

Definition 2: A utility-versus-a profile is a pair of vectors (x,y) such that $|x| = |y| > 0$, $\forall i$: $x_i > 0$, and $\forall i < |x|$: $x_i < x_{i+1}$.

Definition 3: A network N matches a utility-versus-a profile (x,y) if there exists $\lambda > 0$ so that:

If $|x| > 1$,

1) $\forall i$ s.t. $1 < i < |x|$, $\forall a_1$ s.t. $x_{i-1} < a_1 \leq x_i$, $\forall a_2$ s.t. $x_i < a_2 < x_{i+1}$: $T(N,a_2) - T(N,a_1) = \lambda y_i$.
2) $\forall a_1$ s.t. $0 < a_1 \leq x_1$, $\forall a_2$ s.t. $x_1 < a_2 < x_2$: $T(N,a_2) - T(N,a_1) = \lambda y_1$.
3) $\forall a_1$ s.t. $x_{|x|-1} < a_1 \leq x_{|x|}$, $\forall a_2$ s.t. $x_{|x|} < a_2 < \infty$: $T(N,a_2) - T(N,a_1) = \lambda y_{|x|}$.

If $|x| = 1$, $\forall a_1$ s.t. $0 < a_1 \leq x_1$, $\forall a_2$ s.t. $x_1 < a_2 < \infty$: $T(N,a_2) - T(N,a_1) = \lambda y_1$.

In other words, for all i, the time-averaged aggregate utility of the network increases by λy_i at $a = x_i$.

Theorem 10: For every utility-versus-a profile (x,y), there exists a network with sources using delay-insensitive utility functions that matches this profile.

Proof: Consider any utility-versus-a profile (x,y). Define the normalized y as $y^* := |\max_i y_i|^{-1} y$. Construct the network N^* by taking the union of networks $N_1 \ldots N_{|y|}$ where $\forall i$, $N_i := N(x_i, y_i^*)$. (The subnetworks are entirely disjoint in the union network).

It is easy to see that the network N^* matches profile (x,y) with $\lambda = \frac{g}{2|\max_i v_i|}$. □

V. Conclusion

In this paper, we have attempted to reverse engineer TCP/IP-like networks. We have identified a class of delay-sensitive utility functions that is implicitly optimized by an equilibrium of TCP/IP. We have characterized its equilibrium and stability properties for general networks, and exhibited several counter-intuitive results.

Many issues are still open. First, we believe $\mathcal{C}$ is the only class of utility functions that TCP/IP with link cost $ap_l + d_l$ jointly optimize, but we have not been able to prove this. Second, the (delay-insensitive) utility functions obtained from reverse engineering TCP algorithms in the literature, assuming routing is fixed, are all strictly increasing in throughput. Hence the (delay-sensitive) utility functions obtained from reverse engineering TCP/IP should ideally be strictly increasing in throughput when routing is held fixed. However, this is not the case with class $\mathcal{C}$ utility functions. This mismatch should be reconciled.

References

[1] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1995.

[2] D. Bertsekas, E. Gafni, and R. G. Gallager. Second derivative algorithms for minimum delay distributed routing in networks. *IEEE Trans. on Communications*, 32(8):911–919, 1984.

[3] Dimitri P. Bertsekas. Dynamic behavior of shortest path routing algorithms for communication networks. *IEEE Transactions on Automatic Control*, pages 60–74, February 1982.

[4] S. Deb and R. Srikant. Congestion control for fair resource allocation in networks with multicast flows. In *Proceedings of the IEEE Conference on Decision and Control*, December 2001.

[5] R. G. Gallager. A minimum delay routing algorithm using distributed computation. *IEEE Trans. on Communications*, 25(1):73–85, 1977.

[6] R. G. Gallager and S. J. Golestani. Flow control and routing algorithms for data networks. In *Proc. of the 5th International Conf. Comp. Comm.*, pages 779–784, 1980.

[7] Jiayue He, Mung Chiang, and Jennifer Rexford. TCP/IP interaction based on congestion price: Stability and optimality. In *Proc. IEEE ICC*, June 2006.

[8] C. Hedrick. Routing Information Protocol. IETF RFC 1058, `http://www.faqs.org/rfcs/rfc1058.html`, June 1988.

[9] K. Kar, S. Sarkar, and L. Tassiulas. Optimization based rate control for multipath sessions. In *Proceedings of 7th International Teletraffic Congress (ITC)*, December 2001.

[10] Koushik Kar, Saswati Sarkar, and Leandros Tassiulas. Optimization based rate control for multirate multicast sessions. In *Proceedings of IEEE Infocom*, April 2001.

[11] F. P. Kelly and T. Voice. Stability of end-to-end algorithms for joint routing and rate control. In *ACM Sigcomm Computer Communication Review*, volume 35, pages 5–12, April 2005.

[12] Frank P. Kelly, Aman Maulloo, and David Tan. Rate control for communication networks: Shadow prices, proportional fairness and stability. *Journal of Operations Research Society*, 49(3):237–252, 1998.

[13] S. Kunniyur and R. Srikant. End-to-end congestion control: utility functions, random losses and ECN marks. *IEEE/ACM Transactions on Networking*, 11(5):689 – 702, October 2003.

[14] I. Lestas and G. Vinnicombe. Combined control of routing and flow: a multipath routing approach. In *Proc. 43rd IEEE Conference on Decision and Control*, December 2004.

[15] X. Lin and N. B. Shroff. Utility maximization for communication networks with multi-path routing. *IEEE Trans. on Automatic Control*, 51(5):766–781, May 2006.

[16] S. Low and P. Varaiya. Dynamic behavior of a class of adaptive routing protocols (IGRP). In *Proceedings of Infocom'93*, pages 610–616, March 1993.

[17] Steven H. Low. A duality model of tcp and queue management algorithms. *IEEE/ACM Trans. Netw.*, 11(4):525–536, 2003.

[18] Steven H. Low and David E. Lapsley. Optimization flow control, I: basic algorithm and convergence. *IEEE/ACM Transactions on Networking*, 7(6):861–874, December 1999. `http://netlab.caltech.edu`.

[19] Steven H. Low, Larry L. Peterson, and Limin Wang. Understanding tcp vegas: a duality model. *J. ACM*, 49(2):207–235, 2002.

[20] L. Massoulie and J. Roberts. Bandwidth sharing: objectives and algorithms. *IEEE/ACM Transactions on Networking*, 10(3):320–328, June 2002.

[21] Jeonghoon Mo and Jean Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5):556–567, October 2000.

[22] J. Moy. OSPF Version 2. IETF RFC 2328, `http://www.faqs.org/rfcs/rfc2328.html`, April 1998.

[23] F. Paganini. Congestion control with adaptive multipath routing based on optimization. In *Proc. of CISS*, March 2006.

[24] John Pongsajapan. Optimization and stability of tcp/ip with delay-sensitive utility functions. Master's thesis, California Institute of Technology, 2006.

[25] R. Srikant. *The Mathematics of Internet Congestion Control*. Birkhauser, 2004.

[26] Jiantao Wang, Lun Li, Steven H. Low, and John C. Doyle. Cross-layer optimization in tcp/ip networks. *IEEE/ACM Trans. Netw.*, 13(3):582–595, 2005.